Screening Psychological Behaviors with Machine Learning

Diagnosis of Early Autism

by

Vivien Cabannes

Département de Mathématiques et Applications
École Normale Supérieure

Guillermo Sapiro

Guillermo Sapiro, Supervisor, Duke University

Jean-Michel Morel, MVA Supervisor

Vivien Cabannes

Thesis submitted in partial fulfillment of the requirements for the degree of
Master of Science in the Department of Mathématiques
at the École Normale Supérieure de Cachan

Contents

Chapter 1

Introduction

Recent development of machines and algorithms are spreading high hopes to build better information system. Gathering more data is the key point of this going-on revolution. Cinema lovers would remember when, in the late 2000's, was discovered, in the archives of the Museo del Cine in Buenos Aires, an original cut of Fritz Lang's Metropolis [19]; French archeologists, René Ginouvès [15].

This numerical revolution is particularly dreamy when it comes to biology and medicine. Numerous are the reasons to believe in a future breakthrough for health care. With nowadays knowledge about gene expression, bacteria, diet or antibiotic impact, our epoch would probably soon appear as a dark age for medicine. Several billionaires have shared their desire to seek for immortality, for example, believing in genetic disease prevention, if not eradication, thanks to new genome decoding abilities.

Yet, analyzing dataset require methodology. To perform such a task, statistical methods provide a mathematical framework along with theoretical guarantees about putative findings. Effort behind the scene are leading the way to smarter algorithms, and broader impact of so-called "artificial intelligence". Receiving a good amount of media coverage is the research on "big data", as the quest to automatically handle huge datasets (*e.g.* DNA sequences over an entire population).

The above excitement is visible in academia, where are created interdisciplinary initiatives, such as the *Information Initiative at Duke University*, founded in 2012 by Robert Calderbank. Situated in Durham, NC, also known as the "City of Medicine", Duke is famous for its hospital. Naturally benefiting of this atmosphere, data scientists are trying to bring their humble stone in an under-construction scientific edifice. This is the case of Guillermo Sapiro, who kindly host me for this internship, working on early detection of autism.

1.1 Autism Screening

Autism is characterized by difficulty communicating and interacting with others, along with limited interests or repetitive behaviors, impeding ability to function in society.

Yet, several developmental disorders, expressed by a wide range of symptoms, are named under autism, officials rather refer as ASD for *Autism spectrum Disorder* [3].

In their report of 2010, the US Center for Disease Control estimated a prevalence of 24 ‰ of ASD among boys aged 8 years and of 5 ‰ among girls at the same age [4]. Yet, autism causes, typology and treatment are barely understood, inhibiting adequate response to support autistic people in becoming able to stand by themselves. To detect underlying pathology, comprehend pathologies on the spectrum, analyze effectiveness of an intervention, data science could have a major impact.

Even autism screening is a long and complex process, that require a long appointment at the psychologist, appointment often expensive and requesting a long wait. Thus, if ASD can be detected as early as 18 month old, most children aren't diagnosed before 4 year old! And it is even worth for children coming from poor background [8]. While actually, parents often express first concerns before their child's first birthday [18]. There is a consensus that this gap represents a crucial waist of time to better answer child need and prevent the development of unadapted behaviors [9].

Before Hand Work

Work has been done to help parents screening their child without a psychologist. The MCHAT, standing for Modified Checklist for Autism in Toddlers, is a popular survey for parents to fill, giving a good idea about child autism risk. Introduced by Diana Robins in the early 2000's [25], it consists in rating toddler social behavior, such as "answering to one's name", "smiling back", "pointing at objects" and so on. Yet, survey methods are highly subjective, only to be considered as raising a warning flag. Final diagnosis must fall under the approval of a clinical practitioner!

Psychological profiling has been investigated for decades. About ideas to perform such a task, one is to stimulate the patient and characterize in his reaction specific traits of character [22]. This idea drove the Autism Diagnostic Observation Schedule (ADOS), which is currently the main psychologist routine to screen for autism [1]. It consists of standardized interaction between the patient and a psychologist, furnishing plenty of occasion for the practitioner to notice occurrence of behaviors supposed discriminative to diagnose for autism. Thus diagnosis is a subjective observational inventories.

This repetitive assessment lasts a bit less than an hour. It is reasonable to think about casting this schedule on a human-machine interaction. Numerous are the reasons to get excited about it, providing a cheap, instantaneous, easy to access method for psychological screening. Moreover, recording the interaction will provide precious data to refine our analysis and understanding of autism. Furthermore, it could be a first milestone of a long exciting journey in revolutionizing psychological screening.

So was developed at Duke University the application *Autism & Beyond* [16], featuring in Apple research kit. Psychologist have designed small videos supposed to trigger specific responses in autistic kids. While those videos are playing for a child on a tablet or a smartphone, the front camera record the child reaction. Algorithms are then supposed to analyze the children videos to screen for autism.

Data Collection

The data were acquired on 102 children, among those 20 were autistic, 1 didn't received diagnosis, and all the other were supposed not autistic, even if not always completely test about it. The data were labelled with autism diagnosis. MCHAT survey was jointly provided, yet the full ADOS diagnosis was kept by psychologists.

Figure 1.1: A frame on a video played for toddlers

Three videos were shown in a lab setting to the kids. Two presents stuffed animals interacting as shown on Figure 1.1. On this figure, a reckless elephant destroy a tower that an enthusiastic frog has constructed just before. All this in front of a stone face bear. A last one, presenting bubbles moving on the screen was shown twice. Those video were played on a tablet in the lab, while the child was sitting approximatively 1 meter away on one of his or her parent lap. Meanwhile, the front camera recorded the child reaction, leading to sets of four video per child.

Those reaction videos were preprocessed with a classical algorithm to extract landmarks on the face: 5 for each eyebrows, 6 for each eyes, 4 on the bridge, 1 on the tip, and 4 on the wings of the nose, 1 for each labial commissures, 8 for each lips (3 inside, 5 outside); for a total of 49 landmarks. With 30 frame per second, 3 minutes of videos, and landmark flatly extracted in two dimensions, the raw data dimension is over half a billion.

The landmarks extraction algorithm sometimes failed catching the face, which introduce missing frame in our dataset 14 % on average, the worse subject was only catch 24 % of the time on average on the four video (76 % of missing data). Moreover, the algorithm present some instability: detecting mother face instead of the child, overflow, among other.

1.2 Data Analysis

A lot of work has been done on understanding expression of faces, offering easy-to-use libraries to detect landmarks on the face and linked them with feelings. That's the reason why assuming face expression will be enough to diagnose the kids, every video

frames were preprocessed by extracting landmarks on the face. This allows to reduce the complexity and homogenize the data.

This contrasts with the psychologist perspective of noticing occurrence of special responses, like pointing, gazing at others, before analyzing when or how often they appears on a video tape [24]. Here is coming an other army of psychologists, not working on personality but on face expression.

Face Expression

The idea to link face observation with intrinsic state of mind to understand somebody feelings has been discussed by psychologists for more than a century. Early work begin with Duchenne [12] and Darwin [10]. They analyze the different muscle contractions and their combination as expression of feelings. Francis Galton, Darwin's cousin, also had intuition that a lot of information could be extracted from face features [14]. Among the first to express eugenist ideas, he wanted to analyze criminal based on their faces.

A lot has been made since then. Important contribution included by Paul Ekman's Facial Action Coding System (FACS) theory [13]. His idea was to recognize muscle contraction as action unit, and study the correlation between states of mind and combinations of actions unit. The neuroscientist Horace Barlow [5], Darwin's great grandson, points out how visual cortex recognize and analyze faces through convolution filters, leading the way to machine detection of those action unit through filter banks response [27].

Analysis Pipeline

Ideally, extracting muscular activity from landmarks would allow to discover instantaneous feelings. From the analysis of feeling, one could build a psychologic profile through behavioral analysis. Note that this work could be done, without much learning, mainly using psychologist knowledge. However, this ambitious pipeline is out of reach of our humble work and dataset.

Keeping it simple, one can extract features supposed discriminative, before feeding them in a classifier. There is a trade-off between complex classifier able to learn rich structure, and simpler one easier to understand and generalize. Finally, one could stick closer to psychologists intuition, trying to look at response that are supposed abnormal, eventually adding a data analysis step to refine or discover what if a normal response.

1.3 Thesis Organization

In a first chapter, we focus on extracting discriminative features on the dataset. First, by parameterizing the face with muscle activity. Then, trying to decompose the signal into simple components, through wavelets, dictionary learning or scale-space representation. Afterwards, we suggests, based on random process classification ideas, some meaningful transform, ending with a convolutional frameworks to describe the video

smartly. Finally, we draw deeper ideas if we were to think in term of pattern recognition, linking it with filtering, bag of words representation or dictionary learning.

In a second chapter, we build on those descriptors, classifier to discriminate the kids in order to latter screen for autism. We begin with rich and complex classifier, based on neighborhood analysis or generalized linear cutting. Then focusing on easy-to-understand learning, we mainly discover, through features selection, particularity on the dataset rather than offering a screening tool with statistical guarantees. At the end of the day, we turn toward psychologists educated learning, introducing a normal response framework.

Chapter 2

Features Engineering

When trying to understand a phenomenon, a abstract idea consists in separating its different causes. In our language, one could say the different sources of a signal. In a first part, we derive a more adapted face parameterization, by describing head and muscular activity rather than landmark position. Afterward, we process extracted signals to decompose it into simple components, remove noise and deal with missing frame, finally drifting to pattern recognition. Then, we suggest complementary layers to build on top, if we were to deal with a larger or more supervised database.

2.1 Face Parameterization

Let's re-parameterize the face to easily access muscular activity and head pose.

2.1.1 Head Pose Estimation

Autistic tend to make back and forth motion, or to orient their head more randomly. In consequence, let's retrieve from the 2D landmarks, a model of this 3D motion. considering the skull, the head is following a rigid transformation, characterized by a translation and a rotation, this rotation is naturally described by yaw, pitch and roll.

An Inverse Problem

Based on a 3D model of the face, it is easy to derive 2D landmarks from the position of the head. Let's note f_0 the 3D model of the face, f this face with a specific head pose.

$$f = R_x(\theta)\, R_y(\varphi)\, R_z(\psi) \left(f_0 + \begin{pmatrix} x_t \\ y_t \\ z_t \end{pmatrix} \right) = T(\eta) f_0$$

This equation could be developed and rewritten as

$$\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} \cos(\varphi)\cos(\psi) & -\cos(\varphi)\sin(\psi) & -\sin(\varphi) & x_t \\ \cos(\theta)\sin(\varphi) - \sin(\theta)\sin(\varphi)\cos(\psi) & \cos(\theta)\cos(\varphi) + \sin(\theta)\sin(\varphi)\sin(\psi) & -\sin(\theta)\cos(\varphi) & y_t \\ \sin(\theta)\sin(\psi) + \cos(\theta)\sin(\varphi)\cos(\psi) & \sin(\theta)\cos(\psi) - \cos(\theta)\sin(\varphi)\sin(\psi) & \cos(\theta)\cos(\varphi) & z_t \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_0 \\ y_0 \\ z_0 \\ 1 \end{pmatrix}$$

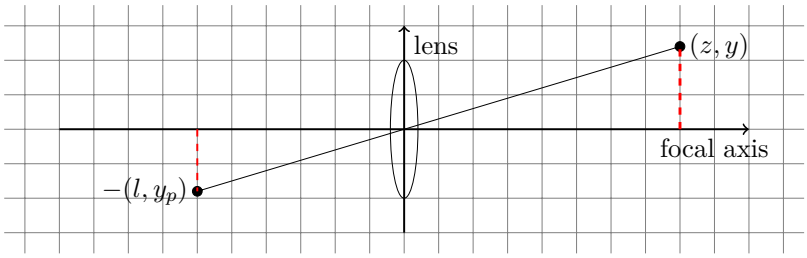

Figure 2.1: Projection of a 3D point in focal plan

Now f is seen through the camera, which act like a projection, as shown on Figure 2.1, thus if the z-axis correspond to the focal axis, this projection reads:

$$p\left(\begin{pmatrix} x \\ y \\ z \end{pmatrix}\right) \propto \frac{1}{z}\begin{pmatrix} x \\ y \end{pmatrix}$$

Collecting everything, we would like to retrieved η by matching $p \circ T(\eta) f_0$ with the observed landmarks. Denoting $(m_i)_i$ those observed landmarks and $(f_{0,i})_i$ their positions on our model, a variational formulation reads

$$\hat{\eta} = \arg\min_{\eta} \sum_i \|m_i - p \circ T(f_{i,0})\|^2$$

Due to the projection, this problem isn't convex. It seems smart to first derive z by hand before fixing it, then relaxing T to describe any affine transformation and be left with a convex minimization. With enough landmarks, the solution should be unique and give us back the rotation matrix. From this rotation matrix it is easy to invert the parameterization to get yaw, pitch and roll.

Hand Made Solution

To find yaw, pitch and depth, one should look at dilation and ratio between distance on the face. Considering Figure 2.2, it is possible to derive close formula to retrieved those values as:

$$z \approx \frac{y}{\sqrt{\frac{(y_1-y_2)^2}{(y_1+y_2)^2} + \frac{(y_1+y_2)^2}{4l^2}}}, \qquad \theta = \arcsin\left(\frac{z(y_1-y_2)}{y(y_1+y_2)}\right)$$

However, since there is no guarantee on the discriminative power of head pose features, one doesn't need to put to much energy in solving perfectly this problem, especially since algebraic formulation could be unstable. Thus our implementation stay informal only considering simple quantity as $(y_1 - y_2)/(y_1 + y_2)$ for yaw, or the area enclosed by the outside of the eyes and the tip of the nose to estimate depth.

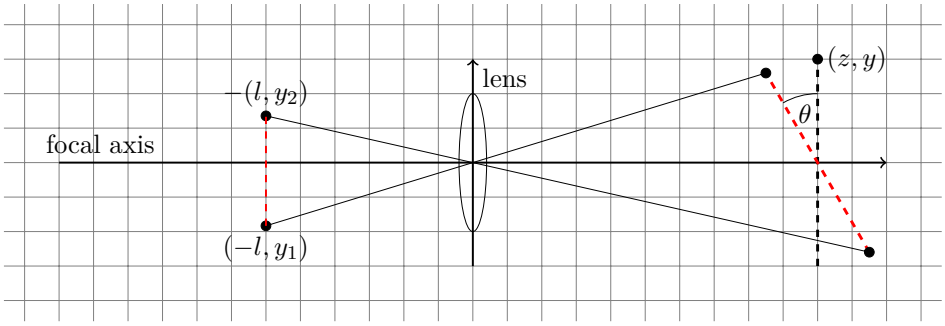

Figure 2.2: Determining yaw and depth

Figure 2.3 shows the features recorded on two children, while their mothers called their names on two videos. The non-autistic kid reacts when his name is called by orienting his head toward his mother as illustrated by yaw curve, while the autistic kid reaction is not as pronounced. On the opposite, the autistic kid moves more than the other one, showing probably difficulties to focus.

