



HAL
open science

Probing machine-learning classifiers using noise, bubbles, and reverse correlation

Etienne Thoret, Thomas Andrillon, Damien Léger, Daniel Pressnitzer

► **To cite this version:**

Etienne Thoret, Thomas Andrillon, Damien Léger, Daniel Pressnitzer. Probing machine-learning classifiers using noise, bubbles, and reverse correlation. *Journal of Neuroscience Methods*, 2021, 362 (109297), 10.1016/j.jneumeth.2021.109297 . hal-03063763

HAL Id: hal-03063763

<https://hal.science/hal-03063763>

Submitted on 7 Jan 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

1 **Probing machine-learning classifiers using noise, bubbles, and** 2 **reverse correlation**

3

4 Etienne Thoret^{*1,4}, Thomas Andrillon³, Damien Léger², Daniel Pressnitzer¹

5

6 ¹ Laboratoire des systèmes perceptifs, Département d'études cognitives, École normale
7 supérieure, PSL University, CNRS, 75005 Paris, France.

8 ² Université de Paris, APHP, Hotel Dieu, Centre du Sommeil et de la Vigilance & EA 7330
9 VIFASOM, Paris 75006, France.

10 ³ Turner Institute for Brain & Mental Health and School of Psychological Sciences, Monash
11 University, Melbourne 3168, Australia.

12 ⁴ Aix Marseille Univ, CNRS, PRISM, LIS, ILCB, Marseille, France

13 * corresponding author: etiennethoret@gmail.com

14 **Abstract**

15 Many scientific fields now use machine-learning tools to assist with complex classification tasks. In
16 neuroscience, automatic classifiers may be useful to diagnose medical images, monitor
17 electrophysiological signals, or decode perceptual and cognitive states from neural signals. Tools
18 such as deep neural networks regularly outperform humans with such large and high-dimensional
19 datasets. However, such tools often remain black-boxes: they lack interpretability. A lack of
20 interpretability has obvious ethical implications for clinical applications, but it also limits the
21 usefulness of machine-learning tools to formulate new theoretical hypotheses. Here, we propose a
22 simple and versatile method to help characterize and understand the information used by a classifier
23 to perform its task. The method is inspired by the reverse correlation framework familiar to
24 neuroscientists. Specifically, noisy versions of training samples or, when the training set is
25 unavailable, custom-generated noisy samples are fed to the classifier. Variants of the method using
26 uniform noise and noise focused on subspaces of the input representations, so-called “bubbles”, are
27 presented. Reverse correlation techniques are then adapted to extract both the discriminative
28 information used by the classifier and the canonical information for each class. We provide
29 illustrations of the method for the classification of written numbers by a convolutional deep neural
30 network and for the classification of speech versus music by a support vector machine. The method
31 itself is generic and can be applied to any kind of classifier and any kind of input data. Compared to
32 other, more specialized approaches, we argue that the noise-probing method could provide a generic
33 and intuitive interface between machine-learning tools and neuroscientists.

34

35 **Keywords:** Data analysis – Interpretability – Deep neural networks – Automatic classifiers – Reverse
36 correlation – Auditory models

37

38 **Introduction**

39 Applications of machine-learning techniques permeate more and more scientific fields, with rapid
40 and sometimes unexpected success (LeCun et al., 2015; Jordan & Mitchell, 2015; Krigeškorte &
41 Douglas, 2018; Richards et al., 2019). At the same time, it is becoming a widely-acknowledged issue
42 that many of these tools are often used as black boxes, and need to be interpreted (Molnar, 2020;
43 Doshi-Velez & Kim, 2017). For instance, if a Deep Neural Network (DNN) was used to make life-
44 changing decisions such as deciding on an intervention based on medical imagery, both the clinicians
45 and patients would have a clear desire to know the rationale that motivated the decision. Also, the
46 power of classifiers to detect useful information in large datasets holds many promises to improve
47 theoretical models, but then, understanding at least to some extent the classifier's operation is crucial
48 (Zihni et al., 2020).

49 Understanding what a complex classifier does after being trained on possibly millions or
50 billions of samples is usually hard. It is hard for a reason: if the task that the classifier solves had a
51 known explicit solution, then there probably would not have been any incentive to develop the
52 classifier in the first place. In addition, modern techniques involve artificial network architectures
53 with interconnected layers, each including highly non-linear operations (Sejnowski, Kienker, &
54 Hinton, 1986). A lot of the computational power of such algorithms lies in such cascades of feed-
55 forward and feed-back non-linear operations. Unfortunately, human reasoning seems most at ease to
56 generate intuitions with linear processes, and not for complex combinations of non-linear ones.
57 As a consequence, designing methods to interpret machine learning tools is a fast-growing field of
58 research of its own right, often designated under the term *Explainable AI* (Guidotti et al., 2018). It
59 has dedicated journals within the machine learning community (e.g. *Distill*) and an associated
60 DARPA challenge (*XAI*). Recent reviews covering the types of methods exist (Molnar, 2020), also
61 covering more specifically the feature visualization approach taken here (Olah et al., 2017). Within
62 this context, our aim is not to outperform the state-of-the art specialized interpretability methods, but
63 rather to provide a general tool that will hopefully be intuitive to neuroscientists, as it is based on
64 familiar methods for this community. The manuscript describes the method, provides an open

65 software library to use it, and shows examples of application, demonstrating how it can achieve
66 useful results.

67 The gist of the method is to try and reveal the input features used by an automatic classifier, a
68 black-box, to achieve its task *without any knowledge* about what is inside the black-box. As such, it is
69 what is termed an “agnostic” method of explanation: it does not attempt to describe mechanistically
70 the operation of a specific classifier, which it considers unknown (even if the classifier’s details are
71 available, as they may be too complex to understand intuitively). Rather, the aim is to relate features
72 of the input space to the classifier’s decisions. Such a problem is closely related to issues that
73 neuroscientists and experimental psychologists have been addressing for years: providing useful
74 insights for theoretical models of, for instance, human perception, without a full knowledge of the
75 highly complex and non-linear underlying information processing performed by the brain.

76 In particular, the method we propose is directly inspired from the reverse correlation
77 techniques developed for studying human vision (Ahumada et al., 1971; Neri et al., 1999; Gosselin &
78 Shyns, 2001, 2003). Reverse correlation is based on linear systems analysis (Wiener, 1966). It uses
79 stochastic perturbations of a system to observe its output. If the system were linear, an average of the
80 inputs weighed by the observed outputs would be able to fully characterize the system. However,
81 even for the highly non-linear systems studied by neuroscience, reverse correlation has a track-record
82 of useful applications. For neurophysiology, averaging input stimuli according to neural firing rates
83 has been used to describe neural selectivity (Ringach & Shapley, 2004 for a review). For
84 psychophysics, averaging input stimuli according to participant’s decisions has revealed stimulus
85 features on which such decisions are made for detection or discrimination tasks (Ahumada et al.,
86 1971; Gosselin & Shyns, 2001, 2002). In this spirit, it seems appropriate to add reverse correlation to
87 the toolbox of techniques to probe automatic classifiers, as its advantages and limitations are already
88 well understood for non-linear systems.

89 One important benefit of using the reverse correlation framework is its complete
90 independence from the underlying classifier’s architecture. Unlike efficient but specific methods
91 tuned to a classifier’s architecture (see Guidotti et al., 2018 for a review), the reverse correlation can
92 be used to probe any algorithm that separates the input data into distinct classes. Even for the

93 currently popular agnostic interpretability methods, this is not always the case: Class Activation
94 Maps (Zhou et al., 2016) are specific to convolutional networks; LIME (Ribeiro et al., 2016) and
95 RISE (Petsiuk et al., 2018) highlight features of specific examples which may or may not be
96 representative of the classification task in general. Also, the method operates in the same
97 representation space used as an input to the classifier, and can be applied to any type of
98 representation (2D images, 1D time series such as audio, higher-dimensional brain imaging data, for
99 instance).

