Living Reviews in Solar Physics (2023) 20:4
https://doi.org/10.1007/s41116-023-00038-x

REVIEW ARTICLE

Machine learning in solar physics

Andrés Asensio Ramos[1,2] · Mark C. M. Cheung[3] · Iulia Chifu[4] · Ricardo Gafeira[5]

Received: 17 January 2023 / Accepted: 13 June 2023 / Published online: 13 July 2023

Abstract

The application of machine learning in solar physics has the potential to greatly enhance our understanding of the complex processes that take place in the atmosphere of the Sun. By using techniques such as deep learning, we are now in the position to analyze large amounts of data from solar observations and identify patterns and trends that may not have been apparent using traditional methods. This can help us improve our understanding of explosive events like solar flares, which can have a strong effect on the Earth environment. Predicting hazardous events on Earth becomes crucial for our technological society. Machine learning can also improve our understanding of the inner workings of the sun itself by allowing us to go deeper into the data and to propose more complex models to explain them. Additionally, the use of machine learning can help to automate the analysis of solar data, reducing the need for manual labor and increasing the efficiency of research in this field.

Keywords Sun: general · Photosphere · Chromosphere · Corona · Activity · Methods: data analysis · Statistical · Techniques: image processing

✉ Andrés Asensio Ramos
aasensio@iac.es

Mark C. M. Cheung
mark.cheung@csiro.au

Iulia Chifu
iulia.chifu@uni-goettingen.de

Ricardo Gafeira
gafeira@uc.pt

[1] Instituto de Astrofísica de Canarias, 38205 La Laguna, Tenerife, Spain

[2] Departamento de Astrofísica, Universidad de La Laguna, 38205 La Laguna, Tenerife, Spain

[3] CSIRO, Space & Astronomy, PO Box 76, Epping, NSW 1710, Australia

[4] Institute for Astrophysics and Geophysics, University of Göttingen, Friedrich-Hund-Platz 1, 37077 Göttingen, Germany

[5] Instituto de Astrofísica e Ciências do Espaço, Departamento de Física, Universidade de Coimbra, OGAUC, Rua do Observatório s/n, 3040-004 Coimbra, Portugal

Abbreviations

AANN	Autoassociative neural network
AE	Autoencoder
ANN	Artificial neural network
CBR	Coordinate-based representation
CNN	Convolutional neural network
CS	Compressed sensing
DBSCAN	Density-based spatial clustering of applications with noise
DDPM	Denoising diffusion probabilistic model
DNN	Deep neural network
ELU	Exponential linear unit
FCM	Fuzzy C-means (FCM)
FCN	Fully connected network
GAN	Generative adversarial network
GD	Gradient descent
GPU	Graphical processing unit
INN	Invertible neural network
INR	Implicit neural representation
LCT	Local correlation tracking
LLE	Locally linear embedding
LSTM	Long short-term memory
ML	Machine learning
MLP	Multilayer perceptron
MOMFBD	Multi-object multi-frame blind deconvolution
NeF	Neural field
NF	Normalizing flow
PCA	Principal component analysis
PCM	Possibilistic C-means
RBF	Radial basis function
ReLU	Rectified linear unit
RL	Reinforcement learning
RNN	Recurrent neural network
RVM	Relevance vector machine
SGD	Stochastic gradient descent
SOM	Self-organizing map
SPoCA	Spatial possibilistic clustering algorithm
SVD	Singular value decomposition
SVM	Support vector machine
t-SNE	Student-t Stochastic Neighbor Embedding
TL	Transfer learning
TPU	Tensor processing unit
VAE	Variational autoencoder

Contents

1 Introduction

Astrophysics, and solar physics in particular, is an observational science in which we cannot change the experimental conditions, we simply observe. Therefore, the only way of learning is by confronting observations with state-of-the-art theoretical modeling. The models are then tuned until the observations are explained and conclusions are drawn from this comparison. As a consequence, our understanding of the universe is based on the availability of data.

The amount of data available until the final decades of the 20th century was very reduced and could easily be stored in relatively standard storage media, from notebooks, books or small computing centers. The scarcity of data forced researchers to use strongly informed generative models based on our theoretical advances, with a heavy use of inductive biases.[1] This is necessary to allow generalization of the conclusions. From a probabilistic point of view, generative models are a way to describe the joint probability $p(x, y)$, where x are the observations and y are the parameters of the model. The ever-increasing quality of the observations allowed researchers to propose more and more complex physical scenarios to be compared with observations.

[1] Set of explicit or implicit assumptions made by an algorithm to properly generalize what is learned from a finite set of observation into a general model.

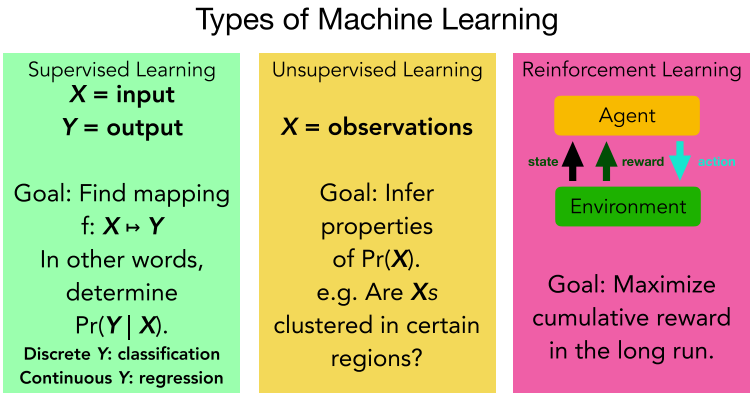

Fig. 1 Categories of machine learning (ML): supervised, unsupervised and reinforcement learning. Supervised learning and unsupervised learning have deep roots in the field of statistical learning (Hastie et al. 2009), while reinforcement learning has strong connections with control theory

Solar physics is rapidly entering into the *big data* era, an era dominated by the availability of data, which cannot fit in current computers and have to be stored, in many cases, in a distributed manner. The storage and access to this data is a technological challenge and has not been completely solved in our field. For example, access to the curated Solar Dynamics Observatory dataset of Galvez et al. (2019a) implies downloading 6.5 TB of data. Unless a dedicated connection is used, the transfer and local storage of all this data is hard. On the other hand, the estimated amount of data in an excellent observing day for the multi-instrument telescopes Daniel K. Inouye Solar Telescope (DKIST) and European Solar Telescope (EST) easily reaches the PB regime.

Having access to large datasets is not very useful unless one can extract relevant information out from them. Such large datasets have made it impossible to have people looking at the data and search for interesting correlations. For this reason, the field of machine learning (ML) has recently bloomed as a very attractive way of using our computing power to extract conclusions from data. The access to a large amount of data is opening up the possibility of using discriminative models to directly learn from the data. From a probabilistic point of view, these models try to directly model the distribution $p(y|x)$. They do not put emphasis on understanding the generation process of the data x, but on directly inferring properties from observations. The machine learning revolution that we are witnessing in solar physics is fundamentally based on discriminative models. The large databases that we have available are allowing us to directly learn from data, or use data for speeding up certain complex operations.

Machine learning methods are often divided into three main classes: supervised, unsupervised, and reinforcement learning (see Fig. 1). Supervised and unsupervised learning have deep roots in the field of statistics known as statistical learning (see the textbook by Hastie et al. 2009), which is concerned with model fitting, parameter estimation, and learning about the structure of data. Reinforcement learning is, however, strongly based on control theory (e.g., Nise 2000). The term ML became

popular in the era of Big Data. To scale statistical learning algorithms to effectively utilize large data sets, new algorithms and the appropriate software and hardware stack were developed in tandem. For instance, the development of general purpose graphical processing units (GPUs) accelerated the development of computer vision techniques. This in turn drove the development of GPU hardware with higher throughput, and the development of ML programming frameworks, such as Tensorflow (Abadi et al. 2015), PyTorch (Paszke et al. 2019) or JAX (Bradbury et al. 2018). In turn, these developments facilitated the development of models with greater expressivity and applicability across scientific and engineering domains.

In the following sections, we briefly introduce the goals of supervised, unsupervised and reinforcement learning. Most applications of ML in solar physics pertains to the first two classes. The applications of reinforcement learning in solar physics have received little attention but it can bring substantial improvements in observational planning and other complex control tasks like adaptive optics. Functional optimization is used in all three classes of ML, so we will discuss optimization too.

1.1 Supervised learning

Supervised learning is the task of learning a mapping between inputs (the collection thereof is often denoted by $\mathbf{X}$) and outputs (denoted by $\mathbf{Y}$; also called targets) for which examples of input–output pairs are available. From a probabilistic perspective, the goal of supervised learning is to model the conditional distribution $p(y|x)$.

Supervised learning is especially suited for the physical sciences because it can be used to infer parameters $\mathbf{Y}$ from the inputs $\mathbf{X}$. To illustrate with a concrete example from solar physics, suppose we have measured the Stokes IQUV parameters of a magnetically sensitive line over a region of interest on the Sun. The goal here is to infer the physical properties of the plasma producing the radiation. This process is commonly known as *inversion*.

For each spatial location on the Sun, the input data is $\mathbf{x} = (\mathrm{I}, \mathrm{Q}, \mathrm{U}, \mathrm{V})$, where each Stokes parameter is a function of the wavelength (so x is a tuple with size $N = 4 \times N_\lambda$, with N_λ being the number of measured wavelength points). It is convenient to collect the set of Stokes profiles measured at all M locations of interest as a matrix $\mathbf{X} = (\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_i, ..., \mathbf{x}_M)^T$, i.e. each row of $\mathbf{X}$ is a sample of input data. In Fig. 2, each sample (row) of input data $\mathbf{X}$ is schematically denoted as a point residing in a subspace. In ML parlance, this is called the input *feature space*, denoted by X.

In our example, the physical parameters of interest include the magnetic field strength B, the orientation of the magnetic field vector (inclination and azimuth, ϕ, θ respectively), the ambient plasma temperature (T), etc. For each spatial location on the solar surface, we have $y = (B, \phi, \theta, T, ...)$. The traditional approach to inferring these physical parameters is to perform an iterative inversion with the help of a physics-based forward model. Except under special conditions (generally not valid on the Sun), the forward model is a nonlinear radiative transfer calculation. The forward model $g : Y \longrightarrow X$, often known in advance and based on physical

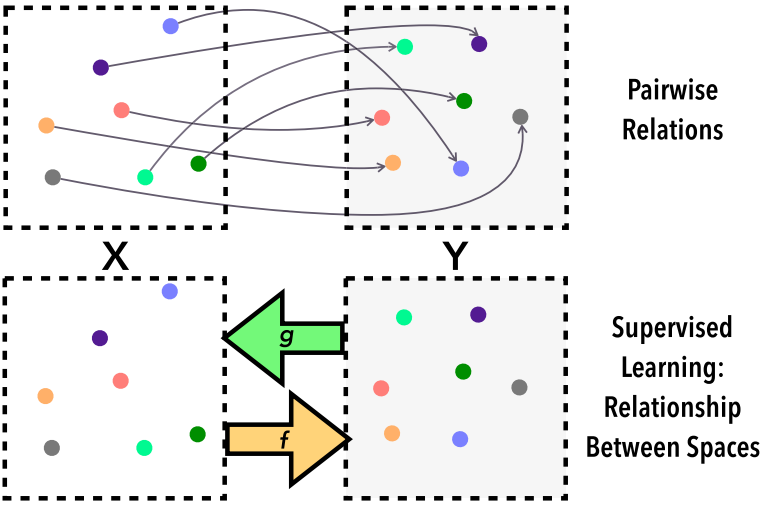

Fig. 2 Schematic representation of supervised learning, with the feature space X and the target space Y. The aim is to define or learn mappings f and g between the two spaces by taking advantage of the information encoded in the pairwise relations were known in advance for a specific sample of the two spaces

arguments, allows us to compute the predicted Stokes profiles $x \in X$ for any $y \in Y$. In Fig. 2, g is denoted by the arrow going from subspaces Y to X.

An iterative inversion begins with an initial estimate of y. The physics model is used to compute $x_{\text{pred}} = g(y)$, which is compared to the target (observed) x with a chosen penalty function. A common function is the mean-squared error, which is motivated by the assumption of Gaussian noise with diagonal covariance in the observations. In general, a suitable probabilistic approach can take into account other sources of noise or regularization (see Sect. 4.3). The physical parameters are adjusted at each iteration to minimize the discrepancy between predicted and observed data. To guide the updates of y in a way to reduce the penalty function, the curvature of the penalty function with respect to the parameters is often used (see del Toro Iniesta and Ruiz Cobo 2016). The forward model g is used to generate pairs of (y, x_{pred}) until a pair is found such that $x_{\text{pred}} \approx x$ to some tolerance. This entire procedure is then repeated for each $x \in \mathbf{X}$. In this approach, the inversion provides a pairwise mapping between each sample pair. Although it works, it has two major drawbacks. First of all, it is inefficient because the inference of a pair $x_i \longrightarrow y_i$ is performed completely independently of other pairs in the data. Secondly, the approach does not let us efficiently compute how a perturbation of the input impacts the output.