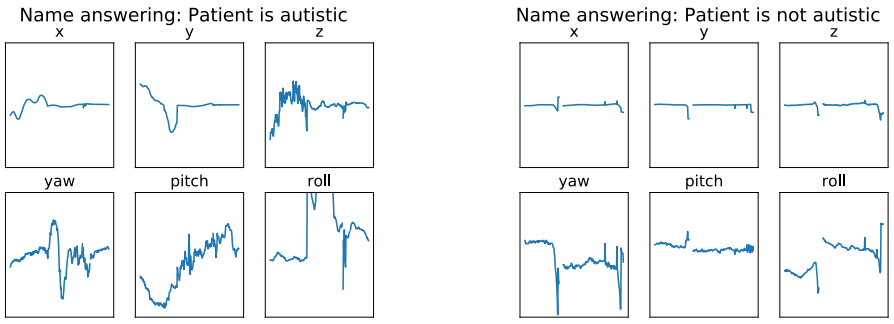

Figure 2.3: Head motion while mother call kid's name during video (time in abscise)

2.1.2 Facial Features Extraction

After describing general head motion, let's focus on muscular action on the face. In absence of a huge quantity of data, or subtile supervision, overfitting rapidly threaten as descriptors or model get too complex. Keeping it simple we will consider:

- Mouth width
- Mouth opening
- Mouth angle (smiling)
- Lips protrusion
- Raising eyebrows
- Pushing eyebrows aside
- Opening eyes

Normalization

To extract meaningful features, one should work on source of unwanted variability. Considering distances and angles enhance 2D translation and rotation invariance. Given head pose estimation, one can center the face for head pose invariance. Yet to avoid facing instability issues, we didn't address redressing rotational pose. Finally to remove head morphology variation, we described extension comparing to a distance at rest, supposed to be given by the median computed over the entire video – eventually taking the logarithm in base 2, so contracting the muscle by two appear as -1, and dilating it by two appear as +1.

2.1.3 Inverting Parameterization

The new parameterized space seems good to recognize action unit, or more generally perform filtering, clustering, and bag of word representation. In order to visualize the filters, clusters or words learnt by our methods, we would like to invert the parametrization. This allowed us to check for information lost by our parameterization.

First averaging all faces and adding depth to landmarks allow to get a 3D face model. One can then perform head pose and extend or contract landmarks to concord with the new parameterization values. This reconstruction even if not conceptually difficult, is a big piece of code. We could from it, extract average sets of features, and reconstruct an average face from this average sets of features.

Comparison

Let's now compare the real landmarks with the one reconstructed after considering this new parameterization. Figure 2.4 presents reconstruction based on slightly richer parameterization. If several differences can be mark down, the overall perception is similar, suggesting that the new parameterization isn't loosing crucial information to perform the screening.

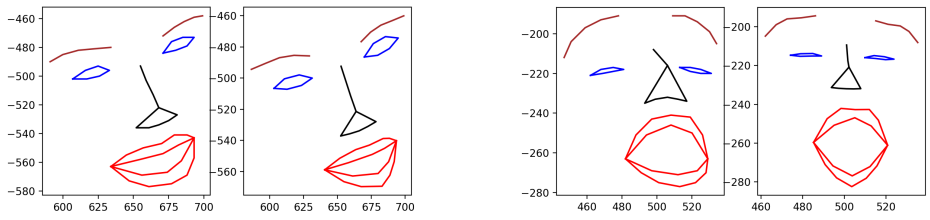

Figure 2.4: Comparisons between original (left) and reconstructed face (right)

2.2 Signal Processing

The extracted signals are high-dimensional, noisy and with frame missing, as anyone can notice on example furnished by Figure 2.5. Yet it seems really simple to understand for a human, the kid slightly turn is head over the entire video, make five momentary head turn that results in missing frames, plus some noise all along. Mathematically we would like to write the signal f as

$$f(t) = H_f(t).\left(\psi(t) + \sum_{i=1}^{5} a_i\phi_i(t) + \varepsilon(t)\right)$$

where ψ corresponds to the long-term slight turn, ϕ to the momentary ones, ε to noise, and $H_f \in \{0,1\}^{[0,T]}$ is modeling the missing frames.

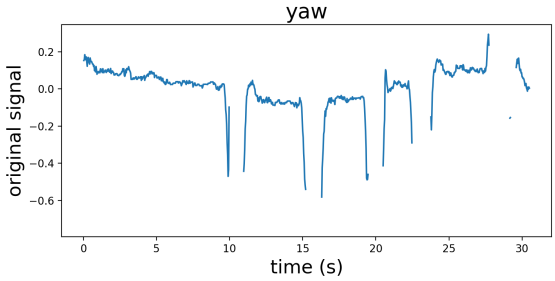

Figure 2.5: Example of signal

2.2.1 Sparse Representation

To make further analysis simple, let's try to decompose any signal f with simple components, or atoms ψ_i as

$$f(t) = H_f(t).\left(\sum_{i=1}^{n} a_i\psi_i\,(t) + \varepsilon(t)\right)$$

This linear relation could be cast in matrix formulation as

$$f = H.\,(\Psi a + \varepsilon) \tag{2.2.1}$$

where f (resp. ε) correspond to the vector $(f[t])_t$ (resp. $(\varepsilon[t])_t$, the bracket being used to underlying the discrete sampling over time). H has become a diagonal operator with $H_{tt} = 0$ if, and only if, the frame t is missing. Ψ is a called dictionary, each of its columns corresponding to an atom ψ_i, and a points out the component of the signal, aspiring to become its new descriptor.

Dictionary construction

To describe the signal by the coefficients a, we need to build a universal dictionary, to give sense to the comparison between two signals f and g according to the vectors a_f and a_g. This dictionary should contains all potential atoms. For yaw signal, atoms are quiet simple, it consists in a head turn, more or less pronounced, more or less fast, and arriving at a certain time. Thus if ψ represent a standardized head turn, we would the dictionary to contain

$$\left\{ t \to \lambda\psi\left(\frac{t-t_0}{s}\right) \middle| (\lambda, s, t_0) \in \mathbb{R} \times \mathbb{R}_+ \times [0, T] \right\}$$

where s is a scaling parameter representing the speed (or the frequency), λ is a amplitude parameter that could be removed cause redundant regarding a, and t_0 is a spacial parameter localizing the head turn. On a abstract level, such a dictionary is translation and scale invariant. Seeking for invariance to capture allow to automatize dictionary construction: one only have to design few mother atoms then.

Variational Formulation

To obtain the new representation, one should solve the linear equation (2.2.1). Yet, the atoms of the dictionary are usually not linearly independent, allowing several solutions. What we really want, and that the reason why we consider a maximum of atoms in the dictionary, is to achieve the sparsest representation of the signal. In other term we want to solve

$$\begin{array}{ll} \text{minimize} & \|a\|_0 = \#\{i \,|\, a_i \neq 0\} \\ \text{subject to} & f \approx H.\Psi a \end{array} \tag{2.2.2}$$

Capturing $f \approx H.\Psi a$ could be done through $\|f - H.\Psi a\| \leq \varepsilon$. Using the ℓ^2 norm allow to control this term with classical linear algebra. This problem is intrinsically hard to solve, leading to a bunch of heuristic methods, or relaxations to achieve a satisfying representation. In the following, we apprehend solutions from simpler to more complex options, between orthogonal dictionary all the way to dictionary learning.

2.2.2 Wavelets

Let's begin in the simple case where $\Psi \in \mathcal{O}$ is actually orthogonal. Seeking an orthogonal basis based on a mother wavelet translated and scaled have been investigated around the 1980's. Pioneer work of Yves Meyer proved that it was possible, before Ingrid Daubechies [11] found compactly supported ones, leading the way to fast implementation [6].

Minimization Scheme

Let's cast the problem as interpolating missing frames. We suggest to interpolate the missing frame by minimizing the complexity of the interpolation, where the complexity

is measure by the sparsity of the wavelet coefficients. For a current signal guess $f^{(l)}$, a greedy iterative scheme reads:

- Compute the wavelet transform of $f^{(l)}$, threshold coefficients to enhance sparsity, inverse the transform to get $f^{(l+\frac{1}{2})}$
- Take $f^{(l+1)}$ as $f^{(l+\frac{1}{2})}$ on missing frames and f otherwise and repeat the scheme

Algorithm 1: Wavelets Reconstruction

$f_r := 0$;
while *not converged* **do**
  $f_r \leftarrow \Psi \circ T_\beta \circ \Psi^*.f_r$;
  $f_r \leftarrow \text{proj}_{f+\ker H_f}.f_r = H_f.f + (1 - H_f)\, f_r$;

This alternative scheme could be seen in different ways to ensure it convergence. For example is T_β is a projector on $\overline{B}_{\ell^1}(0, \beta)$, this is nothing but an alternative projection algorithm to find a point in the intersection of two convex sets. For fancier thresholding, this remark generalize to proximal splitting methods.

Fast Wavelet Transform

The main issue when implementing this method is to faster the matrix multiplication $\Psi^* f$ or Ψa. It will be possible cause of the structure of the wavelet transform, which build a scale-space representation. The natural building scheme reads the following:

- At given given scale, compute signal details with an differential operator h
- Go to the next scale thanks to a smoothing operator g

Let's look at an example to make it clearer. Let's consider the following operators:

$$\begin{array}{lll} D.f[n] = f[n+1] - f[n] = h * f[n] & \text{where} & h = [1, -1] \\ G.f[n] = \frac{1}{2}\left(f[n+1] + f[n]\right) = g * f[n] & \text{where} & g = [\frac{1}{2}, \frac{1}{2}] \end{array}$$

The idea is to say recursively, we can describe the signal by $(d_i)_i$ where:

$$\begin{array}{ll} f_0 = (f[n])_n & \\ f_1 = \left(\frac{1}{2}\left(f_0[2n+1] + f_i[2n]\right)\right)_n, & d_0 = (f_0[2n+1] - f_0[2n])_n \\ \vdots & \vdots \\ f_{i+1} = \left(\frac{1}{2}\left(f_i[2n+1] + f_i[2n]\right)\right)_n, & d_i = (f_i[2n+1] - f_i[2n])_n \end{array}$$

If S denotes the subsampling operator $Sf[n] = f[2n]$, the wavelet transform reads

$$Wf = \left(SD\,(SG)^i\, f\right)_i$$

Figure 2.6 gives a visual example of this wavelet transform. In red are the detail coefficients, and in blue segments is the upscaled signal. The last figure shows the final transform, obtained by collecting all scale details.

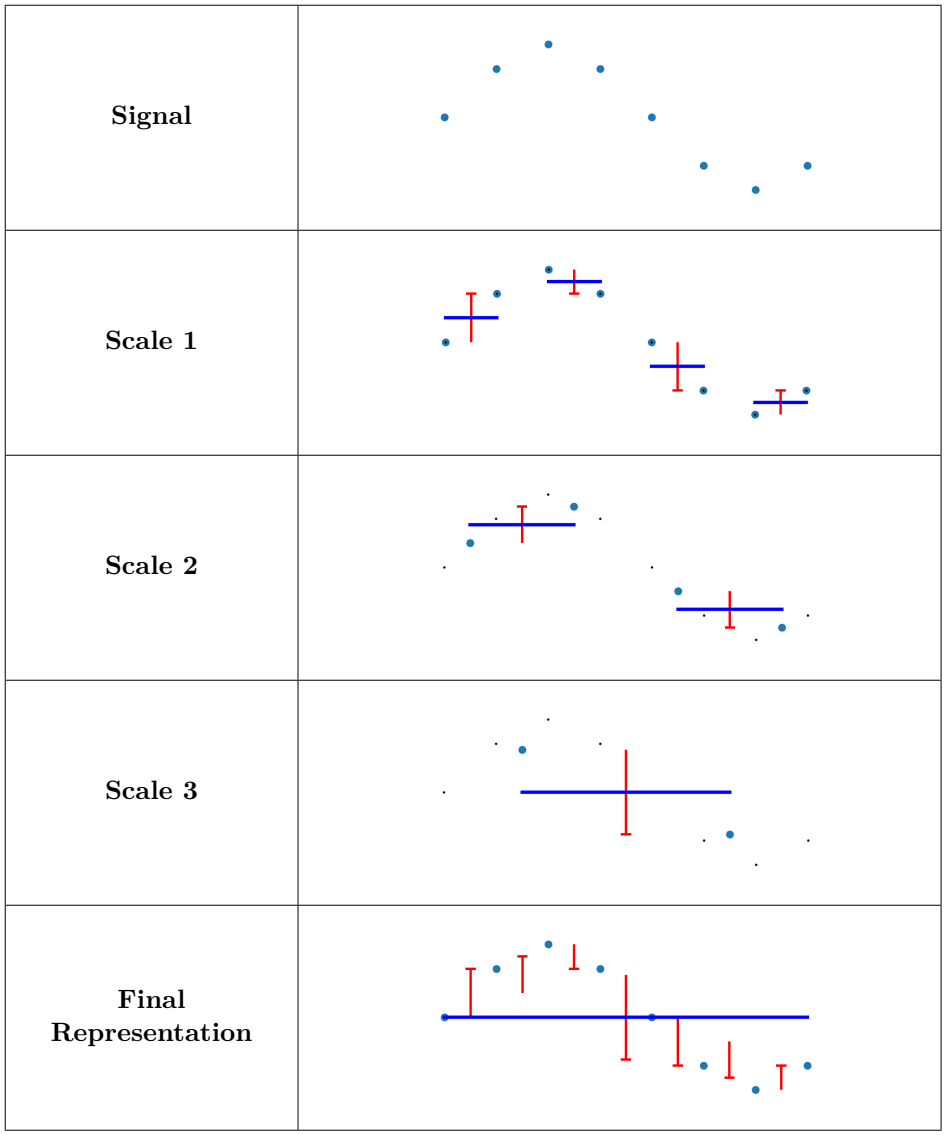

Figure 2.6: Haar Wavelet Transform

The main point is to notice that at each scale, one can divide the description of the signal by a factor 2, allowing to reduce computation to $n \log_2(n)$ rather than a basic matrix multiplication in n^2. It consists in cascading the operator one after the other as shown Figure 2.7.

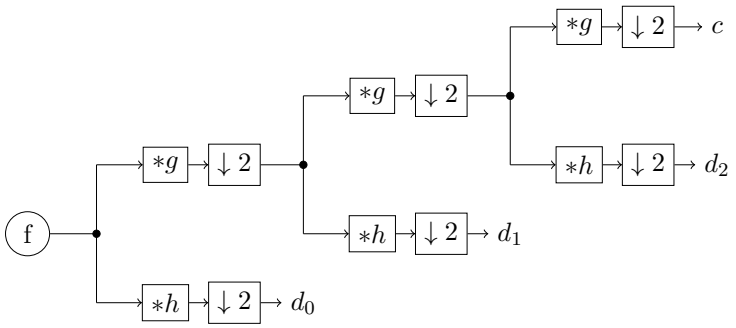

Figure 2.7: Fast Wavelet Transform as a Cascade

Inverse Transform

Under its matrix form, inverting the wavelet transform is really simple, it consists in taking its transpose. To do so let's look closer at this matrix. Let's restrict to a simple layer $(c, d) = (SG.f, SD.f)$. Daubechies wavelets assert that this is also an orthogonal transformation, thus once again, one can inverse considering the transpose

$$f = \begin{bmatrix} (SG)^* & (SD)^* \end{bmatrix} \begin{bmatrix} c \\ d \end{bmatrix} = G^*S^*c + D^*S^*d$$

Or, the adjoint of the subsampling is nothing but the upsampling operator reading:

$$Uf[n] = \begin{cases} f[n/2] & \text{if} \quad n \in 2\mathbb{N} \\ 0 & \text{otherwise} \end{cases}$$

And, the adjoint of a convolution, is nothing but the convolution with the reversed filter $\tilde{g}(t) = g(-t)$. Thus the inverse transform also have a cascade structure as shown Figure 2.8.