100 The outline of the method is as follows. First, a set of stochastic inputs are generated, by
101 introducing noise on the training dataset when available, or, when unavailable, by generating broad-
102 band noise to cover systematically the input space. The noise takes two forms: additive noise, as is
103 classically the case, but also multiplicative low-pass noise known as “bubbles” (Gosselin & Shyns,
104 2001, 2002) to focus the exploration on sub-spaces of the input representation. Second, the inputs are
105 sorted according to the classification results. Third, inputs belonging to the same class are grouped
106 together, with some refinements of the standard reverse correlation methods inspired by signal
107 detection theory (Green & Wets, 1966) to weigh the results with the variability observed after
108 classification. Two variants are described, aiming to probe two kinds of possibly overlapping but not
109 necessarily identical input features: (1) the *discriminative* features, which correspond to the part of
110 the input representation that is the most useful to ascribe a category (2) the *canonical* features, which
111 correspond to the input features most representative of each category. In the machine-learning
112 literature, these would loosely correspond to the “attribution” versus “feature visualization” problems
113 (Ohla et al., 2017). In psychophysics, the distinction overlaps with the “potent information” (Gosselin
114 et al., 2001) versus “prototypical information” (Rosch, 1983).

115

116 **1. Material and Methods**

117 **1.1. Probing discriminative features**

118 We term “discriminative features” the subspaces of the input space that are the most potent in
119 the decision taken by the classifier (Gosselin & Shyns, 2002). The aim of this first method is to

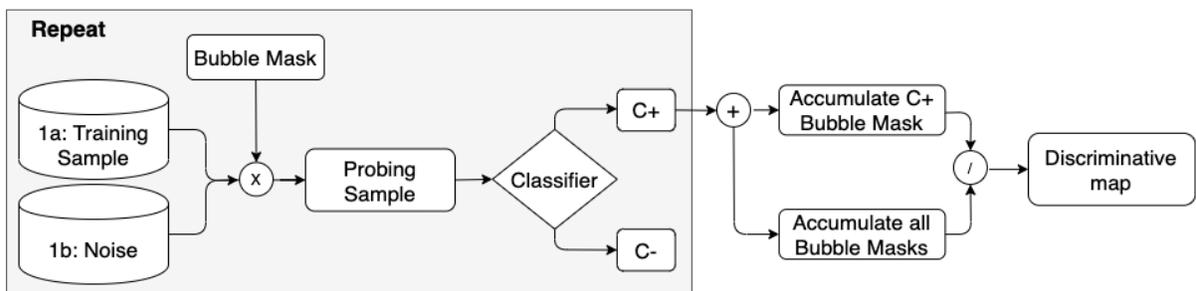
120 visualize such subspaces in the input space. In the following, we assume that the classifier has been
121 trained and is available to the probing method.

122 1.1.1. Procedure

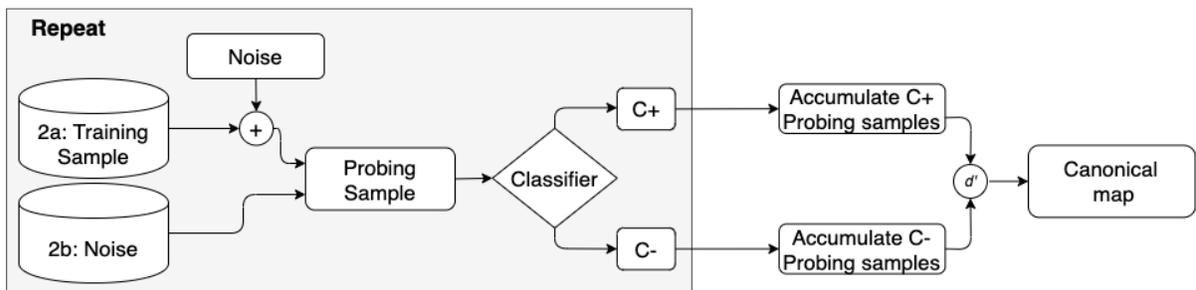
123 To identify discriminative features, the input space is pseudo-randomly sampled with
124 multiplicative low-pass filtered noise. The subspace enabling the highest classification performance
125 is then identified by a reverse correlation analysis of all classified samples. The algorithm is directly
126 inspired by the “bubbles” method (Gosselin & Shyns, 2001), originally designed to characterize the
127 visual features underlying human behavioral performance for image classification tasks.

128 We present two sub-variants of the method, to account for the availability or not of the
129 training set: a) multiplicative lowpass noise is applied to the training set; 1b) multiplicative lowpass
130 noise is applied to broadband noise generated in the input stimulus space. We now describe the
131 algorithm, jointly for a) and b). A textual description is provided as well as a software repository
132 written in Python programming language (<https://github.com/EtienneTho/proise>) and a schematic
133 illustration (Figure 1).

Method 1: Discriminative features



Method 2: Canonical features



134

135 **Fig. 1.** Summary of the two probing methods. Both method have 2 variants depending on the
136 availability of the training set. **Method 1:** Training samples (1a) or noise (1b) are multiplied by
137 bubble masks and then fed to the trained classifier. The bubble masks are then sorted according to

138 *the output of the classifier to compute the discriminative map. **Method 2:** Training samples with*
139 *additive noise (2a) or noise (2b) are fed to the trained classifier. The probing samples are then sorted*
140 *according to the output of the classifier to compute the canonical map. For methods 1b and 2b, the*
141 *noise can be either a gaussian noise or pseudo-random samples.*

142

143 For each pass (gray box in Fig. 1):

- 144 1. A bubble mask is generated. This consists of a mask in the input space, of dimension N,
145 consisting of randomly positioned N-dimensional Gaussian windows (see Figure 2). The
146 number of bubbles, *nbBubbles*, as well as the size of the bubbles in terms of the Gaussian
147 standard deviations can be arbitrarily chosen and are parameters of the algorithm. In practice,
148 an input array of dimension N populated by zeroes except for *nbBubbles* unit values is
149 convolved with N-dimensional Gaussian windows. The resulting mask is denoted
150 *BubbleMask*.
- 151 2. The probing data is generated. For variant a), the probing data is one exemplar of the training
152 dataset, randomly chosen. For variant b), the probing data is an N-dimensional activation
153 noise (see section 2.1.2 for details). The probing data is denoted *ProbingData*.
- 154 3. The probing sample is obtained by multiplying the bubble mask with the probing noise:
155 $ProbingSample = BubbleMask * ProbingData$.
- 156 4. The probing sample is fed to the classifier and the output class is recorded. The probing
157 sample is labeled C+ if it classified in the target class, C- otherwise.

158

159 Analysis:

160 For each point, *i*, in the stimulus space, the discriminative map for the class C+, $D_{i,C+}$, is
161 computed as the sum of all C+ bubble masks divided by the sum of all masks C+ and C-:

162
$$D_{i,C+} = \frac{\sum BubbleMask_{C+,i}}{\sum BubbleMask_i}$$
 (Eq. 1). It should be noted that the analysis is performed on the

163 bubble masks, and not on the probing samples.

164 **1.1.2. Generation of noise activations when the training set is unavailable**

165 As mentioned above, when the training set of the classifier is unavailable, the probing
166 samples are generated from noise in the input space. The choice of the noise distributions is a free
167 parameter of the method. The simplest choice is to draw samples from a uniform distribution at each
168 point of the input space, covering the full range of valid input values. However, this sometimes leads
169 to uneven coverage of the output categories, for instance if the classifier's boundaries are especially
170 complex or if the decision algorithm is highly non-linear. In this case, we suggest to generate pseudo-
171 random probing samples by first whitening the input dimensions, using a Principal Component
172 Analysis (PCA). As the training set is unavailable, the PCA can be done on a representative set of
173 inputs relative to the classifier's task. After the PCA, uniform noise can be generated in the low
174 dimension space – which can be seen as the latent PCA's space – and inverted to obtain noise in the
175 input space. The qualitative goal during the choice of the noise distribution is a balanced coverage of
176 all output categories, and iterative choices may be a part of understanding the classifier's features.

177 **1.1.3. Statistics**

178 The discriminative maps show, in the input space, the features used by the classifier to assign
179 samples to a given category. Visual inspection may be sufficient to get a qualitative understanding of
180 the classifier's operation. However, in some cases, it is desirable to assess statistically the relevance
181 of each part of the discriminative map.

182 There are many options to assess significance of such data, from which we outline one
183 possible methodological choice. First, the maps can be shuffled by running the algorithm described
184 above many times while randomly assigning output categories to each sample. For each point in the
185 actual map, a *t*-test (or a non-parametric equivalent) is applied to compare the map value with the
186 mean of the shuffled data. Maps are usually high-dimensional so a correction for multiple
187 comparisons is needed. Again, several choices exist which are not specific to the methodology
188 presented here, including Bonferroni correction, cluster permutation (Maris & Oostenveld, 2007), or
189 False Discovery Rate (FDR) (Benjamini & Hochberg, 1995). In the illustrative examples below we
190 only provide the raw discriminative and canonical maps, without statistics.