Whereas the traditional iterative optimization procedure gives a mapping between *individual pairs* in the input and output feature spaces, supervised learning aims to provide the mapping between the two spaces, i.e. $f : X \longrightarrow Y$ (see Fig. 2). Supervised learning does so by using the data $\mathbf{X}$ and $\mathbf{Y}$ globally (not individual rows) to fit a model approximating f. This is usually posed as an optimization problem of the following form:

$$f^{\#} = \text{argmin } L(f(\mathbf{X}), \mathbf{Y}). \tag{1}$$

In other words, find an optimal function f such that the loss function L (which compares the predicted and observed values) is minimized.[2] Usually, the function f is expressed in terms of parameters (e.g., weights and biases in a neural network, as explained in Sect. 5), and fitting is performed to adjust the parameters of f to minimize L. This step is called *model training*.

Note that the forward model g is not needed when doing model training. As long as pairwise data linking X and Y is available, the forward model is not a prerequisite for supervised learning. In fact, in many applications, both f and g are unknown prior to fitting. In cases where g is known (e.g., our Stokes spectropolarimetry example) and is used to generate observables $\mathbf{X}$ from $\mathbf{Y}$, supervised learning amounts to learning the inverse mapping $f = g^{-1}$.

Having presented the goal of supervised learning, we introduce some necessary nomenclature to aid discussion throughout this review article.

1.1.1 Classification versus regression

In a supervised learning setting, the target variable $\mathbf{Y}$ may be a continuous variable or may be discrete (e.g., the set of non-negative integers $\mathcal{Z}^+$). These two types of problems are called regression and classification, respectively. Both regression and classification can be tuned to deal with the same problem. For instance, let us consider the problem of flare prediction. The following is a *classification problem*: predict whether the Sun will produce a flare of class M or higher. The ground truth values are binary (Yes or No). Another way to pose a similar question involves a *regression problem*: predict the peak X-ray flux within the next 24 h. This is a regression problem since the peak X-ray flux is a continuous variable. Classification problems are often simpler to solve using machine learning.

1.1.2 Data partitioning

Data partitioning or data splitting is the act of splitting $\mathbf{X}$ and $\mathbf{Y}$ into training, testing and validation sets. Members of the training set $(\mathbf{X}_{\text{train}}, \mathbf{Y}_{\text{train}})$ are used at time of model fitting, and the loss function gives a scalar computed over this set. After training, the loss function is evaluated over the test set $(\mathbf{X}_{\text{test}}, \mathbf{Y}_{\text{test}})$ and is compared with the value for the training set. The case $L_{\text{test}} > L_{\text{train}}$ is a sign of possible overfitting, so that the model is "memorizing" the training set and not generalizing correctly.

The test set is reserved for the evaluation of the performance of the final trained model(s) on certain chosen metrics. If the model contains hyperparameters (for instance, the width of a densely connected layer), a *validation set* is carved out from the training set for exactly this purpose (Russell and Norvig 2009). This further partitioning of the (non-test) data into a training and validation set allows one to

[2] A simple widespread loss function is the mean squared error (MSE) between predicted and measured y, which assumes that the residual between the predictions and the measurements follows a Gaussian distribution.

evaluate the performance of the model during training time without building bias toward fitting the test set.

Data partitioning is one of the most important decisions in a machine learning project. Depending on the goal, the partitioning strategy will vary. For instance, should flares from the same AR be in both test and training sets? The appropriate partitioning strategy is highly dependent on the nature of the scientific/engineering objective as well as the nature of the underlying system under consideration. To successfully apply ML to solar physics problems, it is desirable not to apply it blindly but take into account the existing knowledge.

1.1.3 Encoders and decoders

In ML parlance, the forward model g is sometimes called the *encoder*, and the optimal function f is known as the *decoder*. The two functions in series, i.e., $g(f(x)$ or $f(g(y))$, is called autoencoder. As will be discussed later, autoencoders are useful in a number of applications, including data denoising. They are useful, as well, when both g and f are not known a priori. However, when one of them is known (e.g., a physics model for g), autoencoders can be used to directly learn the other mapping from data. This way of combining machine learning and physical information turns out to be extremely powerful.

1.2 Unsupervised learning

Unsupervised learning is the task of discovering patterns in the data $\mathbf{X}$. Unlike supervised learning, this task does not require matching target values $\mathbf{Y}$. In other words, unsupervised learning is about characterizing the structure of the probability density function $P(\mathbf{X})$. For instance, a common question addressed using unsupervised learning is in regard to clustering of data points. Unless the components of $\mathbf{x}$ are independent and identically distributed (i.i.d.), $P(\mathbf{X})$ will have local minima and maxima, with the latter indicating clustering of data in parts of the input space.

Unsupervised learning can be very useful for a global understanding of the observations. From a probabilistic perspective, the task is to model the prior distribution of observations $p(x)$. As such, unsupervised models do not make use of any labeling, just purely observations. As an illustration, we return to the example application of spectropolarimetry. Due to the physics (e.g., Zeeman or Hanle effects), the values of the Stokes IQUV parameters emergent from the Sun's atmosphere are not independent across the spectral (i.e., wavelength) dimension. Furthermore, the four Stokes parameters are correlated with each other as a consequence of the laws of physics (e.g., $I_\lambda^2 \geq V_\lambda^2 + U_\lambda^2 + Q_\lambda^2$ where the subscript denotes the monochromatic intensity at wavelength λ). Even if a spectrogram has N_λ wavelength positions, the number of degrees of freedom in a Stokes IQUV measurement is significantly less than $4N_\lambda$. If we knew the exact details of the underlying plasma (e.g., turbulence properties, whether there is subpixel structure), the number of degrees of freedom would be known a priori. In the absence of such insights, unsupervised learning can help with dimensionality reduction, by automatically finding the correlations and exploiting them.

1.3 Reinforcement learning

Although reinforcement learning (RL) often involves techniques used in supervised and unsupervised learning, it is considered a separate field of ML. The goal of RL is to explore how autonomous agents (e.g., a robot) interact with an environment (e.g., the world), and how to effectively train such agents to achieve desired objectives (Sutton and Barto 1998). In an RL setting, an agent has an internal state. The agent is exposed to input (stimuli) from its environment (e.g., an image of the scene surrounding the agent). Based on policies available to the agent, it carries out an action which can change the agent's state and its environment. The objective of the agent is to maximize its cumulative rewards, as determined by a suitable reward function. RL has found extensive applications in robotics, the automotive industry, and gaming. As of writing, the authors are aware of a single application of RL in solar physics, that is discussed in Sect. 7.3.6. However, we envisage RL will eventually be used for complex solar physics-related applications, such as observation planning or better adaptive optics systems (Nousiainen et al. 2022).

2 Some ideas about dimensionality

Data living in very high dimensions present difficulties when analyzing and understanding their statistical properties. The efficiency of typical statistical and computational methods usually degrades very fast when the dimensionality of the

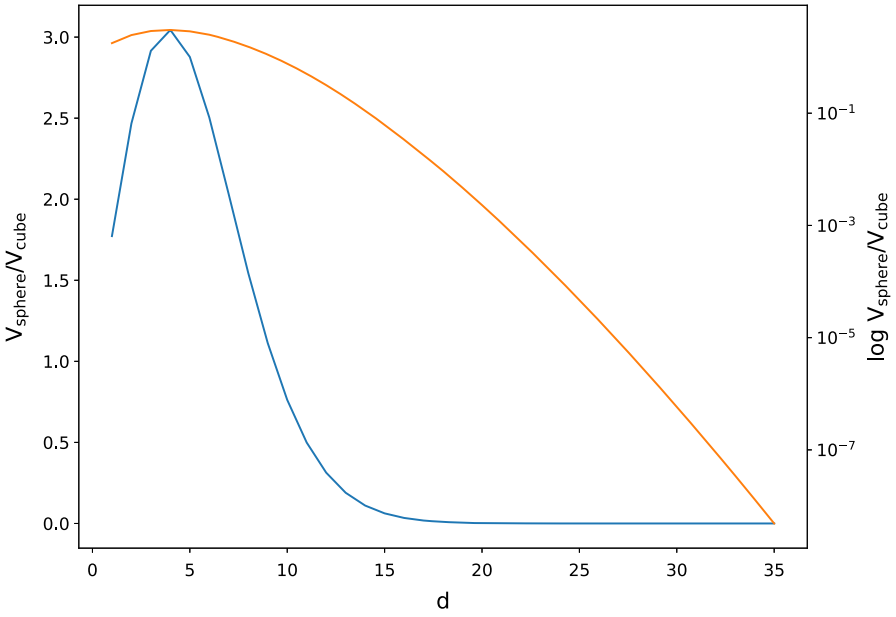

Fig. 3 Curse of dimensionality for an Euclidean space of dimension d. This figure shows the volume ratio between an hypersphere of radius r and the hypercube in which the hypersphere is circumscribed. It shows that the volume resides in the external parts of the space. Both linear and logarithmic scales are shown for clarity

problem increases, thus making the analysis of the observed data cumbersome or, sometimes, unfeasible. This fact is often referred to as the *curse of dimensionality*. The fundamental reason for that lies in the fact that, for high dimensional spaces, almost all the volume of the space tends to accumulate in the borders of the space. We can easily visualize this in an Euclidean space of dimension d. The ratio between the volume of the hypersphere of radius r centered at the origin and that of the hypercube of side length $2r$ centered at the origin in which the hypersphere is inscribed is given by:

$$\frac{V_{\text{hypersphere}}}{V_{\text{hypercube}}} = \frac{\pi^d}{d2^{d-1}\Gamma(d/2)}. \tag{2}$$

As shown in Fig. 3, the ratio exponentially goes down to zero when d increases, so that the volume very quickly accumulates on the borders of the hypercube. As a consequence, any sampling of a high-dimensional space rapidly becomes useless. The enormous success of ML in recent years, thanks to the deep learning revolution, is rooted in the ability of deep learning to overcome the curse of dimensionality.

The advent of computers has permitted us to face the analysis of increasingly complex data. These data usually exhibit an intricate behavior, and in order to understand the underlying physics that produces such effects, we have been forced to develop very complicated models. Ideally, these models have to be based on physical grounds, but there seems to be no way of knowing in advance how complicated this model has to be to correctly reproduce the observed behavior. Despite their inherent complexity, the analysis of large data sets, such as those produced by modern instrumentation, indicates that not all measured data points are equally relevant for the understanding of the underlying phenomena (one of the simplest example is the limited information carried out by spectral points in the continuum versus spectral points sampling spectral lines). In other words, it is clear that the reason why many simplified physical models are successful in reproducing a large amount of observations is because the data itself is not truly high dimensional. Based on this premise, it makes sense to develop and apply methods that are capable of reducing the dimensionality of the observed data sets while still preserving their fundamental properties. Mathematically, the idea is that while the original data may have a very large dimensionality, they are in fact confined to a small manifold of that high-dimensional space. In this case, we can consider that the data "lives" in a subspace of low dimension (the so-called intrinsic dimension) that is embedded in the high-dimensional space. This lower dimension manifold is not simple to describe in general, simply because it is highly nonlinear and unknown.

Instead of fully characterizing the manifold, the simpler task of estimating its intrinsic dimensionality is of interest to understand the complexity of the models used to extract information. Additionally, it is a check that the machine learning method used to analyze the data is able to overcome the curse of dimensionality. One of the simplest examples is the one in which the data consists of spectral, or more in general, Stokes profiles, that encode the polarization state of light. A deep analysis of objects of much larger dimensionality, like images of the Sun, has never been carried out. As shown later in this review, some of the recent deep learning methods have

been successfully applied to solar images. This implicitly demonstrates that solar images also lie in a manifold of reduced dimensionality when compared with the potential dimensionality of all possible images.

The estimation of the dimensionality of the manifold of Stokes profiles for photospheric lines was pursued by Asensio Ramos et al. (2007c). They used an estimator of the dimensionality based on the maximum likelihood principle previously developed by Levina and Bickel (2005). When applied to Fe I lines in the visible and the infrared, they reached the following conclusions. First, the dimensionality of the infrared lines is slightly larger than those in the visible, thus suggesting that there is more variability in the Stokes profiles in the infrared. This is probably a consequence of the fact that Doppler shifts and Zeeman splittings in the infrared are slightly larger than in the visible, producing more deformations in the line profile. Second, the dimensionality of circular polarization is larger than that of Stokes I, a consequence of the fact that Stokes V is much more sensitive to variations in the magnetic field than Stokes I. Finally, they quantitatively proved the idea that adding more spectral lines increases the amount of information available (see, e.g., Semel 1981; Socas-Navarro 2004). Adding more spectral lines monotonically increases the dimensionality of the manifold but clearly not in proportion to the number of added spectral lines. There is a lot of redundant information already encoded in all spectral lines and only small details can be better seen in one spectral line or another.