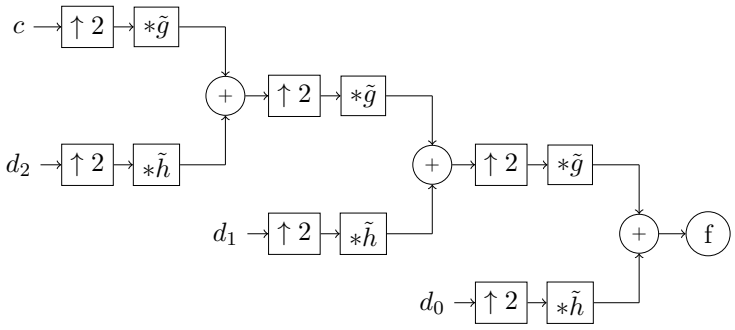

Figure 2.8: Fast Inverse Wavelet Transform

Results

Thanks to the orthogonality, we were able to make derivation and computation easier, yet orthogonal wavelets are not translation invariant. Accommodating to a pic that doesn't fall at the right place for its frequency could introduce some artifact. This is the case on Figure 2.9 where at second 22, we observed a amplification of the head turn by the reconstruction. Also the wavelets have the tendency to kill the signal were it is missing. Computation was made with Daubechies $D4$ wavelets (their form appear like watermark on the reconstructed signal, especially on missing frames).

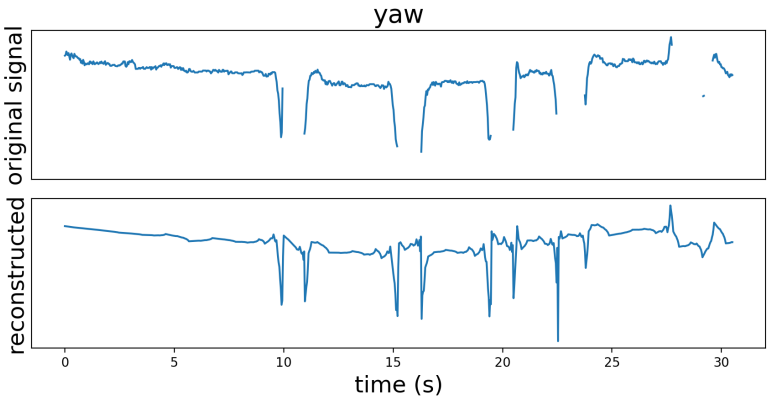

Figure 2.9: Reconstruction with Soft-Thresholding

Graphical Visualization

A lot of previous comments could be seen visually, and generalized, looking at the dictionary matrix represented in Figure 2.10. Two atoms are shown, a blue one could be placed at any time to reconstruct the signal, the red appears at larger scale, inter-step allowing cascading with subsampling but at the price of restricting this atom position in time. On this drawing, one picture how convolution faster the matrix multiplication $\Psi^* f$. Moreover, for Ψa which correspond to invert the wavelet transform, upsampling allow to keep a convolution structure with revert atoms.

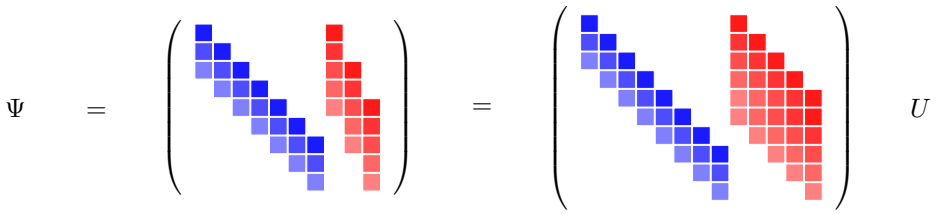

Figure 2.10: Dictionary invariance allows to faster matrix multiplication

2.2.3 Dictionary

Let retake the sparse reconstruction problem (2.2.2), now without restricting our dictionary. Several algorithms have been proposed to solve this problem, due to its application in compress sensing, matrix factorization or features selection. We can dissociate two types: heuristic methods, and convex relaxation.

Convex Relaxation

The classical relaxation of (2.2.2) consists in approaching the ℓ^0 pseudo-norm by the ℓ^1 and cast everything under the variational form:

$$\text{minimize} \quad \|f - H.\Psi a\|_2^2 + \lambda \|a\|_1 \tag{2.2.3}$$

Popular algorithms are balancing between minimizing the left data-fidelity term, before the right regularizing one.

Proximal Method. Let's focus on minimize $g(x) + h(x)$ with respect to x, using a splitting procedure. Assuming regularity of g, one can make sure not to increase to much $g(x)$ while decreasing $h(x)$ if staying at proximity. Mathematically regularity of g naturally translates as its differential being locally Lipschitz. And the idea to stay close of prior guess while minimizing h is captured by the proximal operator that propose to map prior guess x to

$$\text{prox}_{\gamma h}(x) = \arg\min_y \frac{1}{2} \|x - y\|^2 + \gamma h(y)$$

Once again, the Euclidean metric is used because of its computation easiness. Assuming ∇g is β-Lipschitz, and taking a step $\gamma < \beta/2$, the scheme

$$x^{(l+1)} = \text{prox}_{\gamma h}\left(x^{(l)} - \gamma \nabla g(x^{(l)})\right)$$

converge toward the minimizer. Some fancier variants suggest to speed up the algorithm by remembering prior steps. It could be a way to greedily approximate the Hessian and try to condition the descent similarly as Newton-Raphson.

Minimization Scheme. Coming back to (2.2.3) where $h = \lambda \|\cdot\|_1$

$$\text{prox}_{\gamma h}(x) = \left(\max\left\{0, 1 - \frac{\lambda\gamma}{|x_i|}\right\} x_i\right)_i$$

which is nothing but soft-thresholding, and $g = \|f - H\Psi \cdot\|_2^2$

$$\nabla g(a) = -2\left(H\Psi\right)^*\left(f - H\Psi a\right) = -2\Psi^* H\left(f - \Psi a\right)$$

where we used the fact that $H = H^* = H^2$. Thus the minimization scheme

Algorithm 2: LASSO Reconstruction

$a := 0$;
while *not converged* **do**
  $f_r = \Psi a$;
  $a \leftarrow a + 2\gamma \Psi^* H(f - f_r)$;
  $a \leftarrow T_\gamma a$

with $\gamma < 1/\left\|\Psi^* H \Psi\right\|_{\text{op}}$. Taking a step back from the variational formulation, one could change soft-thresholding by a hard one.

Heuristic Method

Let's rethink the original problem of findings principal signal components, where simple components are given by atoms in the dictionary Ψ. It is natural to consider the principal component as the one maximizing the response $\langle \psi_*, f \rangle$, before seeking the second one in $f - a_* H.\psi_*$. In other term, assuming unitary atoms, the scheme reads

- At step l, find the l-th principal component as $i_l = \arg\max_i \langle \psi_i, f - H.\Psi a_{l-1} \rangle$
- Refine signal approximation as $f^{(l)} = \sum_{k \leq l} a_{l,i_k} \psi_{i_k}$.

Algorithm 3: Matching Pursuit Reconstruction

$a := 0$;
while $\|f - H\Psi a\|_2 > \varepsilon$ **do**
  $i_l = \arg\max_i \Psi^*(f - H\Psi a)$;
  $I \leftarrow \{i_k\}_{k \leq l}$;
  $a \leftarrow (H\Psi)_I^\dagger f$;

where $A^\dagger$ denotes the pseudo-inverse $(A^*A)^{-1} A^*$.

Missing Frame. Yet the attentive reader would remark that

$$\langle \psi_i, f \rangle = \langle H.\psi_i, f \rangle, \qquad \text{while} \qquad \|H.\psi_i\|_1 \neq 1$$

Thus, the averaging to recognize appearance of the pattern ψ, performed by $f * \psi$, is lowered on the missing frame, as if a bias was introduce to implicitly consider them as 0. To remove this bias, one could compute instead

$$\frac{1}{\|H.\psi_i\|_1} \langle H.\psi_i, f \rangle \qquad \left(\text{given by} \quad \frac{f * \psi}{(H.1) * \psi}(x_i) \quad \text{if} \quad \psi_i(x) = \psi(x - x_i) \right)$$

On the other hand, there is a lot of freedom to extrapolate on the reconstruction

$$(H\Psi)_I^\dagger f = \arg\min_a \|f - H\Psi_I a\|_2^2$$

One could penalized this reconstruction, or introduce a bias on this step pushing it toward a affine regularization of f on missing frames $A(f, H)$ with $\Psi_I^\dagger A(f, H)$.

Scale-Space. Up to now the method privilege high frequency with strong response, while lower frequency, having a leaner one, aren't reconstruct while they play a big role in signal silhouette. Thus one could pick up at each step, one atom at each scale, rather than one overall. Figure 2.11 was obtained with such a procedure.

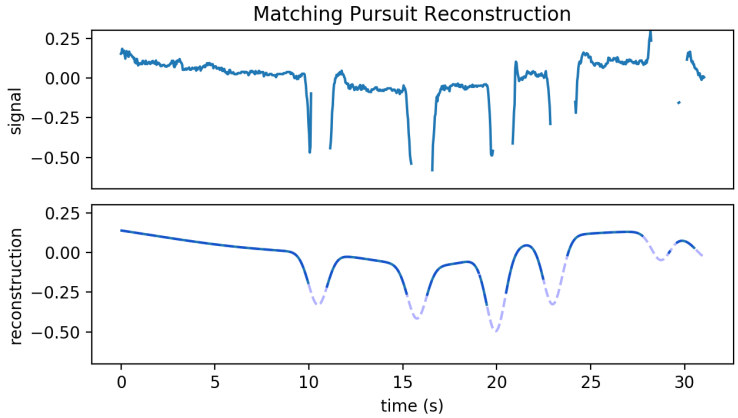

Figure 2.11: Reconstruction with a multi-scaled pursuit

This idea could be push forward. Looking at the response at each scale, rather than keeping only one atom, one could keep the local maxima falling above a threshold, which is a natural way to detect head turn. Moreover one could look at those maxima in space and in scale! This would help to capture the right scale of an event. That's how was obtained the best and more stable reconstructions.

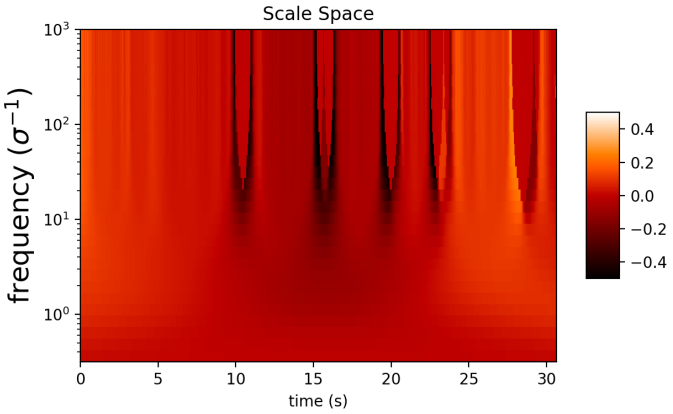

Figure 2.12: Scale-Space: responses to Gaussian of different scales vertically stack

Dictionary Learning

With the scale-space pursuit, one could achieve pretty good reconstruction with few atoms. Yet, the reconstruction really depends of the atoms of the dictionary, that were hand-designed. To design them, one should look at the signal, discover recurrent pattern, draw them manually before putting them in the dictionary. For our simple yaw signals, considering Gaussian was enough.

Variational Formulation. The simplest idea generalizes the problem (2.2.3) as

$$\text{minimize} \quad \sum_{i=1}^{n} \|f_i - H.\Psi a_i\|_2^2 + \lambda \|a_i\|_1 \tag{2.2.4}$$

where the signal $(f_i)_i$ where collected over the different subjects, and the minimization is done in $(a_i)_i$, to recover a sparse representation for each subject, and in Ψ, to find the best atom to do so.

To minimize this objective, it is natural to alternate the minimization in $(a_i)_i$ and Ψ. The minimization in Ψ could be done more precisely, cause costing less, than the other one. Moreover, one should notice that this objective correspond to the empirical risk of the corresponding

$$R(\Psi) = \mathbb{E}_f \left[\min_a \|f - H.\Psi a\|_2^2 + \lambda \|a\|_1 \right]$$

Thus when minimizing Ψ rather than considering all example at a high cost, one could approximate this objective with fewer examples, constituting a so-called batch. This remark is nothing but a justification for stochastic gradient descent alike methods.

Heuristic Methods. The convex relaxation method presents several drawbacks. Among others, we would like to keep the translation invariance of our dictionary. As shown Figure 2.12, building a Gaussian scale-space allow to discover events, giving their positions and scales. However, in previous method, the event were reconstructed with a Gaussian, which for example gave the pic attenuations on Figure 2.11.

Rather than using the detection filter, one could extract those events on signal, do clustering to recognize few events shapes, before taking those shapes as atoms to reconstruct the signal. Moreover, one could avoid reconstructing the signal, but simply describe the signal by those events, given their shapes (after clustering), positions, scales and amplitudes. Yet, our implementation doesn't improved the results, facing several issues to tackle in more details.

Pattern Recognition. Note that dictionary learning provide a generic framework that could be used on the entire signal, rather than channel by channel, to recognize special pattern corresponding to facial expression rather than muscular action. But this is slightly too ambitious regarding the need of soon results.

2.3 Dynamic Characterization

To perform further analysis, one should extract some descriptors on videos, potentially discriminating autism. It could focus on special event responses. It could also aim at finding chronicle behaviors like blinking more frequently. On an abstract level, a video is the realization of a random process. To characterize the processes, mathematics supply generic features such as variance, histograms or cross-correlation.

There is a trade-off between considering simple descriptors (*e.g.*, variance or histogram), and there link with quite strong and unjustified implicit hypothesis (*e.g.*, independence by channel or in time). The following section try to go from simple descriptors to more complex one able to better capture the structure of the data.

2.3.1 Distribution Descriptions

The simple hypothesis is to consider that values given by each channel (corresponding to each parameter of our facial parameterization) at each frame are independent. Then the description task consists in characterizing univariate distribution $\mathcal{L}$ from samples (x_i). Natural descriptors reads

$$\frac{1}{n}\sum_i \phi\left(x_i\right) \approx \mathbb{E}_{\mathcal{L}}\left[\phi\left(X\right)\right]$$

where ϕ could be used to estimate moment ($\phi(x) = x^d$) or density ($\phi = \mathbf{1}_{[a,b]}$).

Moment

If independence hypothesis is clearly too strong, moments can offer discriminative features, as variance could describe kid agitation.

One can derive little physical model to access, for example, energy displayed by the child during the video. When moving his head, eyebrows or so on, the kid have to apply a force F which isn't cost-free cause of friction inside the body. Let model those friction as fluid. Then Newtonian mechanics gives

$$F = m\frac{\mathrm{d}v}{\mathrm{d}t} - \mu v$$

Thus the power the kid has to use to move read

$$P = -F.v = -\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{1}{2}mv^2\right) + \mu v^2$$

And so the energy the kid had used during the video mainly reads

$$E \propto \sum_{i=1}^{n} v_i^2 = \sum_{i=1}^{n} \left(x_i - x_{i-1}\right)^2$$

This remark could lead to further development about modeling face muscular action from a physical perspective, probably to be incorporate in a learning classifier architecture, such as state-of-the-art neural networks.