191 **1.2. Probing canonical features**

192 We define “canonical features” as the representation, in the input space, that would best match
193 the different items of a given class. As an analogy, the canonical information may be viewed as the
194 centroid of a category in the input space.

195 **1.2.1. Procedure**

196 To build the canonical map, the whole input space is randomly perturbed, without bubbles, so
197 the search is not focused on any subspace. The aim is to probe the whole feature space. Then, all
198 probes classified as members of the same category are averaged, in a direct adaptation of the classic
199 reverse correlation method. However, we introduce here two differences, though. First, for
200 generality, we do not separate correct classifications from false positives or false negatives. This
201 would require to know the training dataset or to have a large labeled testing dataset. Second, a
202 normalization of the feature map is introduced, using standard deviations estimates at each point of
203 the map. This facultative step serves to display units similar to z -scores and not arbitrary input values.
204 Again, we propose two sub-variants of the algorithm depending on the availability or not of the
205 training dataset: a) broadband noise in the input space is added to the training set; b) broadband noise
206 is generated in the input space. We now describe the algorithm, jointly for a) and b). A textual
207 description is provided as well as the scripts (<https://github.com/EtienneTho/proise>) and a schematic
208 description (Figure 1).

209 For each pass:

- 210 1. The probing sample is generated. For a), the probing sample is one randomly chosen exemplar of
211 the training dataset, with noise added. The goal is to perturb the input to introduce variability, so
212 that only the most salient information (to the classifier) remains in the reverse correlation
213 average. For b), the probing sample is an N -dimensional activation noise. The probing sample is
214 denoted *ProbingSample*.
- 215 2. The probing sample is fed to the classifier and the output class is recorded. The probing sample is
216 labeled C+ if it classified in the target class, C- otherwise.

217 Reverse correlation analysis:

218 • For each point, i , in the stimulus space, the discriminative information is computed as the mean of
219 all C+ probing samples minus the mean of all C- probing samples, normalized by the standard
220 deviation of all probing samples at this point in the input space:

$$221 \quad P_i = \frac{\text{mean}(\text{ProbingSample}_{i,C+}) - \text{mean}(\text{ProbingSample}_{i,C-})}{\text{std}(\text{ProbingSample})} \quad (\text{Eq. 2})$$

222 This reverse correlation definition adds a normalization factor to the simple average, using the
223 standard deviation observed over all probing samples. This normalization is inspired from the
224 discriminability index d' of signal detection theory (Green & Swets, 1966). It aims to visually
225 emphasize reliably high values in the canonical map, by transforming the input units to z -score units.
226 Note also that in a binary classification task, P_i is symmetric for the two classes.

227 **1.2.2. Dimensionality reduction**

228 Depending on the architecture of the classifying pipeline and input space, the estimation of the
229 canonical map with reverse correlation can be more or less efficient. In particular, a standard
230 technique to improve efficiency when training a classifier is to reduce the number of dimensions of
231 the input space, for instance by using PCA. (e.g., Patil et al., 2012). Here as well, the probing and
232 reverse correlation analysis can be performed in the space with reduced dimensionality before
233 inverting back to the original input space.

234 For the generation of probing noise in variant b), the same remarks made in section 2.1.2 apply,
235 with the same use of PCA to shape the noise for a balanced coverage of all output classes.

236 **1.2.3. Statistics**

237 The statistical analysis of canonical maps can be done with the same tools as for
238 discriminative maps, described in section 2.1.3.

239 **2. Results**

240 To illustrate the methods introduced above and their generality, we present two different use cases:
241 interpreting the classification of handwritten digits, a visual task (2-D input space) performed with a
242 deep neural network; interpreting the classification of speech versus music, an audio task (1-D time
243 series converted to a 4-D auditory model) performed by a support vector machine. These two cases
244 also cover binary versus multiclass decisions. Although voluntarily simple, these examples should
245 cover most of the ingredients needed for use cases relevant to neuroscience, such as vocal

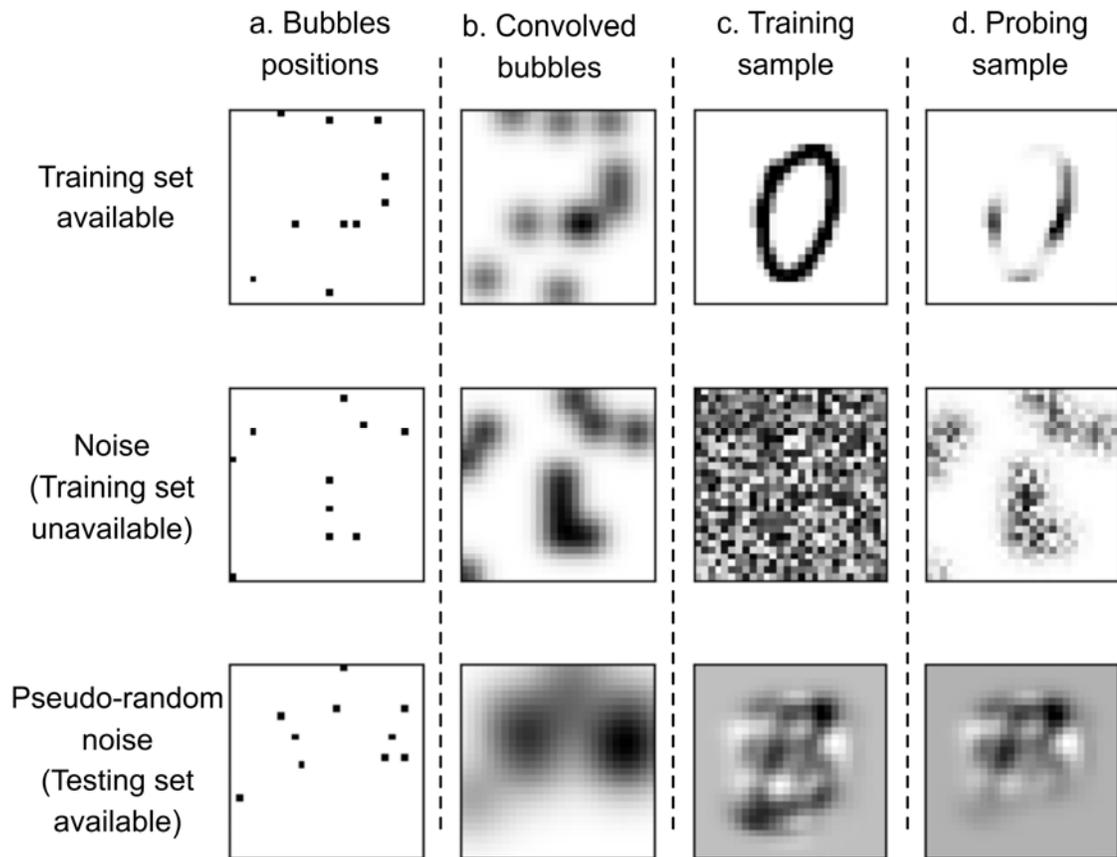
246 classification (Paquette et al., 2018), biomedical images classification (Wang et al., 2016), EEG
247 decoding (King & Dehaene, 2015), Multi-Voxel Pattern Analysis (Formisano et al., 2008).

248 **2.1. Digits classification**

249 In this first example, we classify visual samples of handwritten digits from the *MNIST* database
250 (Deng, 2012). This is a standard database for evaluating image classification algorithms in the
251 machine-learning community. It is composed of handwritten digits, from 0 to 9, with 60000 samples
252 in the training set and 10000 samples in the test set. Each sample is a two-dimensional greyscale
253 image with pixels values between 0 and 1.

254 Many algorithms can now successfully perform this classification task. Here we trained a
255 Convolutional Neural Network (CNN) to discriminate between digits, with the following
256 architecture: 2D-convolutional layer (3, 3), Max Polling layer (2, 2), 2D-convolution layer (3, 3),
257 flattening layer, dense layer with 10 outputs and a softmax activation. Three epochs were run and, as
258 expected, a high classification accuracy of 97% was obtained on the test set.