Given that Stokes profiles sampled at N_λ wavelength points are demonstrated to be lying in a manifold of dimension $d \ll N_\lambda$, it makes sense to exploit this property for different purposes. The two more obvious ones are denoising and compression. Uncorrelated additive noise typically assumed to be present in spectropolarimetric observations, spans the full space of N_λ dimensions. It is advantageous to find a suitable representation in which the signal is separated from the noise by exploiting the fact that the signal lies in a manifold of reduced dimensionality. Similarly, representing the spectropolarimetric data with a reduced set of numbers leads to an important compression factor, which turns out to be important for data storage and transfer via telemetry. The methods presented in the following sections have been successfully used in solar physics for these purposes.

3 Linear models: unsupervised

The availability of data to analyze, their large sizes, and the difficulties in extracting physical information from the observations has led to the widespread application of unsupervised machine learning methods. These methods allow us to extract relevant information from the observations directly, typically focusing on the regularity that can be explained in a posteriori. Clustering or classification is perhaps one of the most obvious tasks of unsupervised machine learning methods. We start first by describing linear models for unsupervised ML. These methods, despite their limitations, have been extremely successful in science and, specifically, in solar physics.

3.1 Principal component analysis

Principal Components Analysis (PCA; Loève 1955), also known as the Karhunen–Loève transformation, is perhaps one of the most used algorithms in multivariate statistics[3]. Briefly, its main use is to obtain an orthogonal basis on which the data can be efficiently expressed. This basis has the property that the largest amount of variance is explained with the least number of basis vectors. It is useful to reduce the dimensionality of data sets that depend on a very large number of parameters and one of its most straightforward applications is denoising.

PCA can be seen as the solution to a linear regression problem in which both the weights and the basis functions are inferred from the data. Since this is an ill-defined problem, PCA imposes the additional restriction of orthogonality for the basis functions. Summarizing, PCA is a way to decompose any observed signal as a weighted sum of empirical orthogonal basis functions. It has been extensively used in the field of spectropolarimetry for denoising and, in general, dimensionality reduction purposes.

Let us assume that the wavelength variation of the Stokes profiles of a particular spectral line is described by the quantity S_{ij}. The index i represents the wavelength position while the index $j = \{\mathrm{I}, \mathrm{Q}, \mathrm{U}, \mathrm{V}\}$ labels the Stokes parameter. Each Stokes parameter is a vector of length N_λ, corresponding to the number of sampled wavelength points. Assume that the spectral line is observed in many locations in the field of view, that we term N_{obs}. By stacking all observations, one can build the observation matrix $\mathbf{O}$, which is of size $N_{\mathrm{obs}} \times N_\lambda$. As well, all Stokes parameters can be stacked together and one ends up with a matrix of size $N_{\mathrm{obs}} \times 4N_\lambda$. The principal components can then be found by computing the eigenvectors of this matrix of observations. This means that the PCA procedure reduces to the diagonalization of the matrix $\mathbf{O}$. Since we often have that $N_{\mathrm{obs}} \gg N_\lambda$, this matrix is not square and one needs to use the singular value decomposition (SVD; see, e.g., Press et al. 1986) to diagonalize O and compute its singular vectors. The SVD decomposition reads as follows:

$$\mathbf{O} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\star}, \tag{3}$$

where $\mathbf{U}$ is an $N_{\mathrm{obs}} \times N_{\mathrm{obs}}$ orthogonal matrix with the left singular vectors in columns while $\mathbf{V}$ is an $N_\lambda \times N_\lambda$ orthogonal matrix with the right singular vectors in columns. $\boldsymbol{\Sigma}$ is a diagonal matrix with the singular values on the diagonal. The real power of PCA lies in the fact that one can truncate the previous decomposition by only leaving r singular values equal to their original value and setting the rest to zero. This way, one gets $\tilde{\mathbf{O}}$, a reconstruction of the original matrix constrained to have $\mathrm{rank}(\tilde{\mathbf{O}}) = r$.

Although carrying out the PCA decomposition using the $\mathbf{O}$ matrix is possible, it is often much more efficient from a computational point of view to compute the singular vectors of the correlation or the cross-product matrices. With the use of simple algebra for the case of the correlation matrix, $\mathbf{X} = \mathbf{O}^{\dagger}\mathbf{O}$, one can verify that:

[3] PCA is available on the `scikit-learn` Python package.

$$\mathbf{O}^\dagger\mathbf{O} = \mathbf{V}\boldsymbol{\Sigma}^\star\mathbf{U}^\star\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\star = \mathbf{V}\left(\boldsymbol{\Sigma}^\star\boldsymbol{\Sigma}\right)\mathbf{V}^\star, \tag{4}$$

so that the right singular vectors of the $\mathbf{X}$ matrix are equal to the right singular vectors of the observation matrix. Likewise, the singular values of the correlation matrix are those of the original matrix but squared. The advantage of this approach is that the matrix $\mathbf{X}$ has size $N_\lambda \times N_\lambda$, so that the diagonalization becomes more computationally efficient if $N_\lambda \ll N_{\mathrm{obs}}$.

On the contrary, in cases in which $N_\lambda \gg N_{\mathrm{obs}}$, one can use a similar approach but using the cross-product matrix, $\mathbf{X}' = \mathbf{O}\mathbf{O}^\dagger$. In such case, the left singular vectors of the matrix $\mathbf{O}$ are obtained. It is important to remark that both descriptions are dual and they are completely equivalent, so one should choose the one that provides the most efficient computation.

Once an orthogonal basis is found, one can reconstruct the original matrix by just computing the projection along each direction and multiplying each projection by the orthogonal basis:

$$\mathbf{O} = (\mathbf{O}\mathbf{V})\mathbf{V}^\star. \tag{5}$$

3.1.1 Denoising

Given that the largest amount of variance of the input dataset is explained with the first singular vectors, reconstructing the data with only a few such vectors lead to a very efficient denoising technique. This can be technically achieved by reconstructing the original dataset with the matrix $\mathbf{V}'$, a submatrix of $\mathbf{V}$ of lower rank that only contains the columns associated with the largest singular vectors. A denoised dataset can then be obtained by computing:

$$\mathbf{O}_{\mathrm{denoised}} = (\mathbf{O}\mathbf{V}')\mathbf{V}'^\star. \tag{6}$$

The selection of the number of eigenvectors to keep is often done by computing the residual $\mathbf{O}_{\mathrm{denoised}} - \mathbf{O}$ and characterizing its statistical properties. In the case of Gaussian noise with fixed variance, one can stop adding eigenvectors once the variance of the residual is similar enough to the noise variance. When this is achieved, one can be sure that the selected eigenvectors are retaining the signal and removing the uncorrelated noise. For a more automatic way, see Gavish and Donoho (2014), which provides a strategy for optimally picking the rank of $\mathbf{V}'$.

If the observations are affected by systematic effects, like interferometric fringes (in the case of Stokes observations), they will be part of the output. In many cases in which these systematic effects are strong, one can be lucky and find them isolated in one or two eigenvectors. If this is the case, it is possible to remove these systematics from the observations by deleting these eigenvectors from $\mathbf{V}'$. However, it is often the case that systematic effects are extracted together with real signals in some eigenvectors and they cannot be easily separated. A technique that has recently been proposed is to carry out a rotation in the subspace described by these eigenvectors with the aim of isolating the contribution of systematic effects and real signal (Casini

and Li 2019). Other techniques, with more control from the user side, are based on the techniques presented in Sect. 4.2.

PCA denoising is now systematically used for removing Gaussian noise from the observations (Asensio Ramos et al. 2007a; Martínez González et al. 2016; Jurčák et al. 2018). This denoising is also very helpful in stabilizing spatial deconvolution methods, like the one developed by Ruiz Cobo and Asensio Ramos (2013), which is routinely used to remove the effect of spatial smearing in observations (Quintero Noda et al. 2015, 2016b, a; Felipe et al. 2016; Borrero et al. 2016, 2017).

3.1.2 Interpretability

Somehow surprising, it has been found that, in some specific cases, the leading singular vectors of the PCA decomposition have a well defined physical meaning. This was first pointed out by Skumanich and López Ariste (2002), who demonstrated this for spectropolarimetric observations of a sunspot. They showed that the first singular vector of Stokes I is associated with the average spectrum, the second one gives information about the velocity, and the third one gives information about magnetic splitting or any other broadening mechanism. Likewise, for Stokes V they found that the first singular vector correlates with the longitudinal component of the field, the second one correlates with velocities in the magnetic component and the third one correlates with broadening mechanisms. This is not surprising if one realizes that PCA is akin to a the Taylor expansion of the Stokes profiles (Skumanich and López Ariste 2002). Despite the results discussed so far, PCA often does not extract interpretable physical information from the observations. The reason has to be found on the fact that PCA focuses on global properties of the observations to maximize the amount of variance explained. However, many of the interpretable features of data are local (i.e., the position of the core of the line, the presence of several velocity or magnetic components in the line, etc). When looking for interpretability, other techniques like t-SNE (Student-t Stochastic Neighbor Embedding; Hinton and Roweis 2002) can be more useful (see Sect. 6).

3.1.3 Inversion with lookup tables

The compression capabilities of PCA have been also exploited for accelerating the inversion of Stokes profiles. The process of inverting Stokes profiles consist of inferring the physical properties that produce a given observation. This inversion is usually solved using a maximum likelihood approach in which a merit function (often the χ^2 as a consequence of the assumption of Gaussian noise), that measures the difference between the observations and synthetic Stokes profiles is minimized. This minimization can be done using several techniques. However, the idea when using PCA is to use one of the simplest methods of inversion one can think of: generate a large database with Stokes profiles synthesized in model atmospheres parameterized with N_{par} parameters and pick up the model providing the best fit.

This inversion method, first suggested by Rees et al. (2000), requires some specialized methods for the construction of the database. The reason is that a trivial

method in which every parameter of the model is sampled at n values requires a database of size n_{par}^{N}, which quickly becomes impractical. Moreover, because of the curse of dimensionality, an exponentially large amount of sampled models will lie in the borders of the space and can become useless for the inversion process. As a consequence, this inversion method only works for very simple models.

With these problems in mind, Rees et al. (2000) developed a Monte Carlo approach for populating the database. They started from a model chosen at random. A new model is randomly proposed and the resulting Stokes profiles are compared with the existing ones. If they both lie in a small Euclidean ball of radius ϵ, they are assumed to be coming from very similar atmospheres and only one of them is kept. This procedure is iterated until a sufficiently large database is obtained or when the fraction of accepted new models becomes impractically low. This can be understood as an indication that the space of models is densely sampled. PCA compresses the database by only storing the projections along a few relevant singular vectors. This can lead to compression factors of an order of magnitude, which also accelerates the database search.

After the first pioneering work of Rees et al. (2000), more works followed. They were especially centered on the interpretation of scattering polarization signals and the Hanle effect in lines of He I. The fundamental reason for this is that the solution of the forward problem for these lines is very time consuming, so one better spends the time building a database that can later be used to carry out very fast inversions. This is in contraposition with what happens when a classical iterative algorithm is used for fitting the observations. López Ariste and Casini (2002) proposed using PCA to compress a database of synthetic profiles of the He I D_3 multiplet at 5876 Å using the optically thin approximation. They used the database to invert observations of prominences carried out with the THEMIS telescope, showing that this technique is promising. They obtained magnetic fields that are almost parallel to the solar surface and with strengths around 40 G. The same code was applied to prominence data from the High Altitude Observatory Stokes II polarimeter (Querfeld et al. 1985) by López Ariste and Casini (2003). For computational reasons, the database building process was specifically tailored for the observations, by restricting the ranges of some of the model parameters. Again, the method yields strongly inclined magnetic fields, almost parallel to the solar surface, with strengths as large as 50 G.

The availability of the PCA-based inversion code opened the possibility of quickly inverting 2D maps. For this reason, Casini et al. (2003) observed a prominence with the Dunn Solar Telescope (DST) of the National Solar Observatory (NSO) with a spatial resolution close to 1". A database of 2×10^5 was built for inverting the physical properties of prominences. The resulting maps show magnetic fields with an average of $\sim$20 G, but with blobs displaying strengths above 50 G. Again, the fields are almost parallel to the solar surface. Some possible limitations of the model used for the inversion were discussed in Casini et al. (2005). The same approach of using PCA-compressed databases were also used by López Ariste and Casini (2005) to deal with the inversion of He I D_3 profiles in spicules. This demonstrates that the generation of a look-up table is a suitable inversion procedure for any observation once the database is built with the appropriate ranges of the

model parameters. Their conclusion is that the magnetic field vectors are aligned with the visible structure of the spicule, finding fields above 30 G in some cases.