Density

Moment estimation try to characterize distribution with parameters. Non parametric methods gives other tools to describe a law $\mathcal{L}$ from samples (x_i). A natural idea is to estimate the repartition function or an hypothetic density. Because of continuity, the density distribution should be smooth. Thus if x is observed, we can increase our estimation for $\hat{p}(x+\varepsilon)$. Let's consider a window function K (smooth, non-negative, concentrated in 0, of integral 1, *e.g.*, a gaussian), and a scaling parameter h. A natural estimation based on the observations $(x_i)_i$ is thus given by

$$\hat{p}(x) = \frac{1}{hn}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)$$

For Gaussian estimation, some work has been done, taking as a rule-of-thumb when taking the gaussian window to scale it with:

$$h \approx \hat{\sigma} n^{-\frac{1}{5}}$$

Histogram method. If autistic kids presents strange motor behavior that could probably be capture by the frequency of muscular contraction at a certain intensity, the density describe in too much detail the different level of intensity. One should rather gather the intensity values in groups, corresponding to a muscle really contracted, a muscle slightly contracted and so on.

To design the histogram boxes, a nice idea is to compute p-values over the entire set of videos, so one can expect on average values to be uniformly distributed among boxes, and thus boxes to have to most discriminative power between distributions.

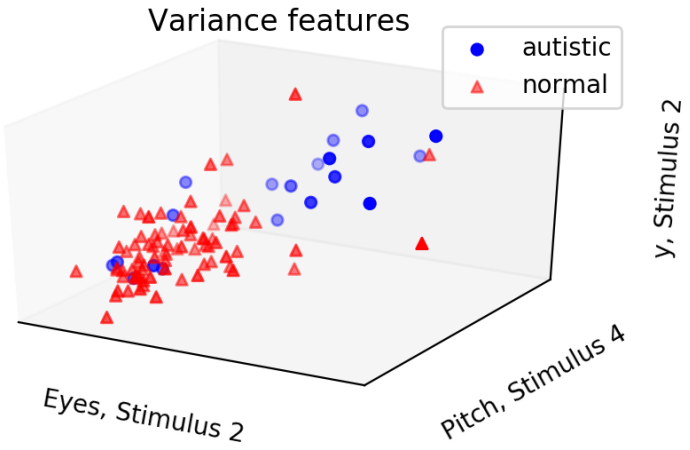

Figure 2.13: Variances allow to discriminate subjects

2.3.2 Channel Correlation

The same settings cast to get correlation between channels with cross-moment and joint-distributon, as the discriminative information is supposed to lay in correlation between the facial features in space and in time. Nevertheless, as the number of dimension increased the number of points needed to span well the space increase exponentially, which is one of the central reason of the so-called curse of dimensionality.

Cross Moment

The easiest description of correlation between features would be the correlation matrix, where the entry (i, j) is the empirical correlation between the features x_i, and x_j reading

$$\sum_t \left(\frac{x_{i,t} - \hat{\mu}_i}{\hat{\sigma}_i} \right) \left(\frac{x_{j,t} - \hat{\mu}_j}{\hat{\sigma}_j} \right)$$

where $\hat{\mu}_i$ and $\hat{\sigma}_i^2$ are the empirical mean and variance of feature i.

Scaled Correlation. To get more precise it is important to differentiate correlation due to slow motion and instantaneous response. In other terms, to get a value for the correlation between yaw and roll on a long term perspective, and an other for this correlation for instantaneous response. this could be obtained by retaking the scale-space description built before.

Figure 2.14 shows discriminative correlation. On the left is normal correlations, and on the right, scaled ones. On one hand, scaling increase the number of correlation pairs, making it easier to obtain spurious findings, yet on the other hand, it seems scaling allow to focus on event response and therefor better cluster reaction

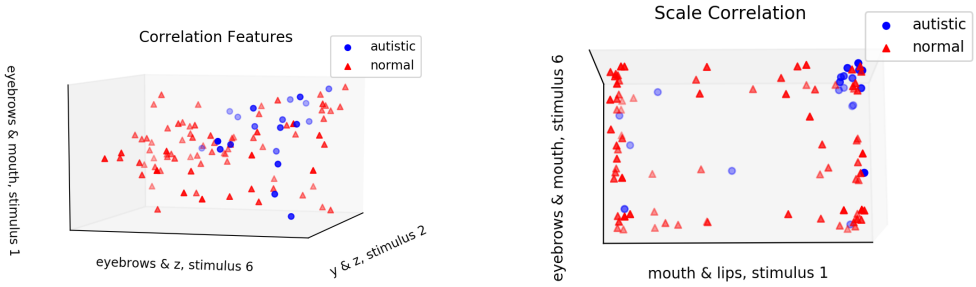

Figure 2.14: Scaling Correlation seems meaningful

Density

More complex descriptions of the correlation could be found in joint density estimation, for example with a window method

$$\hat{p}(\mathbf{x}) = \frac{1}{h^d n} \sum_{i=1}^{n} K\left(\frac{\|\mathbf{x} - \mathbf{x}_i\|_\Lambda}{h}\right)$$

Where h is a scaling parameter, and Λ is a distortion matrix which preserve volumes ($|\det(\Lambda)| = 1$, the volume deformation being encode by h), deforming the metric as

$$\|\mathbf{x}\|_\Lambda = \|\Lambda \mathbf{x}\|_2$$

Let's once again consider a Gaussian estimator, which sum to one when integrating over the whole space. The estimator reads

$$\hat{p}(\mathbf{x}) = \frac{1}{(2\pi)^{d/2} h^d n} \sum_{i=1}^{n} \exp\left(-\frac{1}{2h^2} (\mathbf{x} - \mathbf{x}_i)^T \Sigma^{-1} (\mathbf{x} - \mathbf{x}_i)\right)$$

where we have parameterized with $\Sigma^{-1} = \Lambda^T \Lambda$. Several choice are then possible for each parameter, a simple would be to make our estimation close to a point having the same form has the global cloud of point, with a rule-of-the-thumb like

$$\Sigma \propto \hat{\Sigma}, \qquad h = \left|\det\left(\hat{\Sigma}\right)\right| n^{-\frac{1}{4+d}}$$

A smart observation consists in remarking that one can map one's data into the proper axis of Σ, before having independent components, and being able to then applied a one dimensional estimation. Let's diagonalize $\Sigma = PDP^T$, then

$$K\left(\frac{\|\mathbf{x} - \mathbf{x}_i\|_\Lambda}{h}\right) = \prod_{j=1}^{d} K_{1\mathrm{D}}\left(\frac{\langle p_j, \mathbf{x} - \mathbf{x}_i \rangle}{h d_j^{1/2}}\right)$$

And the rule-of-the-thumb reads

$$h d_j^{1/2} = n^{-\frac{1}{d+4}} \sqrt{\mathrm{var}_i \langle p_j, \mathbf{x}_i \rangle} = n^{-\frac{1}{d+4}} \sqrt{p_j^T \hat{\Sigma}\, p_j}$$

where $(d_j)_j$ are constraint according to $\prod d_j = 1$.

Backwards, one could define the axis one want to be independent, based on prior knowledge, and do the 1D estimation on each of this axis, before going back to the full-dimensional space. From here, one can notice that the choice of $\Sigma \propto \hat{\Sigma}$, was the same as choosing $(\langle p_j, \mathbf{x} \rangle)_j$ independent, where p_j correspond to the singular vector of the design matrix X, since writting $X = UDV^T$ gives $\hat{\Sigma} \propto UD^2U^T$, thus $p_j = u_j$.

Moreover, one is not supposed to stick with Gaussian kernel, and could use different kernels for different axis. In our case, is the yaw is probably distributed according to a Gaussian, the mouth opening would rather followed a exponential law, probably adapted to other kernels, and rule-of-the-thumb.

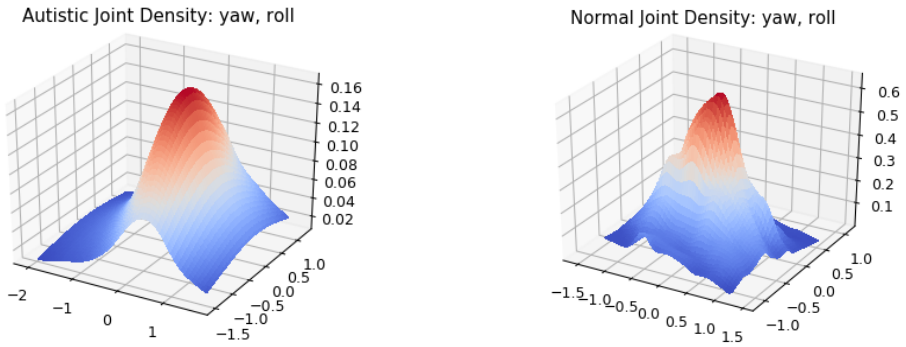

Figure 2.15: Joint density between yaw and roll computed over the entire sets

So was obtained Figure 2.15, which shows that normal people tends to have more correlation in yaw and roll, which is probably due to the fact that they mainly turn to look at their mother on who they are sitting, while autistic turns randomly. In fact, psychologist are expecting those moves for not autistic child as social referencing, which is when the kid encounter something new, looking at how other people are reacting to learn how to feel or react to this new action. However, modeling both classes by a simple distribution seems inappropriate, a mixture model would be better, allowing to limit the impact of a subject particularity in the overall estimation.

Joint Histogram. Once again, the density estimation seems to be a bit too constrictive and an histogram descriptor could more adapted and better suited to describe each processes before going more in depth into classification method. Yet, some features are expected to be independent (*e.g.* roll and eyes opening) thus, one derive only joint histogram on certain sets of features, where correlation between features is supposed to show discriminative information. This allow to faster computation and tackle the curse of dimension.

2.3.3 Time Correlation

To understand muscular action at a given frame, one should not only describe the position, but also if the subject is contracting or dilating this muscle, in other terms, one should access higher-order information such as the speed or acceleration. To describe those quantity, accessing increments seems useful. In fact, several stochastic process are well described from their increment, such as Brownian motion or more generally Levy processes.

Increment can also be seen through convolution, it consists in convolving the signal with filters of the type `w = [-1, 0, ..., 0, 1]`. This could be done in a smoother way considering Gaussian derivatives. Considering different scales can once again orient us toward scale-space representation, which isn't so different from working in the spectral domain in harmonic analysis.

First Layer. Gathering the need for time correlation, with what has been said about signal processing in section 2.2, we introduce a first convolutional layer in the analysis of the signal x, it is made of two types of filters, Gaussian ones, corresponding to smoothing, or event recognition, and derivative of Gaussian, corresponding to increment, higher-order information. This structure isn't without similarity with the wavelet filter h and g for smoothing and details. Yet, those filter are directly reproduce at different scale, without cascading to avoid loosing precise event localization. Let denote $W_1 x$ this first convolutional layer.

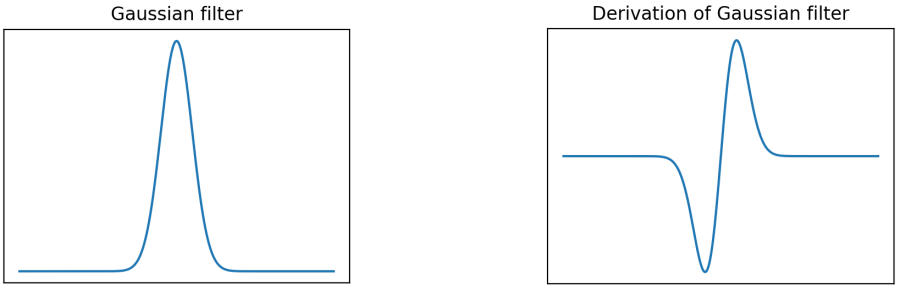

Figure 2.16: Filters of a first convolution layer

2.3.4 Non Stationarity

In the precedent section, we suggest to average signal into variance, mean or histogram descriptors that tend to kill the spacial information, even if eventually keeping the scaling one. Yet this information is primordial to capture response given a specific event. Thus one can consider variance or cumulative distance compute locally as

$$var = x^2 * w - (x * w)^2, \qquad cumul = |x| * w$$

where w is a local averaging filter to represent a local summation as

$$w[k] = \frac{1}{n} \mathbf{1}_{k \leq n}$$

The same localization could be done on histogram descriptors. For that let's notice that histogram consists in clustering the values given by a features (*e.g.* yaw, eyes opening) at each frame, before counting how many of those values fall in each clusters. As already said, clustering consists in labelling the intensity of a muscle contraction, thus consider this clustered representation Tx seems smart, as long as we keep speed information. From this clustered representation, one can then compute local histogram at a more or less broad scale. Moreover, if one represent the clustering as one-hot, the local histogram can be seen as a convolution. In fact it will be nothing but:

$$histo = \left(\mathbf{1}_{a_i < x < a_{i+1}} * w\right)_i$$

where the clustering has been defined by the box $([a_i, a_{i+1}])_i$. Thus histogram can be seen as really similar to a neural network layer, with a non-linearity $\text{sign}(x)$.

Second Layer. So two different way can be thought to build on top of the first layer $W_1 x$. The way is the histogram, corresponding to density description or intensity clustering, It consists in a first thresholding

$$Tx = \left(\mathbf{1}_{a_{i,j}<x_j<a_{i+1,j}}\right)_{i,j}$$

where j correspond to the different feature, and i the different histogram box. Once again, a_i are computed as p-values over the whole dataset. Then some smoothing $W_2 x$ is thought to obtained localized histogram from this one-hot representation, eventually smoothing filter can apply jointly on different channel to get joint-histogram.

The second way correspond rather to moment method or describing energy display by the kid. The smoothing operator doesn't change, yet the non-linearity will correspond to

$$Tx = \left[x^2, |x|, x\right]$$

where all those operations are performed component-wise. To compute the variance rather than the second moment, one should once again correlate the channel in the convolution layer $W_2 x$.

So we end up with a new representation of the type $W_2.T.W_1 x$. The good part of this representation is that we clearly understand what all the coordinates correspond to. Yet, it has really grown in dimension, and more likely to conduct us toward spurious findings. Furthermore, the size of the description slow down further computation, like regression or diagnosis correlation analysis.

Features Reduction

To reduce the dimension of our descriptors, several solution could be thought. First remove features by hand: some that seems pointless, or redundant regarding other features. A second option would be to detect automatically features that are invariant regarding the diagnosis and removing them. A fancy idea could be to pick randomly, several time, few features, and try to obtain the diagnosis with simple classifiers, before inferring which features might be discriminative. Similarly algorithm could analyze redundancy between features.

But, before going in this direction, we already have a lot of redundancy in time. Up to now we are keeping all localized histogram center in any possible position. Yet one could keep only few position with a subsampling operator, or to avoid loosing a pics in the neighborhood cause of this subsampling, keep the highest response on a local window with a pooling operator. At the end of the day, this is nothing but a second non-linearity T_2.