259 The CNN was probed to visualize the output of our algorithms for discriminative and
260 canonical features. The two variants, with the training set available and without the training set, were
261 compared. Figure 2 visually illustrates the method on these 2-D examples, for the bubbles variant.



262

263 **Fig. 2. Illustration of the probing method with bubbles on 2-D images. Top row: construction of**
264 **the probing samples for discriminative features with the training set available. a) Random**
265 **placement of bubbles b) Convolution with 2-D Gaussian distributions to obtain the bubble mask c) A**
266 **training sample d) Bubble mask applied to the training sample to obtain the probing sample. Middle**
267 **row: discriminative features with training set unavailable, uniform noise. Columns as above.**
268 **Using such probing samples do not cover the output categories efficiently (see text), likely because of**
269 **their qualitative differences with digits. Bottom row: discriminative features with training set**
270 **unavailable, dimension-reduced noise. Here noise is generated in a reduced dimension space**
271 **extracted from PCA over the test set. The resulting probing sample is not a recognizable digit but**
272 **shares visual features with actual digits.**

273

274 Figure 3 shows the discriminative features obtained with Methods 1a) and 1b), expressed as
275 the Discriminative Index of Eq. 1 visualized in the 2-D input space (the visual image). The resulting

277 classification, that is, the position of the bubbles most useful for identifying each class. These do not
278 need to correspond to the actual shape of a digit (which will be targeted by canonical features later
279 on). For example, for the digit “1”, the most useful regions are *around* the digit: knowing that there
280 are no active pixels in such surrounding regions is most efficient for deciding that the narrow-shape
281 of “1” was the input. For the digit “7”, the discriminative map highlights the top-right corner, which
282 corresponds to the position of a sharp angle unique to “7”. In summary, while these maps may not
283 make immediate intuitive sense on their own, they do orient the analysis of the input set towards
284 regions of interest. Moreover, if the task was now to classify “7” versus all other digits, the input
285 space could be weighed to emphasize the top-right corner to simplify the new classifier.

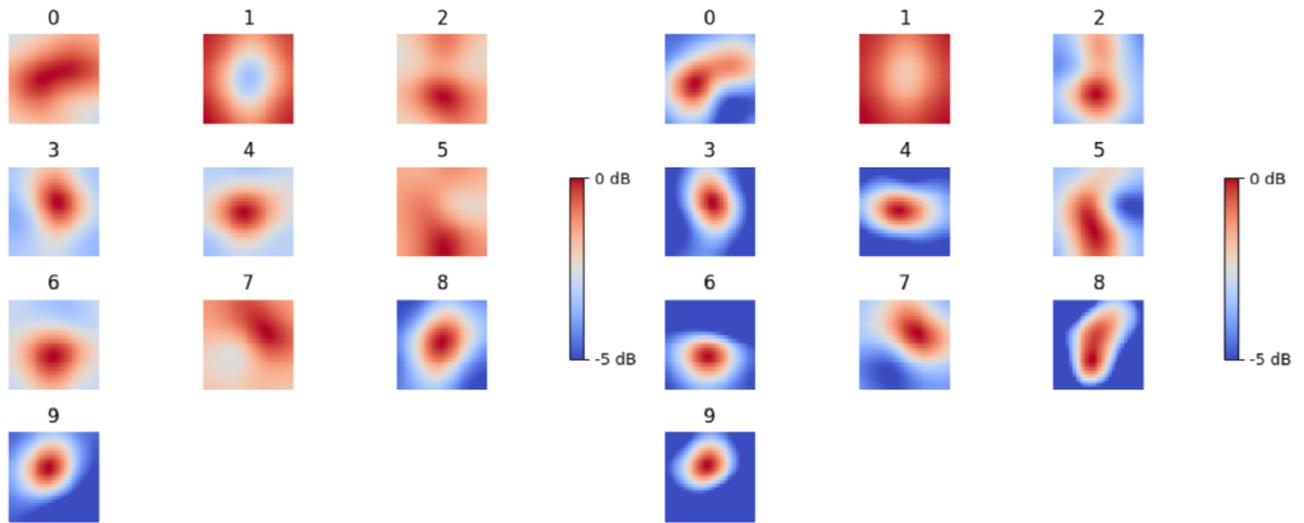
286 The availability of training data is expected to provide faster and more robust convergence
287 towards the features of interest. For each case, 60000 probing samples and 10 bubbles with standard
288 deviation of 4 pixels were used. In the case of Method 1b), a uniform random noise was first tested
289 but only lead to categorization in 5 digits categories, so a pseudo random noise obtained from the
290 inversion of a PCA was instead used generated to probe the CNN. This new noise led to decisions
291 covering the 10 categories. The discriminative information obtained in the two cases correlate
292 strongly ($r = .92$ ($SD = .01$), $df = 783$, $p < 10^{-3}$), showing that the methods’ sub-variants with or
293 without the training dataset converge toward the same masks.

294

295

296

297



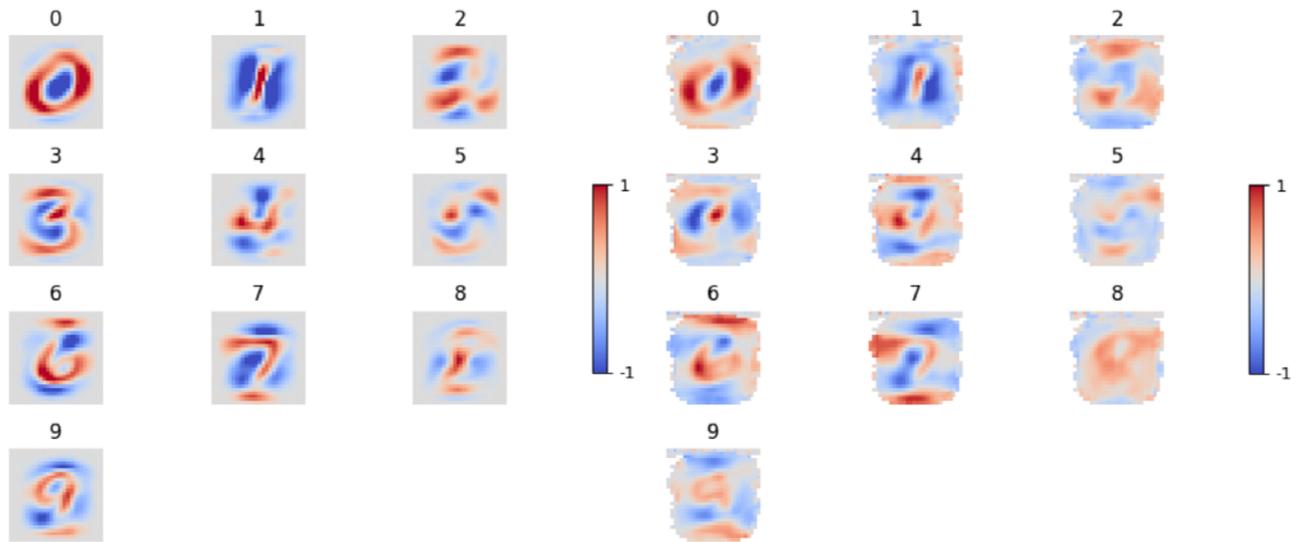
298

299

300 **Fig. 3. Discriminative features maps for a CNN classifying handwritten digits.** The maps show the
301 discriminative maps, in dB, obtained for each digit with Method 1a) with the training set available
302 (left) and Method 1b) with the training set unavailable (right). Regions in red correspond to sub-
303 spaces of the input most important for a correct classification of each digit. The maps are here
304 normalized for each digit and presented in dB ($20 \log_{10} P_i/P_{max}$) for a sake of comparability between
305 digits.

306

307 Figure 4 shows the canonical features obtained with Methods 2a) and 2b), with or without the
308 training set available. These canonical maps look different from the discriminative ones. Here, the
309 maps are weighted averages of probing samples themselves, and not low-pass bubble masks, so finer
310 details are available. As a result, and as intended with a reverse correlation approach, the canonical
311 maps are readily identifiable and visually resemble the average written digits' representation. Such
312 insight is perhaps not very surprising with simple digits, except perhaps for the 'negative' regions in
313 blue that further specify which features are canonically absent from a given digit. Again, Methods 2a)
314 and 2b) provide strongly correlated maps ($r = .67$ (SD = .18), $df = 783$, $p < 10^{-3}$). It can nevertheless
315 be noted that Method 2b) tends to focus on the center of the input space. In particular, some border
316 pixels were never associated with one or the other classification decision, leading to missing values
317 when computing d' .