Later on, databases for the simultaneous inversion of the He I D_3 and 10830 Å multiplets were developed. Lines that are sensitive to scattering polarization and the Hanle effect (as is the case in the mentioned He I lines) suffer from more ambiguities than those whose polarization is controlled only by the Zeeman effect. A careful interpretation of the polarization signal of several lines can potentially help in solving these ambiguities. However, more care needs to be taken when constructing the database precisely to deal with these ambiguities. Casini et al. (2009) built a database with 2.5×10^5 models and used it to invert simultaneous observations of He I D_3 and 10830 Å, resulting in an improved determination of the magnetic field.

Despite its success, the look-up method has some drawbacks, some of them a consequence of the curse of dimensionality:

1. The procedure followed to fill the database has difficulties dealing with ambiguous and quasi-ambiguous solutions. The Zeeman effect is subject to the well-known 180° ambiguity in the azimuth in the reference system of the line-of-sight. In other words, fields whose azimuth on the plane of the sky for 180° produce exactly the same Stokes profiles. When scattering polarization and the Hanle effect dominate, possible additional 90° ambiguities (Hanle ambiguities) appear. Rejecting synthetic profiles that lie inside the ϵ-ball of other preexisting profiles in the database disfavor the representation of physical properties that are subject to ambiguities. This is of almost no importance for profiles controlled by the Zeeman effect but can turn out to be important for those cases dominated by scattering and the Hanle effect.
2. Current observations of Stokes profiles produce noise standard deviations of the noise that reach 10^{-4} in units of the continuum intensity. This means that the ϵ-balls have to be really tiny so that the number of profiles needed to fill a database with such precision quickly becomes unmanageable. Most existing databases have been constructed with larger ϵ-balls, so that we are at the risk of confusing cases in which different (ambiguous) physical configurations but produce similar Stokes profiles. Ideally, one would like to push the limit on the ϵ-balls to very small values, even smaller than the noise level, but this is clearly unfeasible.
3. Filling up the database using the Monte Carlo approach can take a very long time. The first proposed profiles will always be accepted but the fraction of acceptance drops substantially when a few hundred thousand profiles are already part of the database. Additionally, every time one checks for the addition of a new profile, it must be tested against all profiles already present in the database. The number of comparisons to carry out is $N_{\rm tot}(N_{\rm tot}+1)/2$ to fill $N_{\rm tot}$ profiles in the database. Even though each comparison is very fast, the number of them one needs to carry out rapidly makes this approach difficult to use. To partially compensate for this problem, other approaches based on the Latin hypercube sampling have also been used (McKay et al. 1979).
4. When used in evaluation mode, the inversion requires the comparison of the Stokes profiles of interest with all the profiles in the database. This requires the calculation of $N_{\rm tot}$ comparisons for each observed Stokes profile. Recently,

Casini et al. (2013) has devised an indexing method that accelerates the search. It is based on the use of a binary search tree built using the signs of the first n PCA coefficients of each profile. This can potentially lead to an acceleration of a factor 2^{4n} in computing time.

3.2 Fuzzy clustering

The solar corona is the outermost layer of the solar atmosphere and can be observed in various wavelengths (Kasper et al. 2021). In the corona, magnetic pressure dominates over plasma pressure (Gary 2001), and closed magnetic field lines confine plasma, appearing as bright coronal loops in extreme ultraviolet (EUV) wavelengths. Coronal holes are observed as dark areas (Cranmer 2009) and are regions with plasma depletion and lower temperature and density due to the continuous outflow of plasma along "open" magnetic field lines. Proper identification of coronal hole boundaries is crucial as they are a major source of the solar wind (SW), which can affect the Earth's environment (Tsurutani et al. 2006), especially during the declining phase of a solar cycle (Tsurutani et al. 2006). Accurate detection of coronal holes is challenging due to their varying boundaries with wavelength and resolution Ervin et al. (2021).

Developing connectivity models between the Sun and the Earth requires observational constraints from the Sun, and a good evaluation of coronal hole boundaries can help improve these models. Machine learning methods have become increasingly popular for identifying coronal hole boundaries (e.g., Barra et al. 2008, 2009), replacing laborious and experienced observer-based methods. Accurate determination of coronal hole boundaries could also help solve new solar coronal questions such as the "open flux" problem, where the magnetic field in the Earth is two orders of magnitude higher than estimated from the Sun (Linker et al. 2017).

Before machine learning techniques were used, researchers used different techniques for the identification of the CH boundaries. Previously, the identification and mapping of CH were performed based on the helium spectroheliograms and photospheric magnetograms by iterative visual inspection (Henney and Harvey 2005), a laborious process requiring experienced observers. Automatic detection of the CH was realized initially using spectroheliogram images in He I 1083 nm wavelength, or spectral line properties of He I 1083 nm multiplet and other multi-wavelength analysis (Henney and Harvey 2005, and the references therin). In the last decades, we are witnessing a strong increase in the application of ML methods for the identification of the CH boundaries.

One of the earlier identification methods is the spatial possibilistic clustering algorithm[4](SPoCA) which is implemented as part of JHelioviewer[5] (Barra et al. 2008, 2009; Verbeeck et al. 2014). Other approaches are based on segmentation techniques together with ML algorithms (Reiss et al. 2015), or on the fuzzy (Colak and Qahwaji 2013) and k-means clustering (Inceoglu et al. 2022). The SPoCA method, based on an unsupervised fuzzy clustering method (Barra et al. 2008), is a

[4] The latest version can be found in https://github.com/bmampaey/SPoCA.

[5] See Müller et al. (2017) and find it on http://swhv.oma.be/user_manual.

generalization of the k-means clustering discussed in Sect. 3.3. SPoCA implements three types of fuzzy clustering algorithms considered to be appropriate for the EUV solar images: the Fuzzy C-means (FCM); a regularized version of FCM known as Possibilistic C-means (PCM), and a Spatial Possibilistic Clustering Algorithm (SPoCA) that integrates neighbouring intensity values. The SPoCA algorithm was described and implemented by Barra et al. (2008, 2009) for the automatic identification of the CH, AR and quiet sun (QS) in EUV images. The reason for using fuzzy clustering for the EUV images lies in the inherent uncertainty when categorizing visible structures. The SPoCA algorithm works by optimizing the following objective function:

$$J_{\mathrm{SPoCA}}(B,U,X)=\sum_{i=1}^{C}\left(\sum_{j=1}^{N}u_{ij}^{m}\sum_{k\in\mathcal{N}_j}\beta_k d(\mathbf{x}_k,\mathbf{b}_i)+\tau_i\sum_{j=1}^{N}(1-u_{ij})^m\right), \qquad (7)$$

where $C=3$ is the number of clusters ({CH, AR, QS} in this case), N is the number of pixels of the image, $B=\{\mathbf{b}_1,...,\mathbf{b}_C\}$ are the cluster centers, $X=\{\mathbf{x}_j,j=1,\ldots,N\}$ are the feature vectors of dimension p that describe the Sun at each location, U is a fuzzy partition matrix that encodes the membership of feature vector $\mathbf{x}_j$ to class i, $m\geq 1$ is a parameter that controls the degree of fuzzification (a value of $m=1$ means no fuzziness), $\beta_k=1$ if $k=j$, and $\beta_k=(\mathrm{Card}(\mathcal{N}_j)-1)^{-1}$, for any other k, with $\mathrm{Card}(\mathcal{N}_j)$ being the number of elements in the neighborhood of pixel j, d is a distance function in the space of features, and τ_i is the intraclass mean fuzzy distance.

Verbeeck et al. (2014) build upon the SPoCA software to extract, characterize and track CH and AR from EUV images. They used an FCM to initialize a PCM, which is considered more robust to noise and outliers. For the map segmentation, they used different decision rules. Verbeeck et al. (2014) looked mostly at CH and AR and performed a parametric study to determine optimal configurations of the algorithm. The dataset was built based on different EUV imagers. The data used for the study was between 1997 and 2011 and obtained from the EIT/SOHO in 171 and 195 Å. The output of the program is a mask that overlays onto the original image. The solution provides also the location of the AR or CH barycenter. From the results, it was concluded that the FCM yields the best output for extracting CH. The SPoCA detection method is regularly used for feature identification within JHelioviewer and also as a training set for other methods.

One of the challenging tasks in the determination of the CH boundaries is how to discriminate them from filament channels. This is a consequence of the fact that, sometimes, filaments (prominences seen in the solar visible disk) can be mistaken with CH. One of the early attempts in tackling this issue was made by Reiss et al. (2015), who used image segmentation methods together with supervised ML techniques for distinguishing between filaments and CH using AIA/SDO images in the channel at 193 Å. The data is preprocessed by applying intensity-based thresholding and then the CH is identified using SPoCA. After the feature extraction, CH and filaments were manually labeled based on simultaneous Hα images. Additionally, the line-of-sight magnetic field is obtained from HMI/SDO. They

analyzed support vector machines (SVM) for classification (Cortes and Vapnik 1995), decision trees, and random forests as classifiers. They found that the SVM provided the best result, especially when using the magnetic field information.

3.3 k-means

The k-means algorithm (MacQueen 1967) has been widely applied in solar physics, fundamentally in the field of spectropolarimetry, to classify the observed Stokes profiles in 2D maps[6]. k-means tries to cluster the n observed M-dimensional data points $(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$ into k sets $\mathbf{S} = (S_1, S_2, \ldots, S_k)$, defined by their respective cluster centers $(\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_k)$. This is done by obtaining the cluster centers that minimize the intracluster distance for all the points in the dataset:

$$\arg\min_{\boldsymbol{\mu}} \sum_{i=1}^{k} \sum_{j=1}^{n} \mathbf{1}_{S_i}(\mathbf{x}_i) \|\mathbf{x}_i - \boldsymbol{\mu}_j\|^2, \tag{8}$$

where $\mathbf{1}_S(x)$ is the indicator function, which takes the value 1 if the elements belongs to class S and zero otherwise:

$$\mathbf{1}_S(x) = \begin{cases} 1 & \text{if } x \in S \\ 0 & \text{if } x \notin S. \end{cases} \tag{9}$$

Formally, this results into an M-dimensional Voronoi diagram,[7] which has linear decision boundaries. From a practical point of view, this loss function is optimized iteratively as follows:

1. Define a set of cluster centers.
2. Compute the distance between all the observations and the cluster centers.
3. Associate each observation to its closest cluster.
4. Recompute cluster centers and repeat from step 2.

The distance metric used can be tuned for the problem at hand but it is often simply the Euclidean distance (e.g. $\|\mathbf{x}_i - \boldsymbol{\mu}_j\|^2$ in the previous equation). Other metrics like the Mahalanobis distance[8] can be used to account for the covariance in the clusters. All of them produce an M-dimensional Voronoi diagram but the decision boundaries depend on the specific distance metric.

k-means suffers from two fundamental problems. The first one is the inability of the algorithm to infer the number of clusters (i.e., it is a hyperparameter). To automatically extract the number of clusters one needs to resort to more advanced methods. The second problem is that the final positions of the clusters depend on the initialization. The simplest solution to the first problem is to carry out k-means with different values of k and deciding the optimal number by minimizing approximate

[6] The k-means algorithm is available, for instance, on the `scikit-learn` Python package.

[7] A Voronoi diagram is a partition of a hypervolume into regions close to each of a given set of objects.

[8] The Mahalanobis distance measures the distance between two points taking into account the covariance structure of the underlying distribution.

estimations of the Bayesian evidence like the Bayesian information criterion (BIC Schwarz 1978). A better option is to use "density-based spatial clustering of applications with noise" (DBSCAN, see Ester et al. 1996), which can infer the number of cluster centers by finding core samples of high density and expanding clusters from them[9]. Arguably the most robust option is to build a fully hierarchical Bayesian model (Teh and Jordan 2010) in which the number of clusters is considered a random variable.[10] Concerning the second problem, it is often the case that k-means need to be carried out several times to check for proper convergence.