2.4 Pattern Recognition

Based on psychologist work for behavioral analysis, one could forget random process consideration to draw a simple pipeline for our classification task. First, from the face parameterization recognize muscular action, or action unit. Then, correlate those units between them and in time to understand face expression. Finally, linking face expression with kid behaviors, before analyzing those behaviors to screen for autism. When performing this last step, one should dissociate recurrent behaviors from behaviors triggered by events on the videos that the kid are watching.

There is several way to follow this pipeline. One can rather do the learning at once, to get better result by avoiding introducing bias in the classification through intermediate step. Or dissociate the different task, in order to better understand the classifier learnt, and avoid overfitting during the learning step, since using an other objective than the scoring function. On the machine learning side, recognize muscular action and then face expression can be seen through pattern recognition, framework that have already been mentioned. It could be done through filtering, orienting the implementation toward convolutional neural networks, or clustering, orienting it toward words learning. There is naturally two layers, one to recognize muscular action independently, one to correlate them into face expression. Then will follow a statistical review of face expression to discover behaviors, before putting this in a classifier discerning autism from normal kids.

Yet, those ideas would need more supervision or a larger dataset to be perform without having to be scared of spurious findings due to overfitting. And if the dataset is well scaled for testing statistical alternatives, it is not suited for ambitious learning. Nevertheless, let's draw some idea and mention some interesting points for a potential future work. By the way, supervised Databased have been collected for algorithms to learn detecting current emotion from facial information [20].

2.4.1 Convolutional Neural Network

We have shown how prior architecture could be cast under a convolutional network. Yet all the filter were designed by hand, often with a lot of redundancy to make sure not to pass beside a potential perfectly discriminative feature. On the opposite, a convolutional neural network could improve result while keeping a decent size by automatically fitting filters.

One should design a network architecture that can be learnt based on an objective built on information we get on the kid which only the autism diagnosis without psychological profile. Moreover this architecture should allow to understand why the network is classifying a kid as autistic or not. Figure 2.17 suggests a subjective architecture to do so. This architecture would probably have completely modified if we were to try it on for real after adjustment to both achieve understandable parameter and obtain a working classifier.

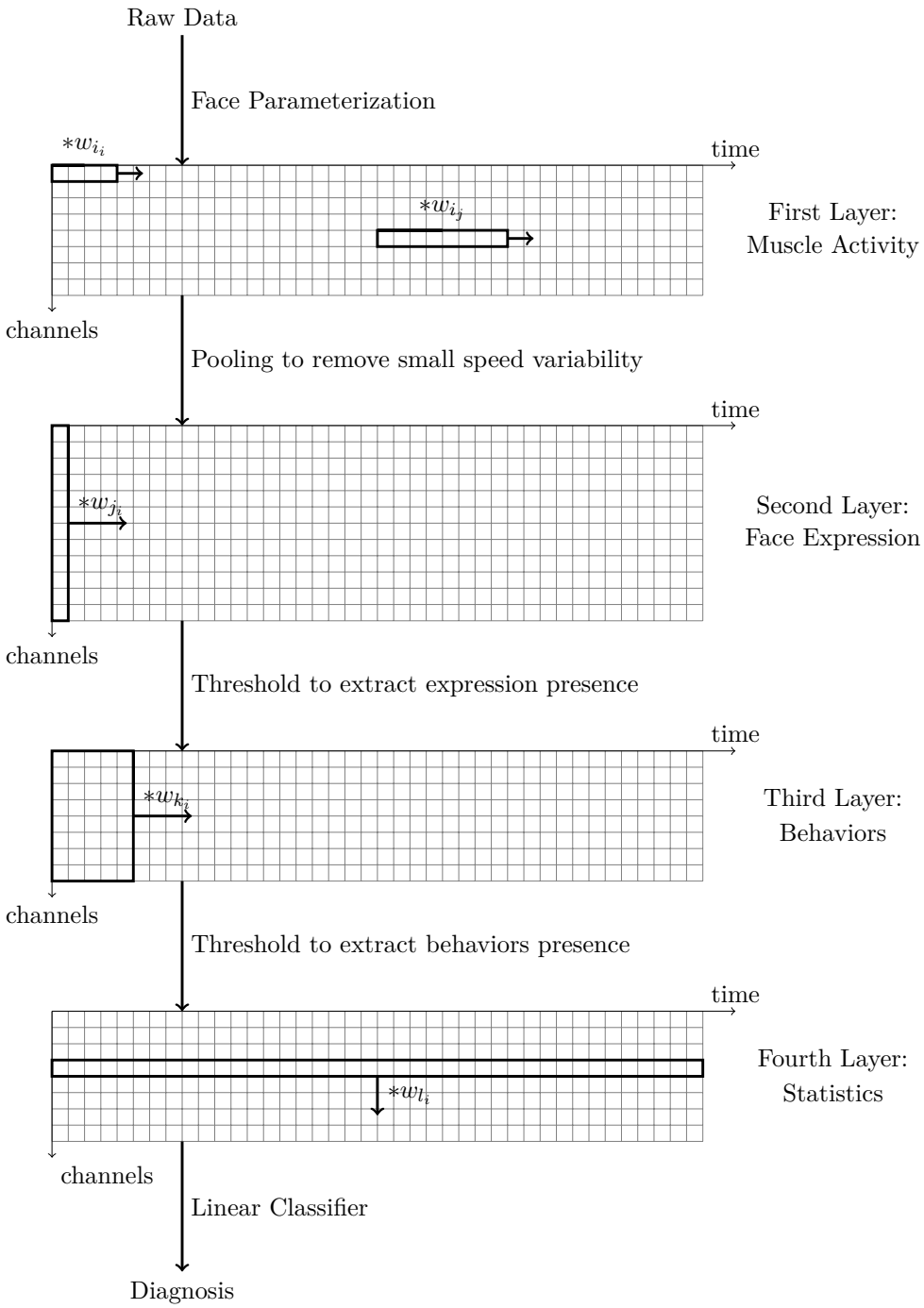

Figure 2.17: A Convolution Network Architecture

Let's explain Figure 2.17. The network take as input our parameterization where each channel correspond to different muscle or head pose parameter. The first layer is supposed to learn special pattern to analyze each muscle activity, in other term, recognize typical muscular action. The second layer build on this, eventually adding a threshold step to extract muscular activity presence and a pooling one to remove little temporal variability due to small speed difference in face expression. This layer is supposed to learn how to correlate muscular activity to recognize facial expression. Once again a thresholding step is needed to extract what expression the face is showing, this threshold could be realize with a local parameter to make sure extracting at most one expression at each frame. A third layer then correlate in time the presence of those expressions (*e.g.* happiness followed by sudden fear) to understand behaviors. A fourth one is supposed to perform statistics on those behaviors, eventually focusing on a specific moment or averaging over the entire video, before feeding a linear classifier.

This learning is ambitious and more likely not to produce the result we are expecting, without a huge database and a smart and subtile learning control. Thus one could try to force the filters to take the structure we want. It would be an interesting development for neural network to be able to incorporate prior information. For example, one could dissociate the learning layer by layer, more or less supervising the learning for each layer. For example, for layer 1, one can just fit a parametric physical model of muscular activity. For layer 2, one can use face expression database and supervision to label the output. For layer 3, one can add sparsity constraints. And for layer 4, one can fit parametric statistical descriptors.

2.4.2 Clustering Network

As we have seen, we would like the network to act like indicating words corresponding to muscular action, expression or behaviors on the video frames. Thus, word clustering and dictionary learning seems well suited. It consists in clustering patterns supposed interested on raw data, before clustering those patterns into words and reporting those words on the data. It would be done hierarchically, respecting Figure 2.17 layers.

There is two main questions to answer. First how to decide that a piece of the data is an interesting pattern that should become a word. Secondly, how to group pattern together to learn specific words, or which metric to use for the clustering. To answer the first question, one can expect certain pattern to appear and look for them, we have also seen in past the benefice of looking at maxima in scale-space. Extending this idea, interesting pattern can be seen as concentrating energy at a certain scale and a certain time. One could also look for inflection points, or zero-crossing of the speed suggesting the beginning of a new motion. Otherwise one could introduce a variational frameworks, based on dictionary learning. This would also answer the second question. Otherwise, for hand made pattern extraction, one then have to introduce a metric to compare them and perform clustering. This metric should take in account time deformation link to speed variability. Thus transport metric seems convenient. Once again, this is an interesting topic where profound ideas could be expressed and subtile math developed, but this is slightly outside of the autism screening task.

Chapter 3

Statistical Learning

Under the term *Stastical Learning* lies several issues to be tackle and ideas to be investigated. In the precedent chapter was obtained new representations of the subjects, ideally we would like the machine to discover discriminative features to screen for autism. Yet, regarding the small size of the database, one's expectations should be reasonable, especially in term of pattern recognition. Thus this present chapter shift from pure blind machine learning with statistical guarantees to educated method with psychologist warrant, based on the idea of normal and abnormal response.

3.1 Machine Learning

On the long run, we would like to replace psychological warrants by statistical ones, and human research by computer exhaustive one. This orients us towards pure machine learning to derive a screening method. It consists in fitting parameter in a generic model to discriminate subjects on a training set. To make sure our method isn't based on spurious correlation, a validation stage is needed testing the derived method on a testing set For algorithmic matters, it is convenient to adopt a input-output framework Let's denote the signal as an input variable X, and the knowledge that we want to extract on it as an output variable y. In our case, X correspond to medical data collected on a patient, y to the patient's diagnosis.

3.1.1 Classifier

Methods could be split in two types: generative and discriminative. Generative model consists in trying to recover the underlying data distribution, before maximizing statistics test property. But they present several limitations, mainly due to the difficulty to define a adequate probabilistic model. And, up to the author understanding, popular *Graphical Models* are more used for the complexity they can capture for cheap computation, rather than the probabilistic framework they offer. Nowadays, discriminative model rules over the state-of-the-art from generalized linear model, such as kernel

SVM, to the more complex Neural Networks. Because of the dataset size, we focus in the following on medium complex classification procedure.

Neighborhood

Let's begin with the simple example of Figure 3.1: how to design a method to redraw the red point in a yellow triangle or a blue disk? The natural answer is that it should be yellow, since it is surrounded by yellow points. Let's try to define more precise rules before implementing them in an algorithm.

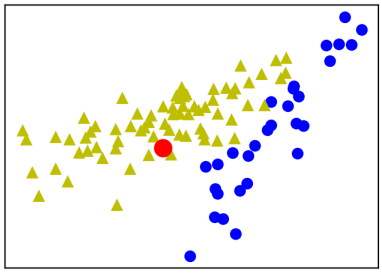

Figure 3.1: A Toy Problem

As we have just said it seems natural to look at the point surroundings. The Nearest Neighbors rules comes with a simple rule, look at the closest point, and assign the unknown point to the same set as this closest point. Eventually, one could look at the k closest points before taking a decision, or even define a certain distance to have the good closest points. In fact, when looking at this example, it seems natural to look at the distance between the red point and the yellow set according to the shape of the yellow cloud of points, and similarly for the blue set. This could be a first step toward generative modeling, here it would be Quadratic Discriminant Analysis.

Metric Learning. In real problems, the ℓ^2 distance is often deteriorated by a lot of meaningless coordinates. A solution to counter this problems would be to re-weight the different coordinates. Then a pseudo-distance would read:

$$d(x,y) = \|\langle x, w\rangle - \langle y, w\rangle\| = \sqrt{(x-y)^T ww^T (x-y)} = \|x-y\|_{ww^T}$$

One could extend the search for a new metric under $\|x-y\|_S$, where S is a semi-definite positive matrix. To transform our data in order to separate classes, a idea consists in minimizing the variability inside a class, while maximizing the ones inter-classes. Several variational formulation could be thought for such a task, [26] suggests

$$\min \sum_{i\sim j} d_S(x_i,x_j)^2 + \lambda \sum_{i\sim j} \sum_{y_l \neq y_i} \max\left\{0, 1 + d_S(x_i,x_j)^2 - d_S(x_i,x_l)^2\right\}$$

Where $i \sim j$ index the clusters to form, *e.g.* the different form of autism – $i \sim j$ imply $y_i = y_j$ but not necessarily the opposite, since one class could correspond to several

clusters. The left term minimize the distance inside clusters, while the right one, force the distance between cluster to present a margin, with the hinge loss capturing the margin idea. Introducing usual slack variables, and expressing d_S this leads to

$$\begin{array}{ll} \text{minimize} & \min \sum_{i\sim j} (x_i - x_j)^T S (x_i - x_j) + \lambda \sum_{i\sim j} \sum_{y_l \neq y_i} \xi_{ijl} \\ \text{subject to} & \xi_{ijl} \geq 1 + (x_i - x_j)^T S (x_i - x_j) - (x_i - x_l)^T S (x_i - x_l) \\ & S \succeq 0, \qquad \xi_{ijl} \geq 0 \end{array}$$

Which is nothing but a semi-definite program. However, solving the problem structure allow to better solve it if implementing the solver ourself. Let's follow [26] work and rewrite our problem under matrix form with $X_{ij} = (x_i - x_j)(x_i - x_j)^T$ as

$$\begin{array}{ll} \text{minimize} & E(S) = \min \left\langle S, \sum_{i\sim j} X_{ij} \right\rangle + \lambda \sum_{i\sim j} \sum_{y_l \neq y_i} \max \{0, 1 + \langle S, X_{ij} - X_{il} \rangle\} \\ \text{subject to} & S \succeq 0 \end{array}$$

Let's differentiate in order to implement a gradient descent as

$$\nabla_S E = \sum_{i\sim j} X_{ij} + \lambda \sum_{(i,j,l)\in\mathcal{A}} X_{ij} - X_{il}$$

Where $\mathcal{A}$ denotes the triplets that don't respect the margin condition. The gradient descent can be made smart remarking the gradient only change with $\mathcal{A}$ which shouldn't change too fast. This implementation, if smartly done, is incomparably faster.

We could spend way more time on this metric learning method to solve several question left under the carpet, such as when to stop the learning to avoid overfitting, or simple normalization and ponderation consideration. A key remark is that one could use the objective function to classify a new example rather than using the nearest neighbors algorithm. In practice, this objective tend to flatten the data on a line as does linear classification.

Euclidean Information Summary. The prior work have only used the vectorial structure of the data, thus one can reduce the dimensionality of the data by projecting it on $\mathcal{S} = \text{span}\{x_i\}_i$, where $(x_i)_i$ is the data description. Indeed, one only need to access $K(x_i, x_j) = \langle x_i, x_j \rangle$. *Reproducible Kernel* [2] have been really popular to implicitly cast the data in a new *Hilbert Space*, without describing each coordinates but only the scalar product. But will this be a limitation to our metric learning idea? The answer is no, and the solution simply consists in finding an orthogonal basis $(f_i)_i$ of $\mathcal{S}$. It could be easily perform with Gram-Schmidt process. One can then work in this new basis. This method allow to easily reduce the dimension of the data space to the number of example, while keeping the Euclidean structure.

Separation

A simpler idea to tackle the classification task is to drop the neighborhood paradigm, and try to cut the space in two with one set on one side, and the other on the other side.