319

320

321 **Fig. 4. Canonical features maps for a CNN classifying handwritten digits.** The maps show the d'
322 sensitivity index for each point of the input space, obtained with Method 2a) (left) and Method 2b)
323 (right). The red portions of the maps indicate the input features most associated with a given class.
324 They visually resemble each digit, more or less blurred. The blue portions of the maps indicate the
325 input features that are most reliably not present for a given class.

326

327 2.2. Speech vs. music

328 In this second example, we classified audio samples in a speech versus music task. We used the
329 GTZAN database composed of 132 excerpts of speech and music (Tzanetakis & Cook, 2002). The
330 database was preprocessed to create samples with a fixed duration of 5 seconds, leading to a dataset
331 of 768 samples. Those samples were randomly separated into a training set (691 excerpts) and a test
332 set (77 excerpts, 10% of the dataset).

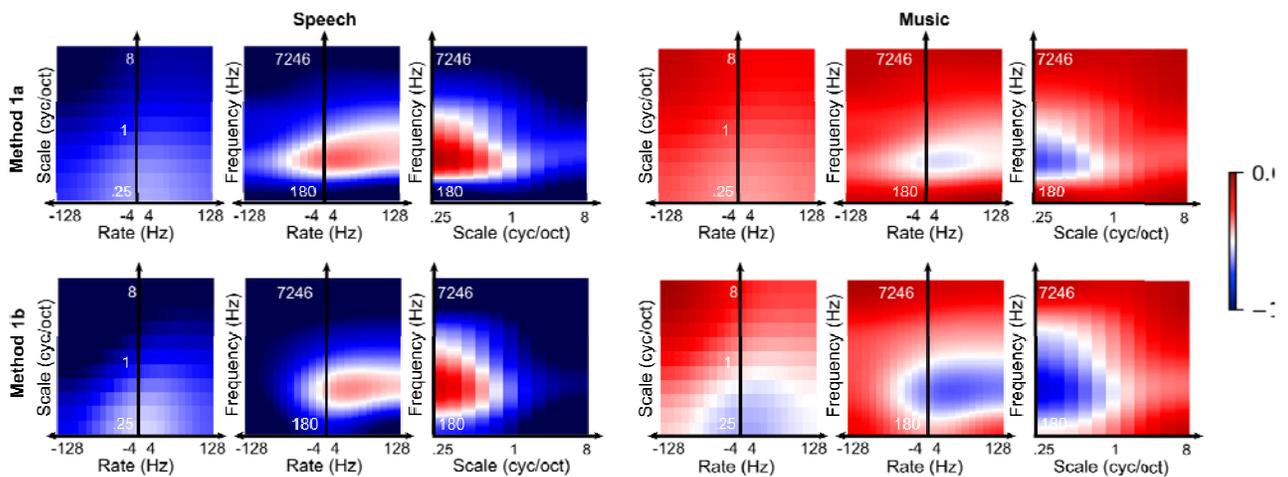
333 Following Patil et al. (2012), who performed an automatic classification of the musical timbre of
334 short audio samples, sounds were first processed by an auditory model (Chi et al., 2005). The idea is
335 to cast the input space into a representation that is interpretable in terms of auditory processing,
336 unlike the raw waveform representation. Briefly, a filterbank corresponding to cochlear tonotopy is
337 initially applied, followed by a 2-D Fourier analysis of the resulting time-frequency representation.
338 The model output thus represents temporal modulations and spectral modulations contained in the

339 input sound (Chi et al., 2005, Elliot & Theunissen, 2009). The 4-D resulting arrays, with dimensions
340 of time, frequency, scale of spectral modulations, and rate of temporal modulation, are termed here
341 Spectro-Temporal Modulation representations (STM). We averaged the time dimension over the 5s
342 of each sample. Next, we applied a PCA to reduce dimensionality (30976 dimensions in our
343 implementation: 128 frequency channels x 11 scales x 22 rates, reduced to 150 dimensions to preserve
344 98% of the variance).

345 For classification, the output of the reduced PCA was fed to a Support Vector Machine (SVM)
346 with a Radial Basis Function (RBF). All of these steps are identical to Patil et al. (2012), to which the
347 reader is referred to for further details, as the specifics of the classifier are not critical to illustrate the
348 probing method. Briefly, a grid search on the RBF was performed to determine the best set of
349 parameters and the classifier accuracy was tested with a 10-fold cross-validation. We obtained an
350 average classification accuracy, i.e. whether the classifier is classifying the STM of a sound to the
351 correct music or speech class, of 94% (SD = 6%) with the 10-fold cross-validation and 98% on the
352 test set.

353 Figure 5 shows the discriminative feature maps for the speech versus music classification task.
354 For each case, we used 691 probing samples and 30 bubbles with standard deviation of 10 Hz in the
355 frequency dimension, 6 Hz in the rate dimension, and 3 cycles/octave in the scale dimension. As the
356 task is a binary classification, the maps for speech and music are simply mirror images of each other.
357 The discriminative regions of the auditory model STM representation appear to be mostly visible in
358 the frequency dimension: speech can be best classified by looking at the input in a broad frequency
359 range around 500 Hz, corresponding roughly to the position of the first formant in speech (Peterson
360 & Barney, 1952). For the other dimensions, the classification depends on slow positive rates and low
361 scales. In other words, the difference between speech and music was in the presence of slow
362 modulations and broad spectral shapes for speech. Again, this matches prosodic and syllabic features
363 of speech, together with the broad spectral shape of formants. By construction the two maps are
364 complementary, but for music, a richness in spectrum, including high and low frequency regions,
365 associated with fine spectral details (high scales) is characteristic of musical instruments (Elhilali,
366 2019), which have been designed to go beyond the physical constraints imposed by voice production.

367 In the case of the Method 1b), with the training set unavailable, a first probing was attempted
368 with a uniform white noise but failed to provide classification decisions sampling the two categories:
369 all noises were classified as music, a perhaps amusing finding which we will not develop here. The
370 uniform white noise was thus replaced by a pseudo random noise generated in a PCA-reduced
371 representation obtained with the testing set. A whitened PCA was first applied to the testing set to
372 reduce it to 40 dimensions and a uniform gaussian white noise was generated on the 40 dimensions to
373 generate samples in the reduced space. Each random reduced sample was then transformed into the
374 original input space by applying the inverse PCA transformation. This procedure allowed to generate
375 noisy samples with distribution relevant regarding the representative set of data relevant to the
376 classification task. The information obtained in the two cases then strongly correlate ($r = .84$, $df =$
377 30974 , $p < 10^{-5}$).



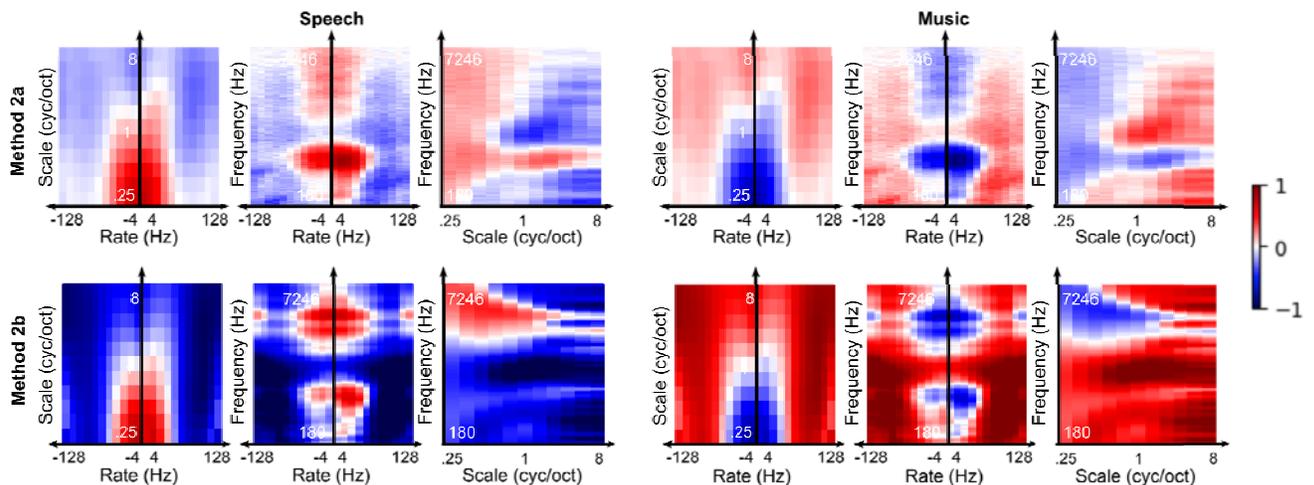
378
379 **Fig. 5. Discriminative maps for speech and music in the STM representation.** The 4-D STM
380 representations are projected in the three dimensions (frequency, scale, rates), and expressed in dB.
381 Method 1a) (left) and method 1b) (right). The complete STM matrices are available in supplementary
382 Figure S1. Method 1a) uses the training set while Method 1b) uses pseudo random noise. The red
383 regions of each map correspond to the features necessary to categorize an audio sample in the given
384 class. The blue regions correspond to less important features. As this is a binary classification task,
385 the speech and music masks are simply opposite versions of each other.