3.3.1 Spectral clustering

k-means was used by Viticchié and Sánchez Almeida (2011) to analyze the circular polarization profiles in the quiet Sun as observed with Hinode/SP (Lites et al. 2013). They ended up inferring that the optimal number of classes is ~ 35 and that they can be grouped in six families according to their general shape. One of the most prominent outcomes is that a large fraction of the observed circular polarization profiles are asymmetric. This means that the inversion of these profiles has to be done with atmospheric models with gradients along the line of sight (see Grossmann-Doerth et al. 1988, for an explanation regarding the physical origin of asymmetric profiles). Later, Kleint et al. (2015) used k-means in the analysis of filament eruption that produced an X-class flare. The strong variability of the observed profiles of the Ca II 8542 Å line (showing emission, absorption, asymmetric, and also flat profiles) made it difficult to estimate the velocity from their Doppler shift. By using k-means to cluster all the profiles into classes, the authors were able to better define a model for each class of profiles to robustly estimate the Doppler shift.

Along this very same line, Panos et al. (2018) used k-means to analyze observations of the Mg II h and k spectral lines in flares with the Interface Region Imaging Spectrograph (IRIS; De Pontieu et al. 2014). Their conclusion, by studying hundreds of thousands of profiles from several tens of flares, is that profiles in flares show a single peak, instead of the double peak typical of the quiet Sun. Additionally, these profiles also show enhanced broadenings and blueshifted central reversals.

Recently, Sainz Dalda et al. (2019) applied k-means for the fast inversion of IRIS profiles. The idea is to cluster the Mg II h and k profiles from a large selection of observations, leading to what they call Representative Profiles (RP). A detailed inversion of this representative profiles with inversion codes like the STockholm Inversion Code (STiC; de la Cruz Rodríguez et al. 2019) can then be done, with the necessary care and the large computing time that these inversions require (they sometimes require of the order of 2 CPU hours per profile). The result is a one-to-one relation between RP and Representative Model Atmospheres (RMA). Afterward, the inversion of maps is carried out by comparing each observed pixel with the list of RP and setting the associated RMA as the solution. When the code is working in

[9] DBSCAN is available on the `scikit-learn` Python package.

[10] In this case, Dirichlet processes are often used as priors. A Dirichlet process is a probability distribution whose range is itself a set of probability distributions. It is used in Bayesian inference to describe the prior knowledge about the distribution of random variables.

evaluation, one finds acceleration factors of 5–6 orders of magnitude in computing time.

The use of RP leads to a huge gain in computing time at the expense of precision in the results. k-means will always select the RP with the smallest distance to the observed profile at every pixel. Although this distance is the smallest, nothing avoids this distance to be large in absolute units, when none of the RP produces a good fit to the observed profile. This would happen, for instance, in pixels with rare spectra, sufficiently rare that it was not statistically present on the training set. Therefore, one should always be cautious and avoid overinterpreting the results. It is always a good practice to visualize the distance (in suitable units) between the observations and the selected RP, paying special attention to those pixels in which the distance is large and its specific reason.

3.3.2 Segmentation of coronal holes

Recently, k-means has been implemented for the identification of CH by Inceoglu et al. (2022). Three of the AIA/SDO wavelengths (171, 193 and 211 Å) were used in different combinations, individual channels, 2-channels (2CC) and 3-channels (3CC) composites, for building the data sets. By computing the within-group sum of square distances as a function of the number of clusters, they manage to give an optimal number of clusters by locating the elbow of the plot (also known as the scree-plot method). The results obtained by applying the k-means method on each of the data sets were compared among themselves to identify the best-performing data set. The results were also compared with CH identified with other methods such as CATCH and HEK (Heliophysics Event Knowledge; Hurlburt et al. 2012). They concluded that k-means has a good overlap with the CHs obtained with CATCH, especially when using the AIA 193 Å channel.

4 Linear models: supervised

Linear regression is arguably the simplest model used in statistics and machine learning and it has become the workhorse of these two disciplines, also in solar physics. Its main assumption is that the signal of interest can be developed as the weighted sum of basis functions:

$$I(x) = \sum_{i=1}^{M} w_i K_j(x), \tag{10}$$

where w_i are the weights associated with the M basis functions $K_j(x)$. The flexibility in the selection of the basis functions is one of the reasons for the power and flexibility of linear models. Additionally, the linear character of the model simplifies the calculations, in many cases allowing to carry out analytical calculations.

4.1 Hermite functions

While PCA provides a purely empirical orthogonal basis set to represent the Stokes profiles, other more classical approaches have been tried in the literature. One that is particularly relevant is the use of Hermite functions, developed by del Toro Iniesta and López Ariste (2003). They realized that these functions when defined as

$$h_n(\lambda) = \left(2^n n! \sqrt{\pi}\right)^{-1/2} \exp\left[-\lambda^2/2\right] H_n(\lambda), \tag{11}$$

where $H_n(\lambda)$ are the Hermite polynomials as a function of the wavelength λ, look very similar to the Stokes profiles when displayed in wavelength units normalized to the width of the spectral line. $h_0(\lambda)$ is a Gaussian function, very similar to Stokes I, $h_1(\lambda)$ looks similar to Stokes V when dominated by the Zeeman effect, while $h_2(\lambda)$ looks similar to Stokes Q and U in the same regime. Although interesting from a mathematical point of view, the Hermite expansion has not been used in real situations because they work well only when all the profiles have a definite width. When Stokes profiles of different widths are present in the field of view, empirical decompositions like PCA are definitely much more efficient.

4.2 Relevance vector machines

Very powerful methods have been proposed and used in solar physics for regression based on non-parametric models. Non-parametric regression relies on the application of a sufficiently general function that only depends on observed quantities and that is used to approximate the observations. A very flexible and efficient non-parametric regression method is that of the relevance Vector Machines (RVM; Tipping 2000), a Bayesian update of the support vector machine learning technique of Vapnik (1995).[11] In this case, the general function is just the linear combination of user-defined kernels of Eq. (10) with $x = \lambda$. The $K_j(\lambda)$ functions are arbitrary and defined in advance, and w_i is the weight associated to the i-th kernel function. The parameters we infer from the data appear linearly in the model once the kernel functions are fixed. For instance, if the kernel functions are chosen to be polynomials, one ends up with a standard polynomial regression.

The main advantage of non-parametric regression is that the model automatically adapts to the observations. For this adaptation to occur, the basis functions should ideally capture all possible ways in which the signal can behave. The number of basis functions one can include in the linear regression can be arbitrarily large, making Eq. (10) a very powerful model for any unknown signal. As an example, one can use a combination of polynomials of many different orders, sinusoidal of many frequencies, and Gaussians at different positions and with different widths to approximate a very general spectral line.

Obviously, this makes the regression problem ill-defined and the solution severely overfits the data provided that M is large enough. For this reason, Tipping (2000) proposed to circumvent overfitting by pursuing a hierarchical Bayesian approach. In

[11] https://github.com/aasensio/rvm.

this case, a Gaussian prior is imposed for each one of the w_i. This prior is made dependent on a set of hyperparameters α_i, which are learned from the data. If a Jeffreys' prior is imposed on α_i, i.e., $p(\alpha_i) \propto \alpha_i^{-1}$, the resulting prior for w_i is $p(w_i) \propto |w_i|^{-1}$. In essence, with the specific priors, in the limiting case that α_i tends to infinity, the marginal prior for w_i is so peaked at zero that it is compatible with a Dirac delta. This means that this specific w_i does not contribute to the model and can be dropped without impact. This regularization proposed by Tipping (2000) leads to a sparse $\mathbf{w}$ vector, so an automatic relevance determination is implemented in the method.

This method was applied for the first time for denoising purposes by Asensio Ramos and Manso Sainz (2012) in solar physics. For this purpose, one selects Gaussian functions of different widths centered at each one of the spectral points observed. This obviously constitutes an overdetermined non-orthogonal dictionary[12] but the regularizing properties of RVM help in keeping only a few active Gaussians, which explain the observations. The remaining signal is considered to be noise. López Ariste (2014) proposed it as a very efficient method for fringe removal from data. Fringes appear in observed spectra because of the internal reflection in thin plates in the optical path. As a consequence, the observed spectrum can be understood as a combination of quasi-periodic fringes plus the original spectrum. López Ariste (2014) proposed Gaussian functions, $G_j(\lambda)$ of different widths for explaining the spectral lines and a combination of sines and cosines, $P_j(\lambda)$ for explaining the fringes:

$$I(\lambda) = \sum_{i=1}^{M} p_i P_j(\lambda) + \sum_{i=1}^{M} w_i G_j(\lambda). \tag{12}$$

The sparsity regularization that is part of RVM produces that spectral lines are not efficiently developed with periodic functions. One would need lots of sines and cosines to do that and this is penalized by the model. Likewise, fringes are not efficiently developed with Gaussians for precisely the same reason. Once the regression is done, defringing is done by removing the quasi-periodic component and computing:

$$I_{\text{defringed}}(\lambda) \approx \sum_{i=1}^{M} w_i G_j(\lambda). \tag{13}$$

4.3 Compressed sensing and sparsity regularization

The theory of compressed sensing has emerged recently to solve strongly undetermined problems. One case of that is the recovery of signals from measurements. It is a well-known fact that band-limited signals need to be sampled according to the Nyquist-Shannon theorem. If not, the latent function cannot be

[12] A dictionary is a set of potentially non-orthogonal functions that are used to represent a signal as a linear expansion.

properly recovered from the samples. During the last few years, the emerging theory of compressed sensing (CS; Candès et al. 2006b; Donoho 2006) has shown that this sampling is indeed too restrictive when some details of the signal structure are known in advance. Although this might sound counterintuitive, it is indeed true that, in many instances, natural signals have a structure that is known in advance, in many cases motivated by physical arguments. For instance, stellar oscillations can be represented by sinusoidal functions of different frequencies, images can be represented in a multiresolution analysis using wavelets, etc. The key point is that, typically, only a few elements of the basis set in which we develop the signal are necessary for an accurate description of the important physical information. The innovative character of CS is that this compressibility of the observed signals is inherently taken into account in the measurement step, and not only in the post-analysis, thus leading to efficient measurement protocols. Instead of measuring the full signal (wavelength variation of the Stokes profiles in our case), under the CS framework one measures a few linear projections of the signal along some vectors are known in advance and reconstruct the signal solving a nonlinear problem. For a more in-depth description, we refer the reader to recent references (e.g., Baraniuk 2007; Candès and Wakin 2008, and references therein).

The usage of compressive sensing techniques for the measurement of a signal, represented as a vector $\mathbf{x}'$ of length M, is based on the following two key ideas:

1. Instead of measuring the signal itself, one measures the scalar product of the signal with carefully[13] selected vectors:

$$\mathbf{y} = \boldsymbol{\Phi}\mathbf{x}' + \mathbf{e}, \tag{14}$$

where $\mathbf{y}$ is the vector of measurements of dimension N, $\boldsymbol{\Phi}$ is an $N \times M$ sensing matrix and $\mathbf{e}$ is a vector of dimension N that characterizes the noise on the measurement process. Note that the previous equation describes the most general linear multiplexing scheme in which the number of measurements M and the length of the signal N may differ. In the standard multiplexing case, the number of measured scalar products equals the dimension of the signal ($N = M$). Consequently, it is possible to recover the vector $\mathbf{x}'$ provided that $\mathrm{rank}(\boldsymbol{\Phi}) = N$, so that the problem is not ill-conditioned. In other words, one has to verify that every row of the $\boldsymbol{\Phi}$ matrix is orthogonal with respect to every other row.
2. The assumption that the signal of interest is sparse in a certain basis set (or can be efficiently compressed in this basis set). Any compressible signal can be written, in general, as:

$$\mathbf{x}' = \mathbf{W}^T\mathbf{x}, \tag{15}$$

where $\mathbf{x}$ is a K-sparse vector (if only K elements of the vector are different from zero) of size M and $\mathbf{W}^T$ is the transpose of an $M \times M$ transformation matrix associated with the basis set in which the signal is sparse. For instance, $\mathbf{W}$ can be the Fourier matrix if the signal $\mathbf{x}$ is the combination of a few sinusoidal components. Other transformations of interest are the wavelet matrices or even

[13] The precise meaning of "carefully" can be found in Candès et al. (2006b).

empirical transformation matrices like those found using principal component analysis.