Several algorithm have been proposed, their different performance have showcased SVM as a good one, easily computable. It consists optimizing the classification margin, with relaxation for points that can't be put on the good side of the cut. Let's consider the classification function $f(x)$, the objective to minimize reads

$$\min_{f\in\mathcal{S}} \frac{1}{n}\sum_{i=1}^{n} l\left(y_i, f\left(x_i\right)\right) + \lambda J\left(f\right)$$

A linear classifier reads $f(x) = \langle w, x\rangle$, usually with the penalization $J\left(f\right) = \|w\|$. The value of w on $\mathcal{S}^{\perp}$ doesn't change the classifier, but only increase the norm of f, thus one could seek f under the form

$$f(x) = \left\langle \sum_{i=1}^{n} \alpha_i x_i, x \right\rangle = \sum_{i=1}^{n} \alpha_i K(x_i, x)$$

Two basic loss function are considered for the SVM

$$l\left(u, v\right) = \max\left\{1 - uv, 0\right\}, \qquad l\left(u, v\right) = \max\left\{1 - uv, 0\right\}^2$$

Given the so-called L1 and L2-SVM. The loss is refer as the Hinge loss. For the L1, this could be transcript in the following quadratic problem:

$$\begin{array}{ll} \text{minimize} & \frac{1}{n}\sum_{i=1}^{n} \xi_i + \lambda \left\|f\right\|_S^2 \\ \text{subject to} & \xi_i \geq 0, \qquad \xi_i \geq 1 - y_i f(x_i) \end{array}$$

Considering f, we can cast the problem in $\mathbb{R}^n$ as

$$\begin{array}{ll} \text{minimize} & \frac{1}{n}\mathbf{1}^T\xi + \lambda\alpha^T\mathbb{K}\alpha \\ \text{subject to} & \xi \geq 0, \qquad \xi \geq \mathbf{1} - \operatorname{diag}(y)\mathbb{K}\alpha \end{array}$$

where α is the vector of coordinate $(\alpha_i)_i$ and $\mathbb{K}$ is the Gram matrix $\left(K(x_i, x_j)\right)_{i,j}$ and the inequality constraint should be understood component-wise. Solving this QP problem is faster through its dual, few derivations are sufficient to find it as

$$\begin{array}{ll} \text{maximize} & \mathbf{1}^T\mu - \mu^T \operatorname{diag}(y)\mathbb{K}\operatorname{diag}(y)\mu \\ \text{subject to} & \frac{1}{n}\mathbf{1} \geq \mu \geq 0 \end{array}$$

The dual and the primal variables are linked as $2\lambda\alpha = \operatorname{diag}(y)\mu$, thus the dual reads

$$\begin{array}{ll} \text{maximize} & 2\,\mathbf{y}^T\alpha - \alpha^T\mathbb{K}\alpha \\ \text{subject to} & \frac{1}{2n\lambda}\mathbf{1} \geq \operatorname{diag}(y)\alpha \geq 0 \end{array}$$

For the L2-SVM, the same derivations lead to the optimization problem

$$\begin{array}{ll} \text{maximize} & 2\,\mathbf{y}^T\alpha - \alpha^T\left(\mathbb{K} + n\lambda\mathrm{I}\right)\alpha \\ \text{subject to} & \operatorname{diag}(y)\alpha \geq 0 \end{array}$$

Unbalanced Dataset. If what we have exposed before is classical knowledge about SVM, we should go more inside to really use the SVM in our case. The main problem is that our data are unbalanced. If one want to minimize the classification error as

$$\sum_{i=1}^{n} \mathbf{1}_{\operatorname{sign}(f(x_i))\neq y_i}$$

It could easily reach 25% by classifying each example as not autistic. Since we really care about autistic classification, it seems better to weight this score as

$$\frac{1}{\#\{i \,|\, y_i = 1\}} \sum_{y_i=1} \mathbf{1}_{\operatorname{sign}(f(x_i))\neq y_i} + \frac{1}{\#\{i \,|\, y_i = -1\}} \sum_{y_i=-1} \mathbf{1}_{\operatorname{sign}(f(x_i))\neq y_i}$$

This to avoid the unbalanced dataset pushing the classifier to classify for the more frequent class to easily. In the SVM formulation, the term $\mathbf{1}^T\xi$ correspond to the accuracy score. Thus we should weight it as

$$w^T\xi$$

where $w = (w_{y_i})_i$ and

$$w_y \propto \frac{1}{\#\{i \,|\, y_i \neq y\}}$$

We can now retake the usual calculation to arrive under the dual formulation. The primal problem reads

$$\begin{array}{ll} \text{minimize} & \frac{1}{\|w\|_1} w^T\xi + \lambda\alpha^T\mathbb{K}\alpha \\ \text{subject to} & \xi \geq 0, \qquad \xi \geq \mathbf{1} - \operatorname{diag}(y)\mathbb{K}\alpha \end{array}$$

Thus the Lagrangian

$$\begin{aligned} \mathcal{L}(\alpha, \xi, \mu, \nu) &= \frac{1}{\|w\|_1} w^T\xi + \lambda\alpha^T\mathbb{K}\alpha - \nu^T\xi - \mu^T(\xi + \operatorname{diag}(y)\mathbb{K}\alpha - \mathbf{1}) \\ &= \left(\frac{1}{\|w\|_1} w - \nu - \mu\right)^T \xi + \lambda\alpha^T\mathbb{K}\alpha - \mu^T \operatorname{diag}(y)\mathbb{K}\alpha + \mu^T\mathbf{1} \end{aligned}$$

The optimization in ξ gives the dual constraint $\nu + \mu = \frac{1}{\|w\|_1} w$, and since ν is unconstraint otherwise, it gives $\mu \leq \frac{1}{\|w\|_1} w$. Everything else stay unchanged, thus we now get the SVM formulation as

$$\begin{array}{ll} \text{maximize} & 2\,\mathbf{y}^T\alpha - \alpha^T\mathbb{K}\alpha \\ \text{subject to} & \frac{1}{2\|w\|_1\lambda} w \geq \operatorname{diag}(y)\alpha \geq 0 \end{array}$$

Implementation. There is two choices to solve this SVM variational formulation, the first one consists in using classical QP solvers that one can find in libraries such as `CVX_opt`. The other one is to solve it ourself, [17] suggests that using a coordinate ascent method would be the best way to do it, especially with a lot of data, which can be understand as the same way as stochastic gradient descent since each entry of α correspond to a data x_i. However, since our dataset is relatively small, and we don't care so much about computation time, implementing ourself a coordinate ascent doesn't seem necessary, even if it could help for some instability features of QP solvers.

3.1.2 Scoring

Once we have design methods, we would like to compare them to choose the best one. For that, we need a way to score methods. Of course, a good method is a method that succeed to screen for autism. Yet, since we have a restrictive number of examples, one shouldn't overdraw conclusions. For this purpose, probabilistic frameworks gives guarantees about ones findings and testings.

Supervised Learning

Let's put explicitly what laid implicitly in prior developments. We have a input-output framework, where some data X are linked to a diagnosis y. We would like to design a method, read a algorithm, which map the data into the diagnosis. Let's arbitrarily index a method by λ, and denote as $\mathcal{A}(\lambda, X)$ the diagnosis delivered by this method on a data X. Ideally this diagnosis will correspond to the right one. In other term we would like our method to realize the matching $y = \mathcal{A}(\lambda, X)$.

To find the best method, one could collect labelled data $(X_i, y_i)_i$ and see which method succeed to match $y_i = \mathcal{A}(\lambda, X_i)$ for each i. To enable method generalization to unseen data, the dataset is usually split in a training and a testing dataset, the first to fit a model, the second to test how well it perform on new data. This to avoid overfitting.

To rank more precisely the different method on test set and fit the method model, it is convenient to introduce a variational framework, allowing, among other to use optimization tools. This could be done naturally by introducing a measure of dissatisfaction, or a cost l to have predict $y_p = \mathcal{A}(\lambda, X)$ rather than y on a given example X. We can then gather all those individual loss to define a global measure of dissatisfaction L after having seen and predict a collection of patient (X_i).

Let's go concrete with the example of linear regression, $\lambda = (t, w)$ where $t =$ `linear regression` design the method type, and w is an internal parameter for the linear regression. With this method, $\mathcal{A}(\lambda, X) = f(w, X) = \langle w, X \rangle$. The least square regression consists in averaging ℓ^2 individual loss with

$$L(\lambda) = \frac{1}{n} \sum_{i=1}^{n} l(y, y_p) = \frac{1}{n} \sum_{i=1}^{n} \|y_i - f(w, X_i)\|^2$$

Statistical Guarantee

To get better guarantee on what we are doing and the accuracy of method testing, statistics provide a useful framework. To do so, let's introduce a new variable T, which correspond to the omniscient explanation of the data observed through the representation X, which could correspond to the abstract scene behind a table of pixel, or the abstract meaning behind a text that we seek to translate. According to the kind of data one is recording, one can design a law on (T). From this omniscient explanation, it would be easy to extract the desired knowledge $(y|\, T)$ and to derive the different representations $(X|\, T)$.

Once we have a law on (X, y), we can define a global score of dissatisfaction, that we want our method to minimize. Retaking the last example it could be

$$L\left(\lambda, (X, y)\right) = \mathbb{E}\left[l\left(y, f\left(w, X\right)\right)\right]$$

Usually, we won't try to specify the law on (X, y), otherwise it becomes a purely statistical problem, but just assume that we have sampled $D_n = (X_i, y_i)_i$ independently according to this law. We could then approximate L by an empirical measure such as

$$\hat{L}\left(\lambda, D_n\right) = \frac{1}{n}\sum_{i=1}^{n} l\left(y_i, f\left(w, X_i\right)\right)$$

To obtain precise guarantees, a first issue is how well $\hat{L}$ approach L. First answers are given by asymptotical statistics. For example if we retake the example for averaging individual losses, the large number theorem gives the convergence as the number of example grow toward infinity. With an idea for the variance of the process, the central limit theorem allow to be more precise regarding the convergence speed.

We are yet facing two problems. First we don't really know the law from which the data are sampled. Then we don't necessarily have enough data to rely on asymptotical statistics. To get non-asymptotic bound, one can rely on concentration inequality, such as Hoeffding's or Mc Diarmid's inequality. The more we know about the distribution and the more example we have, the better those bound are going to be. Similarly the more supervised and the more data we have, the better the learning will be.

The second question would be about using the scoring to select method, the main concerned is to avoid overfitting. Actually, if one try to rank a lot of weak methods with a certain dataset, as the number of ranked methods increase, the probability to have one of these method performing well on the dataset cause by pure coincidence increase. Usually we split the dataset in training and testing, and somehow fit model on training for few different methods, and score the best model of the few different methods on the dataset. Yet, could we have more precise guarantee about this process and how many method we could rank on the testing test before risking overfitting?

On Binary Classification

Let's cast our talk closer to the initial problem, where we can code the output with $y \in \{-1, 1\}$, describing if the kid present a symptom or not, is autistic or not, *etc.*

Performance Evaluation. A natural scoring function is the number of error our prediction made. Assuming $f(w, X) \in \{-1, 1\}$ it would be

$$L(\lambda, (X, y)) = \mathbb{E}\left[\mathbf{1}_{\{y \neq f(w,X)\}}\right] = \mathbb{P}\left(y \neq f(w, X)\right)$$

Two types of error are being made, predicting 1 when the answer should be -1, known as fake positive, and the opposite, known as fake negative. One type of error could be way more annoying than the other. In our case, missing a autistic case is way more important than warning for autism when there is not.

Several performance measure have been thought for binary classification. Let denote by *tp*, *fp*, *tn* and *fn* respectively the number of true positive, fake positive, true negative and fake negative. Main measures include the precision, the recall and the f_β score, which read

$$P = \frac{tp}{tp + fp}, \qquad R = \frac{tp}{tp + fn}, \qquad f_\beta = \left(\beta^2 + 1\right) \frac{PR}{\beta^2 P + R}$$

In other term, P measures the ratio of examples well classified as positive over the total number of example classified as positive. R measures the ratio of examples well classified as positive over the total number of positive example. And f is a weighted harmonic average of both. Note that this could also be done for the negative class.

Soft Classifier. Several classifiers factorize as $\operatorname{sign}\left(g\left(w, X\right)\right)$. This is useful to design an index of prediction confidence depending on $|g\left(w, X\right)|$. This index has an accurate explanation when we have design a generative model $\mathcal{M}$ for $(y|\, X)$ and set

$$g(w, X) = 2\mathbb{P}_{\mathcal{M}}\left(y = 1|\, X\right) - 1$$

In this general setting, g is called a soft classifier. Way more scoring can then be used, first the general dissatisfaction index would write as

$$L(\lambda, (X, y)) = \mathbb{E}\left[\varphi\left(y, g(w, X)\right)\right]$$

Depending on the calibration of the threshold t as $f = \operatorname{sign}\left(g - t\right)$, one could get everything classify as positive or as negative. Describing the phase in between lead to several performance measure. Let's quote the ROC curve, standing for Receiver Operating Characteristic, which describe *tp* against *fp* as t varies; the AUC, for Area Under the Curve, which is the area under the ROC curve; the precision-recall, which work as the ROC curve but showing P against R.

Scoring Procedure

Because of the small database, one should be really careful about the scoring procedure. In fact, one can do a little calculation, on the 101 well labelled children there is 101! way to order them, among which 20!81! order the autistic children first and the not-autistic after, leading to a perfect scoring. This is really not likely to appear. Yet if we know only score on one third of the dataset (keeping the rest for training), this goes to the 7 among 33. Thus one should be careful not to over-estimated result quality.

Leave one subject out. One concern using the training-testing splitting is that autism as several form, and probably if we split our data in 13 autistics for training and 7 for testing, the 7 in testing might present different symptoms than the 13 in training not enabling our method to detect them as autism, while the overall process would work on a bigger database. For this reason, people use the leave-one-subject-out procedure, consisting in training the method on all the kids but one, before evaluating the method on this last kid. Then doing it again with an other kid, and so forth, iteratively. This to score over the entire database. Yet, it suffer of some instability due to the fact that soft-prediction range can depend on the training set, and it sometimes doesn't really make sense to concatenate soft-prediction over different folds, before computing precision-recall or other results.

Beating the state of the art? The state of the art in psychologist-free screening is given by the MCHAT, it shows a AUC of 90 %, and a precision of 50 % for a recall of 97 %, which is to warn for autism with missing only 3 % of autistic kids, there will be 50 % of fake alarms (normal kids warned for autism). After trying different methods fitting parameter on training set and evaluating on testing, some seems to beat this results, as shown Figure 3.2. However, those results weren't stable when switching testing set. Moreover, one can contest the medical interest of having a black-box pretending to screen based on such a small database.

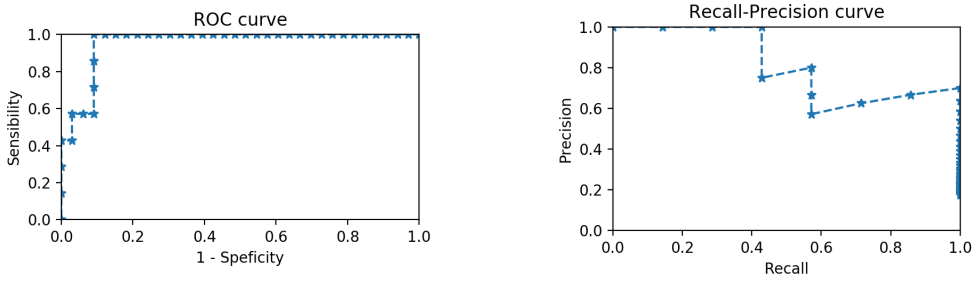

Figure 3.2: Testing results with joint-histogram and rbf. Svm.