386

387 Figure 6 shows the canonical feature maps for the speech versus music classification task. Compared

389 the STM representation, or, similarly, they indicate the “average” speech and music sounds learnt by
390 the SVM. For speech, some formantic structure is visible on the frequency dimension, associated
391 with low rates typical of prosodic modulations (middle panels). These formantic regions extend to
392 higher scales (right panels), perhaps because formants are superimposed on a harmonic structure
393 during vowel sounds. Conversely, musical sounds more typically contain high modulation rates and
394 spectral scales. These observations are consistent with previous analyses of STM representations
395 (Elliott & Theunissen, 2009; Chi et al., 2005). These observations are consistent with previous
396 analyses of STM representations (Elliott & Theunissen, 2009; Chi et al., 2005). Again, the canonical
397 features observed for speech and music are complementary by construction with our method. It
398 should be noted that, as intuitively expected, canonical features depend on the acoustic characteristics
399 of speech and music, but they also depend on the task of the classifier. Probing a classifier trained to
400 discriminate speech from e.g. environmental sounds would likely provide different canonical features
401 for speech. This result may seem like a limitation of the method, but it also highlights the way an
402 automatic classifier performs a binary task. This may be an important difference to keep in mind
403 when comparing classifiers with human perception, which has to perform many concurrent tasks in
404 parallel. Yet, “opportunistic features” that depend both on sensory information and the task at hand
405 have been suggested for auditory timbre recognition, a task not unlike the one probed here (Agus et
406 al., 2019), a task not unlike the one probed here.

407



408

409 **Fig. 6. Canonical STM representations for speech and music.** *The d' sensitivity index is displayed*
410 *for projections of the 4-D representation. The complete STM matrices are illustrated in*
411 *supplementary Figure S2. The red regions of the maps indicate features most often encountered*
412 *within each category, whereas the blue regions indicate features most often not encountered within*
413 *each category.*

414 **3. Discussion**

415 **4.1 Summary**

416 The method presented in this paper used a reverse correlation framework to visualize the input
417 features discovered by an automatic classifier to reach its decisions. When the classifier is successful,
418 such features should provide insights about the structure of the input dataset. Over two examples
419 using different kinds of classifiers (a CNN and an SVM with RBF) and using different kinds of input
420 representations (2-D visual images and 1-D audio samples converted to a 4-D auditory model), we
421 illustrated how the method could highlight relevant aspects of a classifier's operation. Moreover, by
422 combining standard noise perturbation techniques with so-called bubbles (Gosselin & Shyns, 2002),
423 we showed that the probing method can be focused either on discriminative features, related to the
424 decision strategy of the classifier, or on canonical features, related to the output classes' main
425 characteristics.

426 **4.1 Benefits**

427 In the context of neuroscience and experimental psychology, there are benefits in using a
428 reverse correlation framework to interpret classifiers, as a way to complement other more specialized
429 machine-learning interpretation techniques (Zhou et al., 2016; Ribeiro et al., 2016; Petsiuk et al.,
430 2018; Borji & Lin, 2019; Xu et al., 2018).

431 First and foremost, reverse correlation is a familiar tool in the field of neuroscience and
432 experimental psychology. It has proved useful to gain insights about stimulus features relevant to
433 neural activity, at the single neuron (Eggermont et al., 1983; Neri & Levi, 2006) or network level
434 (Arnal et al., 2015; Adolphs et al., 2005; Ringach & Shapley, 2010), and to understand human
435 perceptual decisions (Gosselin & Shyns, 2001; Venezia et al., 2016). Applying it to interpret

436 classifiers amounts to translating and applying a familiar toolbox to another, conceptually similar
437 problem of characterizing a black-box system.

438 Second, the method is by design fully agnostic by design. It operates on the input space of the
439 classifier, whatever this space might be. It does not make assumptions on the classifier's architecture
440 or inner operations. Focusing on the input space rather than the classifier's architecture is especially
441 desirable in situations where the classifier is not the main interest of study, but rather, the structure of
442 the input dataset is.

443 Third, it can be applied to classifiers that have not been designed by the user, as it does not
444 even require the availability of the training dataset. Access to labeled input data is helpful in
445 improving the efficiency of the method, for instance by allowing to shape the perturbation noise, but
446 this is a mild constraint: there are no interesting situations we can think of for which both the
447 classifier and the type of data to classify would be unknown.

448 Finally, the output of the method is a visualization (with statistical evaluation if required) in
449 the input space. Such a representation should make intuitive sense to the user of the method, and the
450 features discovered can be interpreted *a posteriori* in terms of attributes of the stimuli. If the
451 representation does not make intuitive sense, then one possible benefit of the method is to help re-
452 cast the input space into a more meaningful representation, as was done here in the audio example for
453 which the waveform samples were pre-processed with an auditory model. This idea is further detailed
454 in the "Perspectives" subsection.

455 **4.2 Limitations**

456 There are also limitations associated to the use of a reverse correlation approach to interpret
457 automatic classifiers. Broadly speaking, these limitations follow those already described for reverse
458 correlation in neuroscience.

459 First, reverse correlation is inspired from the analysis of linear systems, whereas machine-
460 learning classifiers often rely on a cascade of non-linear operations to achieve computational power.
461 The issue of non-linearity is well-described already in the reverse correlation literature, and its
462 consequences have been clearly described (Theunissen et al., 2000). There are extensions to the
463 reverse correlation technique to describe lower-order non-linear interactions in the input space (Neri

464 & Heeger, 2002). Such extensions could be applied to the interpretation of classifier's features.
465 Interestingly, the reverse correlation approach bears some similarities with the "distillation" method
466 from the machine learning literature (Hinton et al., 2015). Distillation consists in mimicking the
467 behavior of a black-box classifier with an easily-interpretable classifier, such as a linear one (linear
468 SVM, etc.). Both techniques can thus be viewed as attempting to find linear approximations of a
469 classifier's operations, but their precise relationship remain to be investigated.

470 Second, the method has a number of parameters the number and size of bubbles, the space to
471 generate the probing noise with reduction dimension methods such as PCA when the training set is
472 available, which are not algorithmically constrained. In the examples above, the parameter space was
473 explored heuristically. One suggested heuristic was to try and cover the output classes in a balanced
474 manner with the probing set. However, even though statistical tests of the resulting features are
475 available, we do not provide any fitness criterion, i.e. a way to quantify the efficiency of the method
476 for a given set of parameters, for the features obtained with the method. Rather, we would argue that
477 the iterative process for parameter tuning can be part of the interpretation process since finding the
478 right probing structure provides some information on the structure of the dataset. Also, assessing
479 whether the discovered features make intuitive sense relies mostly on the knowledge and goals of the
480 user. Thus, it may not be easily formalized into a fitness criterion. If more formally defined methods
481 are needed, either from the outset or after a first exploration of the classifier with reverse correlation,
482 other classifier-specific tools exist (e.g., Zhou et al., 2016; Ribeiro et al., 2016; Petsiuk et al., 2018).