The combination of those ingredients leads to the multiplexing scheme:

$$\mathbf{y} = \boldsymbol{\Phi}\mathbf{W}^T\mathbf{x} + \mathbf{e}, \tag{16}$$

with the hypothesis that $\mathbf{x}$ is sparse, which renders CS feasible. It has been demonstrated by Candès et al. (2006b) that, even if $\mathrm{rank}(\boldsymbol{\Phi}\mathbf{W}^T) < N$ (we have fewer equations than unknowns), the signal $\mathbf{x}$ can be recovered with overwhelming probability when using appropriately chosen sensing matrices $\boldsymbol{\Phi}$. When the number of equations is less than the number of unknowns, it is usual to solve Eq. (16) using least-squares methods that try to minimize the ℓ_2 norm[14] of the residual. This is usually accomplished using techniques based on the singular value decomposition (see, e.g., Press et al. 1986). However, such minimization is known to return non-sparse results (e.g., Romberg 2008). A more appropriate solution is to look for the vector with the smallest ℓ_0 pseudo-norm (the number of non-zero elements of the vector) that fulfills the equation:

$$\arg\min_x \|\mathbf{x}\|_0 \quad \text{subject to} \quad \|\mathbf{y} - \boldsymbol{\Phi}\mathbf{W}^T\mathbf{x}\|_2 < \epsilon, \tag{17}$$

where ϵ is an appropriately small quantity. The solution to the previous problem is, in general, not computationally feasible. However, Candès et al. (2006b, 2006a) demonstrated that, under certain conditions for the matrix $\boldsymbol{\Phi}\mathbf{W}^T$ (Candès et al. 2006b), the problem reduces to:

$$\arg\min_x \|\mathbf{x}\|_1 \quad \text{subject to} \quad \|\mathbf{y} - \boldsymbol{\Phi}\mathbf{W}^T\mathbf{x}\|_2 < \epsilon, \tag{18}$$

The advantage lies in the fact that very efficient numerical methods exist for the solution to such a problem[15].

The theory of CS has extensively been used in solar physics after Asensio Ramos and López Ariste (2010) introduced it into the field of spectropolarimetry. Given that the theory relies on the compressibility of signals, these authors tested whether this is indeed practically the case for the Stokes profiles. They showed that polarimetric signals in many spectral lines can be efficiently compressed using PCA and also non-empirical basis sets like different families of wavelets. Once this is verified, they proposed several potential applications of the CS theory to the measurement of Stokes profiles. The first one is the conceptual idea of a multiplexing spectro-imager. This is an extension of the classical double pass subtractive spectrographs which work as follows: i) the slit of the standard spectrograph is removed; ii) a coded narrow slit following a Hadamard orthogonal sequence is located in the focal plane of the spectrograph, together with a device to return the light through the spectrograph in subtractive mode. As a consequence, an image is formed at the entrance of the spectrograph where each column corresponds to a different

[14] The ℓ_q norm of a vector is given by $\|\mathbf{x}\|_q = \left(\sum x_i^q\right)^{1/q}$ when $q \geq 1$.

[15] Some CS problems can be solved using the `scikit-learn` Python package.

wavelength. This idea later became real with the development of the Tunable Universal Narrowband Imaging Spectrographs (TUNIS; López Ariste et al. 2010, 2011). The inverse problem to recover the original monochromatic images is solved using CS and a sparsity constraint in the spectral direction.

Asensio Ramos and López Ariste (2010) also proposed a sub-Nyquist spectrograph, in which the pixel size is several times larger than the spectral sampling of the spectrograph. The original spectral resolution of the spectrograph is obtained by solving again a CS problem. The authors demonstrated that, under certain conditions, the original resolution can be recovered. This might be of relevance for very high resolution spectrographs, which also require cameras with a large number of pixels to cover a sufficiently large spectral range.

Another idea suggested by Asensio Ramos (2010) was the application of the CS theory to Fabry-Perot etalons (FPE). Almost all successful FPE consists of three optical elements: a relatively narrow filter and two etalons of different free spectral ranges. An etalon is a thin plate that works as a periodic frequency filter with well-defined transmission peaks of high transparency. When the two etalons are appropriately aligned and tuned, the transmission profile of the combination has a very high transmission peak. The secondary transmission peaks are strongly reduced, although this spectral structure is again periodic with a much larger period. The narrow filter serves to isolate only one of the transmission peaks. Tuning the etalons is a very difficult task and, for this reason, only a few such instruments exist. Asensio Ramos (2010) suggested that one can use the CS theory to remove one of the etalons and still recover the original signal. The numerical experiments using PCA as a sparsity-inducing basis set was successful. However, no instrument is still based on this idea. Probably one of the reasons is that one needs to precompute the basis set, and for this, a normal spectrograph is needed. Recently, Molnar et al. (2020) demonstrated that the application of neural networks for the solution of CS problems can overcome this difficulty and recover a large fraction of the spectral resolution lost during the observation with the instrument.

The idea of a sub-Nyquist polarimeter was put forward by Asensio Ramos (2016) using the CS theory. Such a polarimeter modulates the polarimetric properties of the incoming light at very high frequencies (roughly at kHz rates) to freeze the variations of the refraction index of the Earth atmosphere, but measures at a much slower rate (of only a few hundred Hz). The camera then integrates the modifications to the Stokes parameters produced by seeing variations. Consequently, one ends up solving a linear recovery problem like that of Eq. (18), under the assumption that the seeing variations are compressible in the Fourier basis. This is indeed approximately the case given that the power spectrum of the seeing roughly follows a $1/f^2$ law. The simulations carried out by Asensio Ramos (2016) demonstrated that it is possible to recover the seeing variations at kHz frequencies from integrations one order of magnitude slower, with a very robust behavior with noise.

Another very fruitful field of application of compressed sensing is in the thermal diagnostics of the corona. The multiband observations capabilities of the Atmospheric Imaging Assembly instrument (AIA; Lemen et al. 2012) onboard the Solar Dynamics Observatory (SDO; Pesnell et al. 2012) can be potentially used to

constrain the temperature and densities of the optically thin plasma in the solar corona. This is done via the solution of the linear Differential Emission Measure (DEM) problem, which can be posed as a linear system once the problem is discretized. The DEM problem is severely undetermined and its solution must be regularized to find a reliable result. Cheung et al. (2015) applied a sparsity constraint by posing the DEM inversion problem in the form of Eq. (18), with the additional constraint that each component of the solution vector be non-negative (since the EM is proportional to the square of the free electron density, it must be non-negative to be physically meaningful). The method works by proposing an overcomplete and non-orthogonal dictionary composed of Dirac-delta and Gaussian functions that cover the expected range of temperatures in a logarithmic scale. The solution method imposes sparsity on the coefficients associated with the elements of the dictionary to find the final combination that explains the observations. The method was validated in a large variety of synthetic cases, from simple ones to thermodynamic models obtained from a fully compressible, 3D magneto-hydrodynamic (MHD) simulation of an active region. Later, Su et al. (2018) pointed out that the selection of widths of the Gaussians that are part of the dictionary proposed by default by Cheung et al. (2015) could lead to some problems in flaring regions. They decreased the default width of some of the Gaussians and also increased the $\log T$ gridding to allow for more thermal structure. The thermal structure inferred from AIA data alone is, consequently, more consistent with thermal X-ray observations.

Compressed sensing has also been proposed by Cheung et al. (2019) for the analysis of current and future multi-slit spectroscopic instruments, like the Multi-slit Solar Explorer (MUSE; De Pontieu et al. 2020). These multi-slit instruments observe different regions of the solar surface. The dispersive element used for the analysis of the spectrum produces, at the detector, a superposition of spectra originating from all slits. Disentangling this mixture is again done by solving a linear problem like that of Eq. (14), where the mixture matrix depends on the specifics of the instrument. Cheung et al. (2019) proposed that an ℓ_1 constraint can be used to successfully solve the problem. This method has also been adapted for unfolding overlapping EUV spectra in slitless imaging spectrometer data, e.g., for the COronal Spectroscopic Imager in the EUV (COSIE Winebarger et al. 2019; Golub et al. 2020).

Sparsity constraints can also be applied to the solution of nonlinear problems, like the inversion of Stokes profiles. In this case, the problem to be solved is:

$$\arg\min_{x} \|\mathbf{x}\|_1 \quad \text{subject to} \quad \|\mathbf{y} - f(\mathbf{W}^T\mathbf{x})\|_2 < \epsilon, \tag{19}$$

where $\mathbf{y}_{\rm syn} = f(\mathbf{p})$ are the synthetic Stokes profiles. These are obtained by solving the radiative transfer equation on a model atmosphere parameterized by the vector of physical properties $\mathbf{p}$. Using this approach, Asensio Ramos and de la Cruz Rodríguez (2015) developed a new 2D inversion code under the Milne-Eddington approximation (see Landi Degl'Innocenti and Landolfi 2004). The solution is regularized by assuming that the maps of physical properties are sparse in a wavelet basis. The sparsity constraint effectively reduces the number of free parameters of the problem and produces much cleaner inverted maps. This approach has also been exploited by Asensio Ramos et al. (2016) to invert Stokes profiles that can be affected by

systematic effects that are not part of the line formation model (e.g., fringes, blends, etc.).

Hybrids of unsupervised and supervised models including sparsity have also been built for solar flare prediction (Benvenuto et al. 2018). In this case, a sparsity constrained linear model is used to extract relevant features from the observations, while a variant of k-means is used to cluster the resulting features. The results show that the synergy between the supervised and unsupervised methods performs classification better than previous approaches.

5 Deep neural networks

Arguably the most successful machine learning methods nowadays are based on deep nonlinear artificial neural networks (ANN), especially deep neural networks (DNN). For this reason, we focus this section on the description of ANNs which we consider to be models with great potential in the field.

ANNs are well-known computing systems based on connectionism that can be considered to be universal approximants (Bishop 1996) to arbitrary functions (a theorem demonstrated by Cybenko 1988). They are inspired by the connectivity of animal brains and their origin can be traced back to the 1940 s. At that time, some ideas of how to carry out computations based on mimicking animal brains appeared (McCulloch and Pitts 1943). After some theoretical advances, Rosenblatt (1958) built the Mark I Perceptron machine, the first implementation of a perceptron, a supervised algorithm for binary classification. ANNs slowly evolved over the decades but never emerged as the method of choice for machine learning. The fundamental reason for this was, as demonstrated in recent years with the success of deep learning, fundamentally wrong. Their training is based on the optimization of a scalar loss

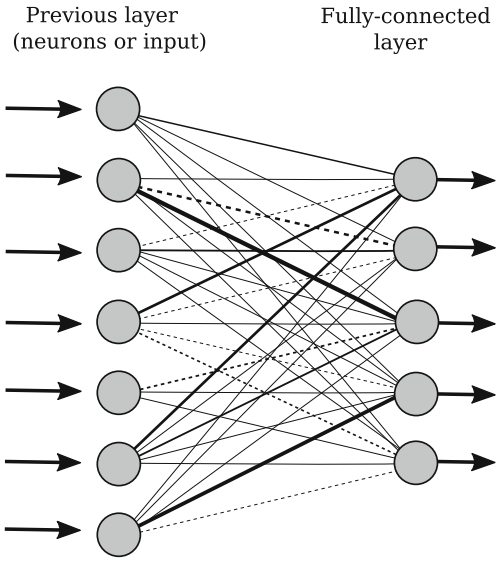

Fig. 4 Building block of a fully-connected neural network. Each input of the previous layer is connected to each neuron of the output. Each connection is represented by different lines where the width is proportional to the absolute value of the weight. Solid lines represent positive weights while dashed lines refer to negative weights

function that is non-convex in the parameters. Consequently, locating the global minima is a daunting task. In fact, it may not exist at all and the loss landscape is made of a plethora of local minima. For this reason, the machine learning community preferred to use methods based on convex loss functions, like the ones presented in the previous section. Only in recent years, researchers are starting to understand the loss landscape and realize that non-convexity is in fact the property that has opened up the current revolution in machine learning.

The building block of an artificial neural network is shown in Fig. 4. The most basic constituent of a neural network is the neuron (inspired by biological neurons but not strictly equivalent), shown as grey circles in the figure. From a mathematical point of view, a neuron can be understood as a simple storage of a real number, which is then used in some predefined operations when this neuron is connected with other neurons. These connections can be massive and this connectivity is precisely the one that gives enormous representation power to neural networks. The state of each neuron i is computed by a very basic operation on the input vector: it multiplies all the input values x_j by some weights w_j, adds some bias b_i and finally returns the value of a certain user-defined nonlinear activation function $f(x)$. In mathematical notation, a neuron computes:

$$y_i = f\left(\sum_j x_j \cdot w_j + b_i\right), \tag{20}$$

which is a generalization of the simple model for a neuron of McCulloch and Pitts (1943). The output y_i is then input in another neuron that does a similar operation. Therefore, neural networks can be considered to be a complex composition of very simple nonlinear functions. Each layer k is parameterized by a set of parameters $\boldsymbol{\theta}^{(k)}$. After passing through the L layers the output can be written as:

$$\mathbf{y} = f(\mathbf{x};\boldsymbol{\theta}) = f^{(L)}_{\boldsymbol{\theta}^{(L)}}(\cdots f^{(2)}_{\boldsymbol{\theta}^{(2)}}(f^{(1)}_{\boldsymbol{\theta}^{(1)}}(\mathbf{x}))). \tag{21}$$

It is sometimes useful to make explicit all intermediate features of the neural network:

$$\begin{aligned}
\mathbf{y}^{(1)} &= f^{(1)}_{\boldsymbol{\theta}^{(1)}}(\mathbf{x}) \\
\mathbf{y}^{(2)} &= f^{(2)}_{\boldsymbol{\theta}^{(2)}}(\mathbf{y}^{(1)}) \\
&\cdots \\
\mathbf{y}^{(L-1)} &= f^{(L-1)}_{\boldsymbol{\theta}^{(L-1)}}(\mathbf{y}^{(L-2)}) \\
\mathbf{y} &= f^{(L)}_{\boldsymbol{\theta}^{(L)}}(\mathbf{y}^{(L-1)})
\end{aligned} \tag{22}$$

Using the standard notation for function composition ($\circ$), a neural network then provides the following output:

$$\mathbf{y} = (f^{(L)}_{\boldsymbol{\theta}^{(L)}} \circ f^{(L-1)}_{\boldsymbol{\theta}^{(L-1)}} \circ \cdots \circ f^{(2)}_{\boldsymbol{\theta}^{(2)}} \circ f^{(1)}_{\boldsymbol{\theta}^{(1)}})(\mathbf{x}). \tag{23}$$

Precisely this composition character is the one that allows graphical models like the one depicted in Fig. 4 to be useful. The compositional character opens up the possibility to split very complex models as the combination of smaller submodels. This abstract *block* representation gives neural networks an enormous potential because they can be engineered quite easily to the solution of a very broad class of problems.