3.1.3 Understandable Learning

Up to now we have been looking at blind machine learning. From a psychologist perspective, the diagnosis should be easily understandable from few full of sense features. Thus, let's try to reduce the number features on which to based our classifier, or at least try to build classifier where it is easy to understand what was learnt.

Features Reduction

If we want to use few features while relying on the Euclidean structure of the design space, one can adapt the precedent method to sum up the Euclidean information. It mainly consists in removing features until achieving linear independence. Yet, one should assure that the features that remain aren't random. For this purpose, we suggest a simple algorithm, where the features are first order regarding how much they discriminate autistic people from normal kid, before applying a Gram-Schmidt process where we also discard features that are too much correlated.

Algorithm 4: Features Reduction

input: $(x_{i,j})_{i,j}$ the value of feature $j \in J$ on child $i \in I$;
for $j \in J$ **do**
  Order I regarding $(x_{i,j})_i$ (argument sort);
  Count the number of autistic people in the extremity of the order;
  Assert this as a score s_j to the feature j;
Deduce a new order on J regarding $(s_j)_j$;
K = [];
for $j \in J$ *with the new order* **do**
  keep = True;
  for j_1 *in keep* **do**
    if *the correlation between* $(x_{i,j})_i$ *and* $(x_{i,j_1})_i$ *is too important* **then**
      keep = False;
  if $(x_{i,j})_i$ *is in the span of* $\{(x_{i,j_1})_i\}_{j_1 \in K}$ *(tracked by Gram-Schmidt)* **then**
    keep = False;
  if *keep* **then**
    K append j

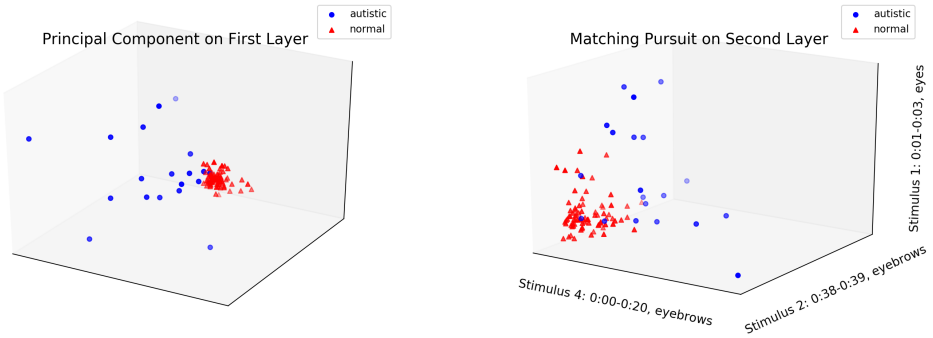

Figure 3.3: Points after the feature reduction step

Sparse Regression

If we are looking for a method supported by a sparse subset of all the putative features put in our descriptors, one can take back to the sparsity problem (2.2.2) already encounter in section 2.2. Let's call X the design matrix, correponding to the earlier $(x_{i,j})_{i,j}$. We are looking for $(w_j)_j$ such that $Xw \approx y$. w_j is nothing but the weight associated with features j. Not using feature j means $w_j = 0$, thus using a minimum of features means minimizing $\|w\|_0$. And the sparsity problem

$$\begin{array}{ll} \text{minimize} & \|w\|_0 = \#\{i \,|\, w_i \neq 0\} \\ \text{subject to} & y \approx Xw \end{array}$$

We have already develop a convex relaxation and a pursuit method to approach this problem. The pursuit method consist in adding features one by one. Reciprocally, one could try to remove features one by one, looking at the feature the most useless for a classifier, for example using a least square classifier:

Algorithm 5: Recursive Features Elimination

Begin with $K = J$, $w = X^\dagger y$ and $i = \arg\min |w_i|$;
while $\|Xw - y\| \leq \varepsilon$ **do**
  Remove i of K;
  $w_K = X_K^\dagger y, \qquad w_{\bar{K}} = 0$;
  $i = \arg\min_{i \in K} |w_i|$;

Generalization. One can argue that autism has different form, thus linear separation seems less relevant than a neighborhood method. Yet, how to enhance features sparsity for such algorithms. One response if that we are seeking to replace x by Wx, where W is a matrix maximizing the number of unused coordinates, or mathematically, the number of canonical vector e_i in its kernel, or graphically, the number of null columns. This could be cast in a variational framework with the penalty:

$$\lambda \|W\|_{p,0}$$

And its convex relaxation as

$$\lambda \|W\|_{p,1}$$

As one can see, with $p = 1$ this formulation looses the information we wanted to put on column sparsity, but what would be the good value for p? A simple understanding of the ℓ^1 penalty is the following, let's consider the least square problem, the solution is given by

$$w \in X^\dagger y + \ker X$$

If we want to minimize $\|w\|$ keeping this constraint, a graphical way to obtain the solution can be seen on Figure 3.4. It consists in blowing the ℓ^1 ball as a balloon, and wait for the ballon to hit the hyperplane $X^\dagger y + \ker X$. Because this balloon is like a

diamond (ℓ^1 ball), it is more likely to hit the plane on a corner or a sharp edge rather than a plane surface and this sharp edge correspond to some coordinate being null.

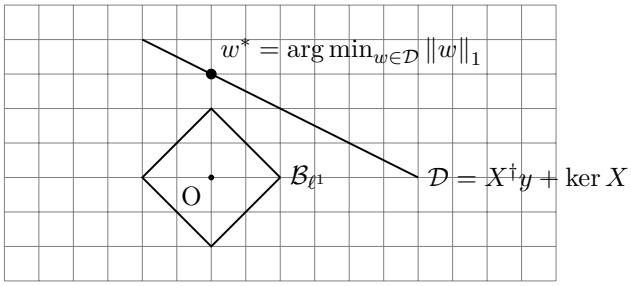

Figure 3.4: ℓ^1 penalty enhance sparsity

Similarly, we would like now to construct a ball $\mathcal{B}_{\mathcal{N}}$ that we choose for convenience of the type $\mathcal{B}_{p,1}$, such as the sharper part of the ball more likely correspond to null columns. To make this more precise, we could look at the protuberance of the ball in the Euclidean metric, and make sure they correspond to points where a maximum of column are nulls. In other term looking to solve with the norm $\mathcal{N}$ as a variable

$$\begin{array}{ll} \text{minimize} & \|A\|_{*,0} \\ \text{subject to} & A \in \arg\max_{\|D\|_2=1} \mathcal{N}(D) \end{array}$$

Yet this formulation doesn't seem the easiest to manipulate. An other solution consists in making sure that the ball $\mathcal{B}_{\mathcal{N}}$ is shaped to contain a maximum of the spaces

$$V_i = \{D \,|\, \forall\, j \neq j, \quad De_j = 0\}$$

Because we are looking for coordinate symmetric ball, let's only consider $V = V_1$, and cast the problem as maximizing

$$\text{maximize} \quad \frac{\lambda_n \left(V \cap \mathcal{B}_{\mathcal{N}}\right)^n}{\lambda_{n^2}\left(\mathcal{B}_{\mathcal{N}}\right)}$$

Which is nothing but a heavy integral derivation, followed by a function study. Yet, the derivation seems time-waisting regarding it applications. However, the author intuition lead him to expect the result to be $p = 2$, for a question of parameterization invariance of the columns.

Let's now consider this variational penalization with a neighborhood classifier. Radial Basis Function Svm capture the neighborhood idea through the metric

$$K_{i,j} = \exp\left(-\frac{\|x_i - x_j\|^2}{\sigma^2}\right)$$

This could be cast under the matrix formulation and a point-wise exponential

$$K = \exp_\odot\left(\frac{1}{\sigma^2}\left(2XX^T - \sum_{kl} E_{kk} X X^T E_{kl} - \sum_{kl} E_{lk} X X^T E_{kk}\right)\right)$$

With the transformation of the design as XW^T, one can write a penalized SVM version as

$$\begin{array}{ll} \text{minimize} & \frac{1}{n}\mathbf{1}^T\xi + \lambda\alpha^T\mathbb{K}\alpha + \mu\mathbf{1}^T\sum_{i=1}^n E_{ii}Me_i \\ \text{s.t.} & \mathbb{K} = \exp_\odot\left(\frac{1}{\sigma^2}\left(2XMX^T - \sum_{kl}E_{kk}XMX^TE_{kl} - \sum_{kl}E_{lk}XMX^TE_{kk}\right)\right) \\ & M \succeq 0, \qquad \xi \geq 0, \qquad \xi \geq \mathbf{1} - \operatorname{diag}(y)\mathbb{K}\alpha \end{array}$$

before factorizing M as W^TW while seeking to maximize the number of null columns of W. Supposing σ^2 large enough, this could be relax in an SDP in M and a quadratic problem in α and ξ, even if it won't become jointly convex in (M, α), removing guarantee to find optimal solution with an intuitive block-coordinate descent algorithm.

Moreover, once getting M, one still have to factorize it as $M = W^TW$ with maximizing the number of null columns. Naturally one can reuse the same ideas minimizing

$$\left\|M - W^TW\right\|^2 + \lambda\mathbf{1}^T\left(\sum_{i=1}^n E_{ii}W^TWe_i\right)$$

Formulation that can be cast as a quadratic problem in the coefficient of W.

However, this seems too weak to take the time to implement it, one could rather try an other way to generalize features selection to non-linear model, probably dropping of the penalized variational framework.

Classification Trees

Looking at some features response could give us some strong assurance about a kid weird behavior related to autism. Yet for a middle range response, one could look at an other indicator, remembering this first response. In other term we are building a decision tree.

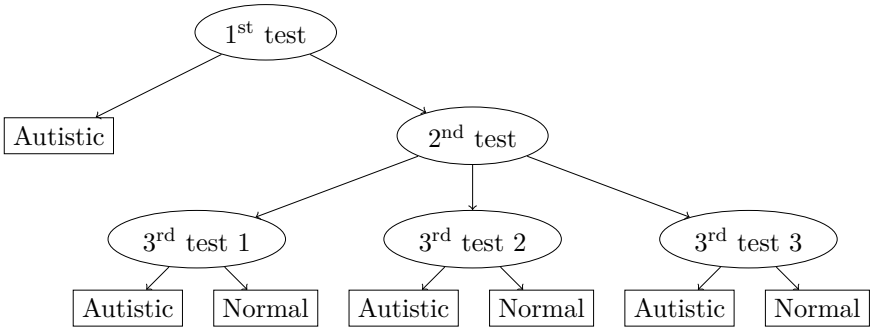

Figure 3.5: Decision Tree

Let's think about how to build this decision tree. Simple test could be offered by each features of our second layer. As the architecture of the tree grows up, the more complex decisions could be made, which can offer great utilities to capture the

subtlety of autism expression, but also is a risk for overfitting. Thus, we would like to build the architecture automatically. At the top we would like to put the most discriminative test. Then, based on statistic property, we will choose if it can lead to a final diagnosis at least on a part of the dataset. However, this test will split the dataset, and on each part of the splitting we can perform again the same procedure. This suggest a recursive algorithm.

So we have moved the main difficulty in designing, given a dataset, a simple-to-understand test to split the data in several part in the best way for the overall classification tree. We would like to achieve a discriminative and robust splitting. Discriminative mean that we would like to reduce the class-variability into each part of the splitting, and robust mean that we would like the separation between the different clusters to be wide. In our implementation, we retook the scoring method consisting in looking at queues to find the features to design the test on, before looking at a clear separation close to the middle of the set.

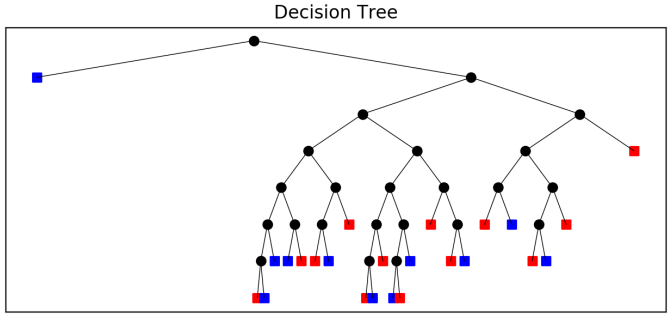

Figure 3.6: Example of a learnt tree

This method with our internal decision and parameter create tree like Figure 3.6, it often allow to obtain perfect results on training set, but those results are unstable. The tree sometimes becoming to precise and lacking of generalization power. Similarly to cross-validation, one could think about adjustment method to avoid overfitting. A first idea consists in trying to remove branches after training with a validation set. An other idea, is to let replace when the tree become to specific, or the depth too important, the tree structure by a linear classifier. What we did is that when few example should be discriminate by a node, rather than looking at a single features, build a linear classifier retaking the whole design.

If the implementation stay really simple, classification tree could become richer by designing fancier test, for example combining all features at a given time, rather than focusing on one channel. Then one would have to rethink about a procedure to choose the good test at each node, without performing a too costly research.

Pattern Recognition

In the recent years, a lot has been done to understand neural network operations. Convolutional Network are more and more well understood. In fact, we have discuss about how one can see filtering-and-thresholding layer as pattern recognition. If we were to implement a neural network, one could look at the filter learnt to understand pattern that were used to diagnose the kids. To perform such a task one should invert locally the network, work that has already been done for convolution network [21].

And when the network gives a diagnosis on a new kid, one can look at the first layer, to see which muscular action was recognize at which moment, the second to see how those were combined to extract expression, the third for behaviors and the last one for the statistics perform on that.

Results Improvement

The features selection step indeed improved the results, showing better results than the MCHAT at least on our dataset, as shown Table 3.1.

Table 3.1: Cross-Evaluation (Second Layer, Features Reduction, rbf Svm)

Fold	AUC	$R \mid P = 1$	$P \mid R = 1$
1^{st}	1	1	1
2^{nd}	.87	.67	.43
3^{rd}	.9	.5	.5
4^{th}	1	1	1
5^{th}	.87	.67	.43
6^{th}	.83	.33	.43
7^{th}	1	1	1
8^{th}	.95	.5	.67
MCHAT	.9	P = .53, R = .97	

If we perform features selection on the whole dataset before performing cross-evaluation, once again we achieve really good results with only five features shown Table 3.2, AUC derive with cross-evaluation is .96 on average.

Table 3.2: Few features to diagnose kids

Stimulus	Channel	Layer 1	Layer 2
2	x	Derivative 1 s	cumul 0:32–0:33
2	x	Derivative 100 s	var 0:40–0:41
2	eyebrows	Gaussian 10 s	var 0:39–0:45
4	eyes	Gaussian 1 s	var 0:19–0:25
4	eyes	Derivative 10 s	mean 0:26–0:27

3.2 Modeling of Normal Behaviors

If we primary aimed at blind machine learning, the precedent work have shown us the importance of going toward psychological knowledge. In fact, if the tendency pushes toward machine learning tools, the size of the dataset rather only enable to discriminate between tests suggested by psychologists.

Since, up to the author knowledge, autism names rather atypical behaviors than a biological pathology, it is natural to learn from normal patient a model for normal behaviors before looking for atypical response in the toddlers and screen for autism. In the following, we focus on different potential factor of interest to design for each of them a model of normal behaviors. There is an interaction between psychological supposition and numerical findings, to declare what are those factor of interest and what are normal behaviors for each of those.