483 Third, the method implicitly assumes that there are no invariances by translation or otherwise
484 in the classifier's algorithm. With reverse correlation, each point of the input space is treated
485 independently of all others, so a feature discovered in one sub-part of the input space will not impact
486 other, perhaps similar features in other sub-parts. This assumption is obviously falsified by CNN
487 architectures, which are purposely designed to incorporate such invariances. In the CNN example
488 illustrated here with digits recognition, this limitation was circumvented by the fact that all digits in
489 the probing set were roughly spatially aligned. For the SVM on audio data, a time-averaging over the
490 time dimension achieved a similar effect. Thus, a mitigation strategy is available: a rough alignment
491 of the probing data (spatially or temporally) should be sufficient for the reverse correlation to

492 produce meaningful results. Another possible direction to address these invariance issues is to
493 generate the probing noise in an appropriate space. Using a PCA partly achieves this. Finally, using
494 another representation space with built-in invariances, e.g. by using wavelets transforms, can be
495 considered.

496 **4.3 Perspectives**

497 The probing method is technically applicable to any classifier's architecture with any kind of input
498 data. It is thus beyond the scope of this final section to list all possible use cases in the context of
499 neuroscience. We will simply provide a few suggestions, to illustrate the kind of problems that could
500 benefit from the probing method.

501 When studying perceptual decisions, one possible insight gained from interpreting a classifier
502 is the exploration of the input representation fed to the classifier. The hypothesis is that, the more
503 appropriate the representation, the more explainable the classifier should be. For instance, one could
504 assume that the massively non-linear transformations of auditory and visual information that
505 characterize perceptual systems serve to build a stimulus manifold within which perceptual
506 boundaries are approximately linear (Georgopoulos et al., 1986; Jazayeri & Movshon, 2006; Kell et
507 al., 2018). So, with the correct representation, a classifier modeling a perceptual decision process
508 should be easily interpretable, or at least more easily interpretable than if the input representation was
509 not reflecting perceptual processing. It is with this hypothesis in mind that the audio samples of the
510 example illustrated above were first processed with an auditory model. Even though there are
511 successful deep learning models operating on the raw audio waveforms (e.g. Wavenet, Oord et al.,
512 2006), it is not expected that interpreting them in terms of waveform features will be meaningful. For
513 instance, inaudible phase shifts between frequency components in the input would impact the
514 waveform representation, but should not change the classifier's decision. An auditory model, in
515 contrast, incorporates transforms inspired by the neurophysiology of the hearing system. If the
516 features extracted resemble those available to a human observer, then they should be revealed when
517 probing a classifier. In fact, the ease of interpreting a classifier feature could be a proxy to evaluate
518 an input representation's adequation to a perceptual task.

519 Another possible application is when building “ideal observer” models (Geisler, 2004). The
520 idea of an ideal observer model is to compute the best theoretical performance on a task, given a set
521 of assumptions (classically, endowing the ideal observer with unbiased decision criteria, perfect and
522 unlimited memory, and so on). This upper performance boundary is then compared to the observed
523 performance with human participants or neural recordings. When considering classification or
524 discrimination tasks, and when a formal model of the ideal observer is unavailable, it can be of
525 interest to build pseudo-ideal observer models with machine learning classifiers. The advantage of
526 our probing method is then that the classifier’s strategy can be directly compared to a reverse
527 correlation analysis of neural or psychophysical data, to ask whether the classifier and the
528 experimental observer used the same decision features.

529 Finally, the general benefits of interpreting classifiers also apply to the field of neuroscience.
530 In a broad sense, probing is intended to help an expert making sense of a classifier’s strategy. If the
531 features discovered through probing fit a theoretical model, this would reassure the expert that the
532 performance relies on reasonable principles, which is especially important in clinical applications. In
533 return, the expert’s intuition may also help improve the classifier, for instance by simplifying its input
534 representation through pre-processing, and so hopefully making it less brittle to irrelevant variations
535 in input that may have been picked up by overfitting during training (Goodfellow et al., 2015). The
536 discriminative features could be particularly useful to reduce the complexity of a classifier. Based on
537 the discriminative features map, it may be possible to select a subset of important and intelligible
538 features, which can then be used to build a more computationally efficient classifier, for very large
539 dataset and/or for real-time processing.

540 **4. Conclusions**

541 We presented a novel method to interpret machine-learning classifiers, with the aim that the method
542 should be agnostic and well-suited to applications in the neuroscience domain. Based on the reverse
543 correlation framework, the method uses stochastic perturbation of inputs to observe the classifier’s
544 output. It then visualizes, in the input space, the discriminative and canonical features discovered by
545 the classifier for each category. In theory the method can be applied to any kind of classifier,
546 including deep neural networks, support vector machines, etc. It displays the same well-established

547 benefits and limitations as reverse correlation when applied to psychophysical or neural data. Our
548 hope is that such a method can provide a simple and generic interface between neuroscientists and
549 machine-learning tools.
550

551 **5. References**

- 552 Ahumada Jr, A., & Lovell, J. (1971). Stimulus features in signal detection. *The Journal of the*
553 *Acoustical Society of America*, 49(6B), 1751-1756. <https://doi.org/10.1121/1.1912577>
- 554 Adolphs, R., Gosselin, F., Buchanan, T. W., Tranel, D., Schyns, P., & Damasio, A. R. (2005). A
555 mechanism for impaired fear recognition after amygdala damage. *Nature*, 433(7021), 68-72.
556 <https://doi.org/10.1038/nature03086>
- 557 Arnal, L. H., Flinker, A., Kleinschmidt, A., Giraud, A. L., & Poeppel, D. (2015). Human screams
558 occupy a privileged niche in the communication soundscape. *Current Biology*, 25(15), 2051-
559 2056. <https://doi.org/10.1016/j.cub.2015.06.043>
- 560 Benjamini, Y., & Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful
561 approach to multiple testing. *Journal of the Royal statistical society: series B (Methodological)*,
562 57(1), 289-300. <https://doi.org/10.1111/j.2517-6161.1995.tb02031.x>
- 563 Borji, A., & Lin, S. (2019). White Noise Analysis of Neural Networks. arXiv preprint
564 arXiv:1912.12106.
- 565 Chi, T., Ru, P., & Shamma, S. A. (2005). Multiresolution spectrotemporal analysis of complex
566 sounds. *The Journal of the Acoustical Society of America*, 118(2), 887-906.
567 <https://doi.org/10.1121/1.1945807>
- 568 Elhilali, M. (2019). Modulation Representations for Speech and Music. In *Timbre: Acoustics,*
569 *Perception, and Cognition* (pp. 335-359). Springer. [https://doi.org/10.1007/978-3-030-14832-](https://doi.org/10.1007/978-3-030-14832-4_12)
570 [4_12](https://doi.org/10.1007/978-3-030-14832-4_12)
- 571 Elliott, T. M., & Theunissen, F. E. (2009). The modulation transfer function for speech intelligibility.
572 *PLoS computational biology*, 5(3). <https://doi.org/10.1371/journal.pcbi.1000302>
- 573 Eggermont, J. J., Johannesma, P. I. M., & Aertsen, A. M. H. J. (1983). Reverse-correlation methods
574 in auditory research. *Quarterly reviews of biophysics*, 16(3), 341-414.
575 <https://doi.org/10.1017/s0033583500005126>
- 576 Deng, L. (2012). The mnist database of handwritten digit images for machine learning research [best
577 of the web]. *IEEE Signal Processing Magazine*, 29(6), 141-
578 142. <https://doi.org/10.1109/msp.2012.2211477>