In many cases, an ANN can be understood as a pipeline where the information goes from the input to the output, where each neuron makes a transformation like the one described above. Each transformation deforms the topology of the input space (Naitzat et al. 2020) with the hope that the final prediction turns out to happen in a much simpler space. Neurons are usually grouped in layers and the number of connected layers defines the depth of the network. For reasons that will become clear in Sect. 5.3.3, very deep neural networks are hard to train, and only in the last decade, we have been able to do that. Currently, some of the most successful neural networks contain millions or billions of neurons organized in several tens or hundreds of layers (Simonyan and Zisserman 2014).

One may ask: How do we know which weights and biases to use to get an optimal result for our supervised learning problem? The optimal values for the weights and biases are unknown before training. They are parameters of the ANN, typically initialized by sampling from a random distribution (e.g., normal distribution). The task of supervised training is to provide samples of input and targets (rows of $\mathbf{X}$ and $\mathbf{Y}$) so that the loss function can be evaluated, and gradient descent be used to update the parameters of the ANN. See Sect. 5.3 for a discussion of how ANNs are efficiently trained.

5.1 Architectures

5.1.1 Multi-layer fully connected neural networks

The most used type of neural network from the 1980 s to the 2000 s is the fully connected network (FCN; see Schmidhuber 2014, for an overview), in which every input of all considered layers is connected to every neuron of the following layer. Likewise, the output transformation becomes the input of the following layer (see left panel of Fig. 4). This kind of architecture succeeded to solve problems that were considered to be not easily solvable, such as the recognition of handwritten characters (Bishop 1996).

5.1.2 Convolutional neural networks

Despite the relative success of neural networks, their application to high-dimensional objects like images or videos turned out to be an obstacle. The fundamental reason was that the number of weights in a fully connected network increases extremely fast with the complexity of the network (defined by the number of neurons) and the computation quickly becomes unfeasible. As each neuron of a given layer is

connected to every neuron of the previous one, adding a new neuron to a layer also implies adding a large number of weights, equal to the number of neurons in the layer. The number of weights of a deep fully connected neural network is, then:

$$N = \sum_{\mathrm{i} \in \mathrm{layers}} N_i N_{i-1}, \tag{24}$$

where N_i is the number of neurons of layer i. A larger number of neurons implies then a huge increase in the number of connections. This became an apparently insurmountable handicap, which was only solved with the appearance of convolution neural networks (CNN or ConvNets; LeCun and Bengio 1998). The idea brought forward by LeCun and Bengio (1998) was motivated by biological processes and exploit the fact of sharing weights across the input. From a mathematical point of view, CNNs define a set of kernels of small size that are then used as convolution kernels. The input is then convolved with them, providing as output that is known as *feature map*. The fundamental advantage of CNNs is that sharing the weights across the whole input drastically reduces the number of unknowns. As a side effect, convolutions also make CNN's shift invariant (features can be detected in an image irrespectively of where they are located), a very powerful inductive bias.

For a two-dimensional input X of size $N \times N$ with C channels[16] (a cube or tensor of size $C \times N \times N$), each output feature map O_i (with size $1 \times N \times N$) of a convolutional layer is computed as:

$$O_i = K_i * X + b_i, \qquad i = 1, \ldots, M, \tag{25}$$

where K_i is the $C \times K \times K$ kernel tensor associated with the output feature map i, b_i is a bias value ($1 \times 1 \times 1$) and the symbol $*$ is used to refer to the the convolution operation[17]. Once the convolution with M different kernels is carried out and stacked together, the output O will have size $N \times N \times M$. All convolutions are here indeed intrinsically three-dimensional, but one could see them as the total of $M \times C$ two-dimensional convolutions plus the bias.

Like the weights of a fully-connected network, the optimal weights of a convolutional kernel are unknown. Instead they are initialized (often with values sampled from random distributions) and updated during training time. Regardless of the value of the kernels, the individual application of the convolutional operator (as given in Eq. (25)) and the serial composition of such operations remain linear operations. To introduce nonlinearities and increase the expressivity, convolutional layers are often succeeded by activation functions (see Sect. 5.2). Pooling layers are also used to improve the spatial connectivity of CNNs and to reduce the dimensionality of the input. For example, the *maxpool* operation returns the maximum value in non-overlapping windows of size $N_{\mathrm{sub}} \times N_{\mathrm{sub}}$ pixels. Often,

[16] The term channels is inherited from the those of a color image (e.g., RGB channels). However, the term has a much more general scope and can be used for arbitrary quantities (see Asensio Ramos et al. 2017, for an application).

[17] In most ML framework implementations of convolutional layers, the $*$ operator is actually the cross-correlation instead of convolution, as is usually defined in the mathematical literature. The difference between the two operations are irrelevant because kernels will be learned during training, but the correlation is more computationally efficient.

applications of the convolutional layer and/or maxpool layers are strided. For a stride of one, a convolution displaces the kernel on the input with a step of one pixel. When the stride is larger than one, the convolution kernel is displaced in larger steps. This practice reduces the nominal dimensionality (i.e., the number of components of the output, not the intrinsic rank) of the output. Repeated application of this type of downscaling reduces the number of trainable parameters, and thus the computational effort needed to train the network.

Like fully connected layers, CNNs are typically composed of several layers. This layer-wise architecture exploits the property that many natural signals are generated by a hierarchical composition of patterns. For instance, faces are composed of eyes, while eyes contain a similar internal structure. This way, one can devise specific kernels that extract this information from the input. CNNs work on the idea that each convolution layer extracts information about certain patterns, which is done during the training by iteratively adapting the set of convolutional kernels to the specific features to locate. This obviously leads to a much more optimal solution as compared with hand-crafted kernels. Despite the exponentially smaller number of free parameters as compared with a fully-connected ANN, CNNs often produce much better results.

It is interesting to note that, since a convolutional layer just computes sums and multiplications of the inputs, the same operation could be done with a multi-layer FCN. However, training such a neural network would require huge amounts of training data to learn the natural inductive biases of locality and shift invariance of CNNs (Peyrard et al. 2015).

Although a convolutional layer significantly decreases the number of free parameters as compared with a fully-connected layer, it introduces some hyperparameters (global characteristics of the network) to be set in advance: the number of kernels to be used (number of feature maps to extract from the input), the size of each kernel with its corresponding padding (to deal with the borders of the image) and the stride (step to be used during the convolution operation) and the number of convolutional layers and specific architecture to use in the network. As a general rule, the deeper the CNN, the better the result, at the expense of a more difficult and computationally intensive training.

5.1.3 Recurrent neural networks

The efficient description of sequences of data requires neural networks with a different architecture. In this case, it turns out to be important to have feedback connections to keep track of long-term dependencies in the input sequences. Recurrent neural networks (RNNs; Rumelhart et al. 1986) can keep track of these dependencies by unrolling the network for all the elements of the sequence and connecting the output of each neuron in the sequence to the input of the next one. RNNs are designed to learn sequential or time varying patterns (Medsker and Jain 2021), like for example the solar cycle variation. RNNs started to be used initially for solving character recognition problems, but they were also implemented in many other fields, like financial predictions, the verification of the water quality, etc. The architecture can be built on fully or partially-connected layers, including multilayer

feedforward networks and specific learning algorithms were developed for the RNNs (Medsker and Jain 2021). The RNNs were initially difficult to train because they suffer from the vanishing gradient problem (see Sect. 5.3.3). Different architectures were proposed to cure this problem, and the long short-term memory (LSTM; Hochreiter and Schmidhuber 1997b) is arguably the most successful. In solar physics, the LSTM was intensively applied for the prediction of the current solar cycle (see Sect. 7.6).

5.1.4 Attention and transformers

The attention mechanism, which is a variety of algorithms that compute the output by weighting the importance of different features of the data, has become important thanks to the Transformer model (Vaswani et al. 2017). Transformers can translate a sequence of arbitrary length into a sequence of the same length of features of arbitrary dimensionality using self-attention. Given an input $\mathbf{X}$, self-attention works by building matrices of values ($\mathbf{V}$), queries ($\mathbf{Q}$) and keys ($\mathbf{K}$) by using trainable weight matrices:

$$\mathbf{V} = \mathbf{W}_V \mathbf{X}, \quad \mathbf{Q} = \mathbf{W}_Q \mathbf{X}, \quad \mathbf{K} = \mathbf{W}_K \mathbf{X} \tag{26}$$

and computing:

$$\mathrm{Att}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \mathrm{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V}, \tag{27}$$

where d_k is the dimensionality of the queries and keys. The product of the query and key matrices is a score matrix that defines the amount of attention that each element of the output pays to every element of the input sequence. This score matrix is then scaled down to allow for more stable gradients, and a softmax is applied to transform the scores into probabilities. Finally, these attention weights are applied to the values. Transformers have not been yet applied in solar physics (see Sect. 7.10.6 for an usage of attention). However, given their success in other fields, we anticipate that this kind of attention model will eventually emerge as suitable ones in the analysis of images or sequences.

5.1.5 Graph neural networks

Neural networks can also be defined in graphs, which is sometimes appropriate for specific problems. These problems are still hard to find in solar physics but at least one application already exists (see Sect. 7.7). A connected graph $G = (V, E)$ is defined by the set of grid points V (also known as nodes or vertices) and the set of edges connecting the grid points, E. Each node can encode relevant properties $\mathbf{p}_i$. Each edge $\mathbf{e}_{ij}$ connects the two nodes i (sender) and j (receiver), and describes relevant inter-node properties. The computation inside the graph is based on the so-called *processor*, made of N consecutive message passing processes. Message passing is fundamental to connect the information in very distant nodes in the graph, given that all updates are local, as shown in the following. Each message passing

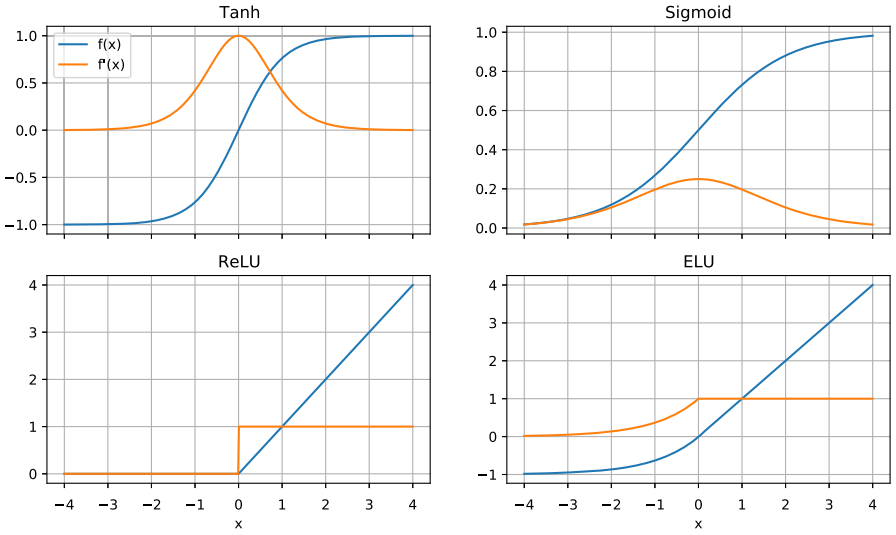

Fig. 5 Some activation functions often used in ANNs: hyperbolic tangent (Tanh), sigmoid, rectified linear unit (ReLU), and exponential linear unit (ELU)

consists of updating the latent information contained in all edges and then in all nodes, as follows:

$$\begin{aligned}\mathbf{e}_{ij}^{t+1} &= \mathbf{e}_{ij}^{t} + f_E^{t+1}(\mathbf{e}_{ij}^{t}, \mathbf{v}_i^{t}, \mathbf{v}_j^{t}),\\ \bar{\mathbf{e}}_j^{t+1} &= \sum_k f_A^{t+1}\left(\mathbf{v}_k^{t}, \mathbf{e}_{kj}^{t+1}\right),\\ \mathbf{v}_j^{t+1} &= f_V^{t+1}\left(\mathbf{v}_j, \bar{\mathbf{e}}_j^{t+1}\right),\end{aligned} \tag{28}$$

where f_E, f_A and f_V are neural networks. After a predefined number of message passing steps, one ends up with updated information in the nodes and in the edges. In a supervised training setup, this updated information is then compared with that of the training set and the weights of the neural networks are updated until convergence.