3.2.1 Psychologist Review

Let's first try to guess features or behaviors that could distinguish between a normal and a abnormal response. To do so, best is to refer to psychologist knowledge, or in persons, or reading psychological paper on autism, such as [7]. But also trying to generalize those behaviors that were designed to be catch by human to potential behaviors that only a computer can completely get such as micro-movements, or face asymmetric expression.

We could differentiate social behaviors, such as lack of interaction, connectedness and play, from motor behaviors, such as delays, non-smooth visual tracking and slow reactivity, and from attention behaviors, such as repetitive interests and inability to disengage attention. On the dataset, this might results in two different types of features to compute, looking for local response (such as reaction when the bunny fall on one video), and precise behaviors over the entire dataset (such as blink rate).

We suggest the following long-term behaviors to review:

- Blink rate
- Asymmetry
- Micro-movements
- Non-smooth movements
- Cumulative motion
- Motion variability
- Organization of motion

Computing such features are more or less subjective, for blink rate one should look at the number of time eyes get closed ; for asymmetry, compute differences between left and right facial features. For motion, we have told the importance to look at response against a Gaussian derivative filter ; for micro-movements, look at micro-scale response, for non-smooth movements at correlation in time ; for cumulative distance at ℓ^1 distance, for variability at variance ; and for motion organization at correlation between muscles or channels.

We suggest the following events to respond:

- Tower construction (three times)
- Tower falling (twice)
- Sound (twice)
- Name call (twice)
- Bunny moving ear (four times)
- Bunny falling
- Bunny lift up

Eventually looking during those event for response time, following of objects and disengagement of attention.

3.2.2 Average and Abnormal Modeling

Let's begin with the motion features, that we can easily define retaken the prior work on features extraction. Once we achieved to extract those features, one could retaken classifiers introduced previously and see how well they perform on those new features. However, we have changed of framework, and we know want to detect weird response that can set of thinking of a singularity of the kid.

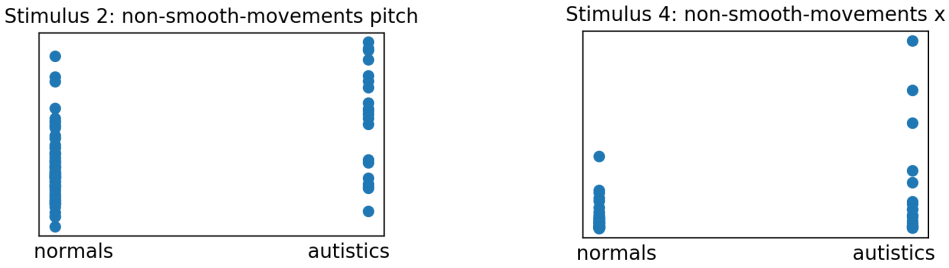

Figure 3.7: Generative or detection Framework (each dot is a kid)

Figure 3.7 show two types of features. The left one where the overall autistic distribution is different from the normal one, suggesting to introduce a generative framework to do the screening. The right one suggest rather to use a detection framework, noticing response that are singular before learning if those push the diagnosis toward autism or not.

Generative Modeling

Generative modeling consists in inferring the distribution of the classes, before using those to classify a new example. We have already seen in subsection 2.3.1 some technics to model a distribution from some samples, even for vectorial distribution.

Figure 3.8 show two examples of density, on the left, one can remark that autistic kid tend to make less smooth-movement than normal kids. The difference is especially significative for the pitch on the second stimuli. On the right, one can remark that normal kids tend to correlate more depth motion with mouth opening. One can try to give an explanation for this phenomenon, as if normal kids where to make depth motion when astonished or paying attention, linked with mouth motion, while

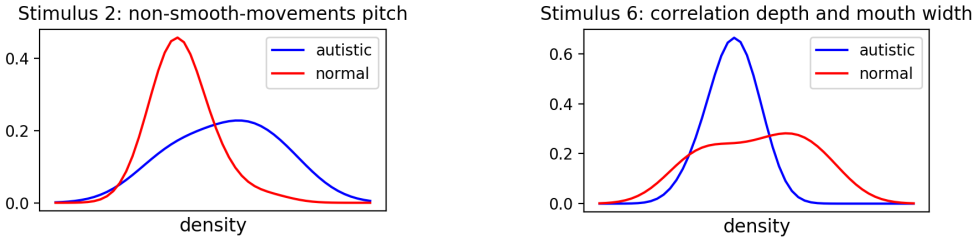

Figure 3.8: Density Estimation over classes

autistic kid make back-and-fourth depth motion without linking mouth. Of course, this explanation is subjective.

One we have modeled $(X|\, y)$, one can classify a new example looking at

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}$$

This justify to calibrate our classifier regarding the likelihood ratio:

$$\frac{p(x|y=1)}{p(x|y=-1)}$$

In fact, thresholding this ratio to classify x is the test that minimize the probability of fake negative, given its probability of fake positive. This results is known as Neyman-Pearson lemma [23]. Yet, here we don't access the true distribution, but an empirical estimation of it, and one should remember that this test can sometime hold to strong view on things if the estimation was not flexible enough to unseen-before singularity.

Of course, one could use this for a vectorial distribution. Yet, as the dimension increase the number of sample need to estimate the density increase, and it will be better to introduce independence hypothesis to factorize the law. Moreover, conditional independence hypothesis could also help in factorization. Using smartly those hypothesis to reduce computation is the goal of *Graphical Models* with the well-known sum-product or Viterbi algorithms.

With independence hypothesis, keeping only well-separated features, this simple method achieve 80 % of AUC on average with some perfect trials. And it is actually as good as our Svm classifier, even on second layer. Moreover, this method is way more robust to the small number of example.

Singularity Detection

Once again implementing an abstract idea can be done in several way. What is a singularity? One can look at tails of at the overall distribution, or point showing a particularly low likelihood. Regarding those singular response, two conclusions can be drawn. Either it is a unexpected behaviors pushing the diagnosis toward autism. Either it is a response that show kid reactivity pushing the diagnosis toward normality.

Computing all singularities of a kid, one should then perform a vote to draw a final diagnosis. Each singularity should contribute in a vote with a certain weight. Eventually those weight could be learn on a training set. Yet, since singularity are more likely not to occur, the dataset seems small to learn how to explain a singularity. We suggest to learn the weight associating with a feature x by looking at the average likelihood of example of both classes, in other term:

$$w = \log\left(\frac{\mathbb{E}\left[\varphi\left(p(X)\right)|\, y=1\right]}{\mathbb{E}\left[\varphi\left(p(X)\right)|\, y=-1\right]}\right)$$

where φ is a function allowing to amplify the low response (*i.e.* the singularities) in the ponderation. For example, one could take $\varphi(x) = x^{\alpha}$ with $\alpha < 0$. Then a classfication score is given collecting all the features

$$f(x) = \sum_i w_i.\psi\left(p(x_i)\right)$$

where ψ compute a singularity score, *e.g.* $\psi(x) = \mathbf{1}_{x<a}$.

However, this empirical method seems a little bit fetched, and one could correct this process based on statistical test property to distinguish two distributions

$$\mathcal{D}_i = (1 - w_i).\mathcal{L} + w_i.\mathcal{L}_i$$

where the left term correspond to the normal response, and the right term correspond to singularity proper to each distribution.

In fact taking $\psi = \varphi$, for example both equal to $x \to x^{-1}$, suggest to rather compute from the all the dataset the new features $\varphi\left(p(x_i)\right)_i$, before learning a linear classifier on it. In other terms, we have added two new classifiers to our collection. In practice, the first one works better, allowing to improve results given by precedent methods up to 95 % of AUC if tested after features selection on second layer, and around 85 % without features selection, using $\varphi(x) = x^{-2}$, $\psi(x) = \mathbf{1}_{x<.1}$.

3.2.3 Average and Abnormal Responses

Characterize what is a normal and an abnormal responses isn't an easy task, psychologists can have some guess but were not able to turn it into an procedural method for us to implement it in an algorithm. This part is difficult cause in it raw form, it is once again in high dimension and one can easy over-interpret over the small dataset. To reduce the number of parameter, we have used a simple implementation of the clustering method proposed in the pattern recognition section, section 2.4, to identify different types of response.

This simple method, consists in extracting on the video of a kid, a part where the kid is supposed to response to an event, *e.g.* the three seconds after the tower is falling, see Figure 1.1. We then try to cluster each frame of the video, regarding the face parameterization, before looking at an histogram of the clusters appearing. In

other terms, we look at few different types of face that can appears and describe how often each of them appear on this specific event response.

How to characterize the different types of faces and how to assign a particular frame to a specific type? This is done through the introduction of a metric on the different faces. In this metric space, we suppose that few clusters will appear corresponding to the different types of faces. Recognizing clusters is easily done with classical algorithm such as the *k-means*. One can choose if one want the cluster to be the same for each event or if one want them to be event-specific. To design the metric, we take the ℓ^2 distance on the face parameterization after having homogenize the different features by normalizing the mean and standard deviation. It perform around 75 % of AUC on average on cross-evaluation, which is honorable for a simple method.

3.2.4 From Clustering to Filtering

As we have discuss in section 2.4, the cluster learnt can be seen as word or pattern. And those pattern could be use as a filter against which to convolve the signal in order to recognize it. This provide a simple way to look at word appearing all along the video on the two different groups of kids to try to discriminate some specific behaviors. Of course, our implementation stay simple and if we aimed to this general face recognition framework, in reality, we obtained a gray-box method.

In practice, we don't notice a big differences between the two groups with this method. The main remarkable thing is that normal people tend to response against the filter learnt than autistic people, as shown on Figure 3.9. Moreover, simple SVM or other method allow to achieve 80 % of AUC on cross-evaluation.

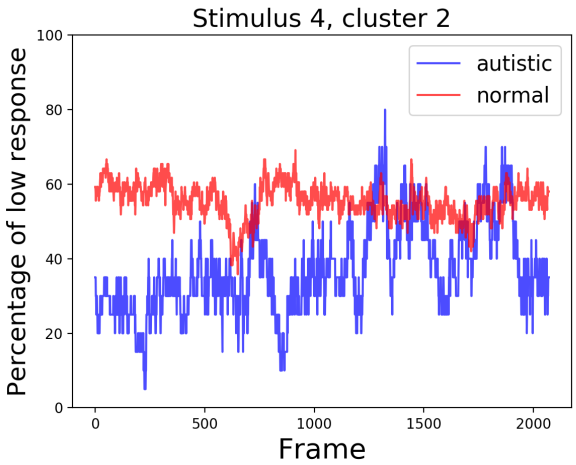

Figure 3.9: Percentage of low responses by group, computed over a video

Bibliography

[1] N. Akshoomoff, C. Corsello, and H. Schmidt. The role of the autism diagnostic observation schedule in the assessment of autism spectrum disorders in school and community settings. *The California school psychologist: CASP*, 11:7–19, 2006.

[2] N. Aronszajn. La Théorie des Noyaux Reproduisants et ses Applications. *Proc. Cambridge Philosophical Society*, 39:133–153, 1943.

[3] American Psychiatric Association. *Diagnostic and statistical manual of mental disorders : DSM-5*. APA Publishing, 5$^{\text{th}}$ edition, 2013.

[4] J. Baio. Prevalence of autism spectrum disorder among children aged 8 years — autism and developmental disabilities monitoring network, 11 sites, United States, 2010. *Morbidity and Mortality Weekly Report*, 63(SS02):1–21, 2014.

[5] H. Barlow and P. Földiák. Adaptation and decorrelation in the cortex. *The Computing Neuron*, pages 54–72, 1989.

[6] G. Beylkin, R. Coifman, and V. Rokhlin. Fast wavelet transforms and numerical algorithms. *Communications on Pure and Applied Mathematics*, 44:141–183, 1991.

[7] S.E. Bryson, L. Zwaigenbaum, and W. Roberts. The early detection of autism in clinical practice. *Paediatrics & Child Health*, 9(4):219–221, 2004.

[8] D. L. Christensen, J. Baio, K. V. Braun, et al. Prevalence and characteristics of autism spectrum disorder among children aged 8 years. *Autism and Developmental Disabilities Monitoring Network*, 65(3):1–23, 2012.

[9] National Research Council. *Educating Children with Autism*. National Academy Press, Washington, DC, 2001.

[10] C. Darwin. *The Expression of the Emotions in Man and Animals*. John Murray, London, England, 1872.

[11] I. Daubechies. Orthonormal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics*, 41(7):909–996, 1988.

[12] G-B. Duchenne. Mécanisme de la physionomie humaine, ou analyse électro-physiologique des différents modes de l'expression. *Archives générales de médecine*, 1, 1862.

[13] P. Ekman and W. V. Friesen. *Facial Action Coding System: A Technique for the Measurement of Facial Movement*. Consulting Psychologists Press, Palo Alto, California, 1978.

[14] F. Galton. Composite portraits. *The Journal of the Anthropological Institute of Great Britain and Ireland*, 8:132–142, 1878.

[15] R. Ginouvès and A-M. Guimier-Sorbets. *La constitution des données en archéologie classique. Recherches et expériences en vue de la préparation de bases de données.* Éditions du CNRS, 1978.

[16] J. Hashemi et al. A scalable app for measuring autism risk behaviors in young children: A technical validity and feasibility study. *EAI Endorsed Trans. Scalable Information Systems*, 3(10), 2016.

[17] C.-J. Hsieh et al. A dual coordinate descent method for large-scale linear svm. *Proceedings of the 25th International Conference on Machine Learning*, pages 408–415, 2008.

[18] A. M. Kozlowski, J. L. Matson, M. Horovitz, J. A. Worley, and D. Neal. Parents' first concerns of their child's development in toddlers with autism spectrum disorders. *Developmental Neurorehabilitation*, 14(2):72–78, 2011.

[19] R. Lang and P. K. Knox. The new metropolis: Rethinking megalopolis. *Regional Studies*, 43(6):789–802, 2009.

[20] P. Lucey et al. The extended cohn-kanade dataset (ck+): A complete facial expression dataset for action unit and emotion-specified expression. *IEEE workshops on CVPR*, pages 94–101, 2010.

[21] Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. *CoRR*, 2014.

[22] H. A. Murray. *Explorations in personality.* Oxford University Press, New York, 1938.

[23] J. Neyman and E. S. Pearson. On the Problem of the Most Efficient Tests of Statistical Hypotheses. *Philosophical Transactions of the Royal Society of London Series A*, 231:289–337, 1933.

[24] Julie Osterling and Geraldine Dawson. Early recognition of children with autism: A study of first birthday home videotapes. *Journal of Autism and Developmental Disorders*, 24(3):247–257, 1994.

[25] D. L. Robins, D. Fein, M. L. Barton, and J. A. Green. The modified checklist for autism in toddlers: an initial study investigating the early detection of autism and pervasive developmental disorders. *Journal of autism and developmental disorders*, 31(2):131–144, 2001.

[26] K. Weinberger and L. Saul. Distance metric learning for large margin nearest neighbor classification. *J. Mach. Learn. Res.*, 10:207–244, 2009.

[27] T. Ying-li, T. Kanade, and J. F. Cohn. Recognizing action units for facial expression analysis. *IEEE transactions on pattern analysis and machine intelligence*, 23(2):97–115, 2001.