- 579 Geisler, W. S. (2004). "Ideal Observer analysis," in *Visual Neurosciences*, eds L. Chalupa and J.
580 Werner (Boston, MA: MIT press), 825–837.
- 581 Georgopoulos, A. P., Schwartz, A. B., & Kettner, R. E. (1986). Neuronal population coding of
582 movement direction. *Science*, 233(4771), 1416-1419. <https://doi.org/10.1126/science.3749885>
- 583 Gosselin, F., & Schyns, P. G. (2002). RAP: A new framework for visual categorization. *Trends in*
584 *Cognitive Sciences*, 6(2), 70-77. [https://doi.org/10.1016/s1364-6613\(00\)01838-6](https://doi.org/10.1016/s1364-6613(00)01838-6)
- 585 Gosselin, F., & Schyns, P. G. (2001). Bubbles: a technique to reveal the use of information in
586 recognition tasks. *Vision research*, 41(17), 2261-2271. [https://doi.org/10.1016/s0042-](https://doi.org/10.1016/s0042-6989(01)00097-9)
587 [6989\(01\)00097-9](https://doi.org/10.1016/s0042-6989(01)00097-9)
- 588 Gosselin, F., & Schyns, P. G. (2003). Superstitious perceptions reveal properties of internal
589 representations. *Psychological science*, 14(5), 505-509. [https://doi.org/10.1111/1467-](https://doi.org/10.1111/1467-9280.03452)
590 [9280.03452](https://doi.org/10.1111/1467-9280.03452)
- 591 Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014). Explaining and harnessing adversarial examples.
592 arXiv preprint arXiv:1412.6572. <https://doi.org/10.5220/0006123702260234>
- 593 Green, D. M., & Swets, J. A. (1966). *Signal detection theory and psychophysics* (Vol. 1). New York:
594 Wiley.
- 595 Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., & Pedreschi, D. (2018). A survey
596 of methods for explaining black box models. *ACM computing surveys (CSUR)*, 51(5), 1-42.
597 <https://doi.org/10.1145/3236009>
- 598 Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. arXiv
599 preprint arXiv:1503.02531.
- 600 Jazayeri, M., & Movshon, J. A. (2006). Optimal representation of sensory information by neural
601 populations. *Nature neuroscience*, 9(5), 690-696. <https://doi.org/10.1038/nn1691>
- 602 Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects.
603 *Science*, 349(6245), 255-260. <https://doi.org/10.1126/science.aaa8415>
- 604 Kell, A. J., Yamins, D. L., Shook, E. N., Norman-Haignere, S. V., & McDermott, J. H. (2018). A
605 task-optimized neural network replicates human auditory behavior, predicts brain responses,

- 606 and reveals a cortical processing hierarchy. *Neuron*, 98(3), 630-644.
- 607 <https://doi.org/10.1016/j.neuron.2018.03.044>
- 608 King, J. R., & Dehaene, S. (2014). Characterizing the dynamics of mental representations: the
609 temporal generalization method. *Trends in cognitive sciences*, 18(4), 203-210.
- 610 <https://doi.org/10.1016/j.tics.2014.01.002>
- 611 Kriegeskorte, N., & Douglas, P. K. (2018). Cognitive computational neuroscience. *Nature*
612 *neuroscience*, 21(9), 1148-1160. <https://doi.org/10.1038/s41593-018-0210-5>
- 613 LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436-444.
- 614 <https://doi.org/10.1038/nature14539>
- 615 Lillicrap, T. P., & Kording, K. P. (2019). What does it mean to understand a neural network?. *arXiv*
616 *preprint arXiv:1907.06374*.
- 617 Maris, E., & Oostenveld, R. (2007). Nonparametric statistical testing of EEG-and MEG-data. *Journal*
618 *of neuroscience methods*, 164(1), 177-190. <https://doi.org/10.1016/j.jneumeth.2007.03.024>
- 619 Neri, P., Parker, A. J., & Blakemore, C. (1999). Probing the human stereoscopic system with reverse
620 correlation. *Nature*, 401(6754), 695-698. <https://doi.org/10.1038/44409>
- 621 Neri, P., & Heeger, D. J. (2002). Spatiotemporal mechanisms for detecting and identifying image
622 features in human vision. *Nature neuroscience*, 5(8), 812-816. <https://doi.org/10.1038/nn886>
- 623 Neri, P., & Levi, D. M. (2006). Receptive versus perceptive fields from the reverse-correlation
624 viewpoint. *Vision research*, 46(16), 2465-2474. <https://doi.org/10.1016/j.visres.2006.02.002>
- 625 Oord, A. V. D., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., ... & Kavukcuoglu, K.
626 (2016). Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*.
- 627 Paquette, S., Takerkart, S., Saget, S., Peretz, I., & Belin, P. (2018). Cross-classification of musical
628 and vocal emotions in the auditory cortex. *Ann. NY Acad. Sci*, 1423, 329-337.
- 629 <https://doi.org/10.1111/nyas.13666>
- 630 Patil, K., Pressnitzer, D., Shamma, S., & Elhilali, M. (2012). Music in our ears: the biological bases
631 of musical timbre perception. *PLoS computational biology*, 8(11).
- 632 <https://doi.org/10.1371/journal.pcbi.1002759>

- 633 Peterson, G. E., & Barney, H. L. (1952). Control methods used in a study of the vowels. The Journal
634 of the acoustical society of America, 24(2), 175-184. <https://doi.org/10.1121/1.1906875>
- 635 Petsiuk, V., Das, A., & Saenko, K. (2018). Rise: Randomized input sampling for explanation of
636 black-box models. arXiv preprint arXiv:1806.07421.
- 637 Richards, B. A., Lillicrap, T. P., Beaudoin, P., Bengio, Y., Bogacz, R., Christensen, A., ... & Gillon,
638 C. J. (2019). A deep learning framework for neuroscience. Nature neuroscience, 22(11), 1761-
639 1770. <https://doi.org/10.1038/s41593-019-0520-2>
- 640 Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). " Why should i trust you?" Explaining the
641 predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD international
642 conference on knowledge discovery and data mining (pp. 1135-1144).
643 <https://doi.org/10.1145/2939672.2939778>
- 644 Ringach, D., and Shapley, R. (2004). Reverse correlation in neurophysiology. Cogn. Sci. 28, 147–
645 166. doi: 10.1207/s15516709cog2802_2
- 646 Rosch, E. (1983). Prototype classification and logical classification: The two systems. New trends in
647 conceptual representation: Challenges to Piaget’s theory, 73-86.
- 648 Theunissen, F. E., Sen, K., & Doupe, A. J. (2000). Spectral-temporal receptive fields of nonlinear
649 auditory neurons obtained using natural sounds. Journal of Neuroscience, 20(6), 2315-2331.
650 <https://doi.org/10.1523/jneurosci.20-06-02315.2000>
- 651 Tzanetakis, G., & Cook, P. (2002). Musical genre classification of audio signals. IEEE Transactions
652 on speech and audio processing, 10(5), 293-302. <https://doi.org/10.1109/tsa.2002.800560>
- 653 Venezia, J. H., Hickok, G., & Richards, V. M. (2016). Auditory “bubbles”: Efficient classification of
654 the spectrotemporal modulations essential for speech intelligibility. The Journal of the
655 Acoustical Society of America, 140(2), 1072-1088. <https://doi.org/10.1121/1.4960544>
- 656 Wang, D., Khosla, A., Gargeya, R., Irshad, H., & Beck, A. H. (2016). Deep learning for identifying
657 metastatic breast cancer. arXiv preprint arXiv:1606.05718.
- 658 Wiener, N. (1966). Nonlinear problems in random theory. Nonlinear Problems in Random Theory,
659 by Norbert Wiener, pp. 142. ISBN 0-262-73012-X. Cambridge, Massachusetts, USA: The MIT
660 Press, August 1966.(Paper), 142. <https://doi.org/10.1063/1.3060939>

661 Xu, T., Garrod, O., Scholte, S. H., Ince, R., & Schyns, P. G. (2018). Using psychophysical methods
662 to understand mechanisms of face identification in a deep neural network. In Proceedings of the
663 IEEE Conference on Computer Vision and Pattern Recognition Workshops (pp. 1976-1984).
664 <https://doi.org/10.1109/cvprw.2018.00266>

665 Zihni, E., Madai, V. I., Livne, M., Galinovic, I., Khalil, A. A., Fiebach, J. B., & Frey, D. (2020).
666 Opening the black box of artificial intelligence for clinical decision support: A study predicting
667 stroke outcome. Plos one, 15(4), e0231166. <https://doi.org/10.1371/journal.pone.0231166>

668 Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., & Torralba, A. (2016). Learning deep features for
669 discriminative localization. In Proceedings of the IEEE conference on computer vision and
670 pattern recognition (pp. 2921-2929). <https://doi.org/10.1109/cvpr.2016.319>

671

672

673 **Acknowledgements:** *Research supported by grants ANR-16-CONV-0002 (ILCB), ANR-11-LABX-*
674 *0036 (BLRI) and the Excellence Initiative of Aix-Marseille University (A*MIDEX) (ET), ANR-10-*
675 *LABX-0087 IEC and ANR-10-IDEX-0001-02 PSL (DP). TA was supported by the Human Frontier*
676 *Science Program (LT000362/2018-L).*

677

678 **Supplementary Materials**

679 **Figure S1.** Complete Spectro-Temporal Modulation representations discriminative maps, in dB, for
680 speech vs. music classification task.