5.2 Activation layers

The output of a linear layer of a neural network is often passed through a nonlinear function, known as the *activation function*. This function introduces the non-linear character into the neural networks, which is the source of its strength. Although hyperbolic tangent, $f(x) = \tanh(x)$, or sigmoidal, $f(x) = [1 + \exp(-x)]^{-1}$, activation units were originally used in ANNs (see Fig. 5), nowadays a panoply of more convenient nonlinearities are used. Probably the most common activation function is the Rectified Linear Unit (ReLU; Nair and Hinton 2010) or slight variations of it, like

the exponential linear unit (ELU; Clevert et al. 2015). The ReLU replaces all negative values in the input by zero and keeps the rest untouched:

$$\mathrm{ReLU}(x) = \max(0, x). \tag{29}$$

This activation has the desirable property of having a constant derivative for positive arguments, which greatly accelerates the training and reduces the vanishing gradient problem. Examples of a few activations functions are displayed in Fig. 5.

5.3 Training

Neural networks (either deep or shallow) can be seen as a very flexible parametric function that produces an output, $\mathbf{y}$, from an input, $\mathbf{x}$, with the aid of some internal parameters, $\boldsymbol{\theta}$. These parameters are the weights and biases of all layers, together with any possible learnable parameter of the activation layers. Training is performed by iteratively modifying the vector of parameters $\boldsymbol{\theta}$ until a loss function is minimized. This can be seen as a standard maximum-likelihood optimization when the loss function is given by the likelihood function.

5.3.1 Loss function

In general, a loss function is a differentiable scalar function that depends on the inputs and outputs, as well as any parameter or internal feature of the neural network. Using the definition of the neural network functional form of Eq. (23), the most general loss function is represented by the following scalar:

$$L = g(\mathbf{x}, \mathbf{y}, \{\boldsymbol{\theta}^{(L)}, \ldots, \boldsymbol{\theta}^{(1)}\}, \{\mathbf{y}^{(L-1)}, \ldots, \mathbf{y}^{(1)}\}), \tag{30}$$

where the dependence on $\boldsymbol{\theta}^{(i)}$ shows the contribution of the weights of all intermediate layers in the neural network, while the dependence on $\mathbf{y}^{(i)}$ shows the dependence on all intermediate features.

5.3.2 Gradient descent

In general, and irrespective of the specific loss function, the optimization is routinely solved using simple first-order gradient descent algorithms (GD; see Rumelhart et al. 1988), which modifies the weights using the gradient of the loss function with respect to the model parameters.

In practice, procedures based on the so-called stochastic gradient descent (SGD) are used, in which only a few examples from the training set (a batch) are used during each iteration to compute a noisy estimation of the gradient and adjust the weights accordingly. A training set is then divided into n batches, each one containing B training examples. Although the calculated gradient in a batch is a noisy estimation of the one calculated with the whole training set, the training is often faster and more reliable. To formalize SGD, let us consider the loss function as the addition of losses over all the n batches of the training set, so that:

$$L(\boldsymbol{\theta}) = \sum_{j=1}^{n} \sum_{k=1}^{B} L_{kj}(\boldsymbol{\theta}), \tag{31}$$

where L_{kj} is the loss function for the k-th element of the j-th batch. The standard gradient descent algorithm optimizes the loss function by updating the parameters of the neural network using:

$$\boldsymbol{\theta}_{i+1} = \boldsymbol{\theta}_i - \eta \nabla L(\boldsymbol{\theta}_i) = \boldsymbol{\theta}_i - \eta \sum_{j}^{n} \nabla L_j(\boldsymbol{\theta}_i), \tag{32}$$

where η is the learning rate. The SGD method updates the parameters following the same idea but calculating the gradient using only a single batch:

$$\boldsymbol{\theta}_{i+1} \approx \boldsymbol{\theta}_i - \eta \nabla L_j(\boldsymbol{\theta}_i) = \boldsymbol{\theta}_i - \eta \sum_{k=1}^{B} \nabla L_{jk}(\boldsymbol{\theta}_i), \qquad j = 1, \ldots, n. \tag{33}$$

The learning rate is used to tune the step size defined by the gradient, which is often not optimal unless one is very far from the optimal solution. The learning rate can be kept fixed or it can be changed according to our requirements. It is usually tuned to find a compromise between the accuracy of the network and the speed of convergence. If η is too large, the steps will be too large and the solution could potentially overshoot the minimum. On the contrary, if it is too small it will take too many iterations to reach the minimum. In recent years, adaptive methods like Adam (Kingma and Ba 2014) or RMSProp (Tieleman and Hinton 2012) have been developed to automatically tune individual learning rates for each variable. These are still first-order algorithms in which some second-order information from the Hessian is estimated using consecutive iterations.

5.3.3 Backpropagation

The gradient of the loss function with respect to the free parameters of the neural network needed during training is obtained via the backpropagation algorithm (LeCun et al. 1998). The composite character of neural networks makes the calculation of these gradients easier than for a general nonlinear function because one can recursively apply the chain rule. To demonstrate this, it is advisable to start with the simple case of two layers:

$$L = g(\mathbf{x}, \mathbf{u}) \tag{34}$$

$$\mathbf{u} = f^{(2)}_{\boldsymbol{\theta}^{(2)}}(\mathbf{v}) \tag{35}$$

$$\mathbf{v} = f^{(1)}_{\boldsymbol{\theta}^{(1)}}(\mathbf{x}), \tag{36}$$

where $\mathbf{u}$ and $\mathbf{v}$ are used as intermediate results of hidden layers. The gradient of the loss function with respect to both sets of $\boldsymbol{\theta}$ parameters are given by:

$$\frac{\partial L}{\partial \boldsymbol{\theta}^{(1)}} = \frac{\partial L}{\partial \mathbf{u}} \frac{\partial \mathbf{u}}{\partial \mathbf{v}} \frac{\partial \mathbf{v}}{\partial \boldsymbol{\theta}^{(1)}} \tag{37}$$

$$\frac{\partial L}{\partial \boldsymbol{\theta}^{(2)}} = \frac{\partial L}{\partial \mathbf{u}} \frac{\partial \mathbf{u}}{\partial \boldsymbol{\theta}^{(2)}}. \tag{38}$$

The case of three layers is similarly given by:

$$L = g(\mathbf{x}, \mathbf{u}) \tag{39}$$

$$\mathbf{u} = f^{(3)}_{\boldsymbol{\theta}^{(3)}}(\mathbf{v}) \tag{40}$$

$$\mathbf{v} = f^{(2)}_{\boldsymbol{\theta}^{(2)}}(\mathbf{w}) \tag{41}$$

$$\mathbf{w} = f^{(1)}_{\boldsymbol{\theta}^{(1)}}(\mathbf{x}) \tag{42}$$

The gradients are given by:

$$\frac{\partial L}{\partial \boldsymbol{\theta}^{(1)}} = \frac{\partial L}{\partial \mathbf{u}} \frac{\partial \mathbf{u}}{\partial \mathbf{v}} \frac{\partial \mathbf{v}}{\partial \mathbf{w}} \frac{\partial \mathbf{w}}{\partial \boldsymbol{\theta}^{(1)}} \tag{43}$$

$$\frac{\partial L}{\partial \boldsymbol{\theta}^{(2)}} = \frac{\partial L}{\partial \mathbf{u}} \frac{\partial \mathbf{u}}{\partial \mathbf{v}} \frac{\partial \mathbf{v}}{\partial \boldsymbol{\theta}^{(2)}} \tag{44}$$

$$\frac{\partial L}{\partial \boldsymbol{\theta}^{(3)}} = \frac{\partial L}{\partial \mathbf{u}} \frac{\partial \mathbf{u}}{\partial \boldsymbol{\theta}^{(3)}} \tag{45}$$

In general, except for the first element in both previous equations, the rest of the terms of the shape $\partial \mathbf{u}/\partial \mathbf{v}$ are the Jacobian matrices. Therefore, the backpropagation can be understood as the multiplication of Jacobian matrices of the effect of each individual layer. The algorithm can be implemented with relative simplicity by just multiplying Jacobian matrices when traversing the neural network in the backward direction (that is precisely the reason for the name of the algorithm). Note that this calculation is also very efficient because one can store precomputed products of Jacobian matrices and use them afterward.

To efficiently calculate all gradients, one starts by computing the gradient $\partial L/\partial \mathbf{u}$ with the loss function and the last layer of the network. Then one goes to the previous layer and computes the Jacobians $\partial \mathbf{u}/\partial \mathbf{v}$ and $\partial \mathbf{u}/\partial \boldsymbol{\theta}^{(3)}$. Both Jacobians are used to update the gradients with respect to the variables $\boldsymbol{\theta}^{(2)}$ and $\boldsymbol{\theta}^{(3)}$ respectively. The procedure is iterated until the first layer is found. In practice, this process is currently done with automatic differentiation techniques, implemented in packages like PyTorch[18] (Paszke et al. 2019), Tensorflow[19] (Abadi et al. 2015) or JAX[20] (Bradbury et al. 2018). Because these tools deal with the product of Jacobians in the neural

[18] https://pytorch.org/.

[19] https://www.tensorflow.org/.

[20] https://github.com/google/jax.

network graph, they allow the user to easily define flexible neural network architectures tailored to specific needs.

5.3.4 Vanishing gradient problem

The way neural networks are trained suffers from a problem known as the vanishing gradient problem (e.g., Kolen and Kremer 2001). This was the reason why the field of artificial neural networks was somehow stalled during the years before the first decade of the 21st century. If one considers typical nonlinear activation functions like the $\tanh(x)$, their derivative becomes very close to zero if the input is relatively far from zero. Consequently, the Jacobian of this activation function becomes very small and the gradient is not propagated backwards to the previous layers. As an effect, the gradient of the loss function with respect to the first layers of the neural networks using *tanh*(x)-like activation function rapidly becomes zero. As a result, the stochastic gradient descent cannot produce any correction on their weights. As commented before, new activation functions like $\mathrm{ReLU}(x)$ largely solve this problem because their derivative does not saturate.

5.4 Bag-of-tricks as of 2023

5.4.1 Initialization

Tuning the initial value of the weights and biases of all the connections turned out to be crucial for the success of deep learning. The aim of the initialization is to avoid the explosion or vanishing of the layer activations so that gradients can seamlessly be backpropagated and producing changes in all the layers of the model. If symmetric activation functions like *tanh* are used, Glorot and Bengio (2010) noticed that good results are found when initializing weights with a uniform distribution bounded in the interval $[-\sqrt{6}/\sqrt{n_{\mathrm{in}}+n_{\mathrm{out}}}, \sqrt{6}/\sqrt{n_{\mathrm{in}}+n_{\mathrm{out}}}]$, where n_{in} and n_{out} are the number of input and output connections at a given layer, respectively. This is currently known as *Xavier* initialization. For asymmetric activation functions, He et al. (2015) checked that initializing weights from a normal distribution with zero mean and variance $2/n_{\mathrm{in}}$ can be efficiently used to train very deep neural networks. This is currently known as the *Kaiming* initialization.

5.4.2 Augmentation

The supervised training of deep neural networks often requires a large number of examples in the training set. Many times, especially in science, building such large databases is unfeasible simply because of the lack more training examples. In such a case, one can apply augmentation techniques as a remedy to artificially increase the training set. Rotations, reflections, changes in contrast, and many other such transformations can produce new training cases that produce a more stable result and better generalization after training.

5.4.3 Regularization and overfitting

Because of the large number of free parameters, especially in very deep CNNs, overfitting can be a problem. One would like the network to generalize well and avoid any type of "memorization" of the training set. There is increasingly stronger empirical and theoretical evidence showing that stochastic gradient descent methods, specific neural architectures, and the overparameterization of very large models leads to flat minima in the loss function that automatically produce good generalization (e.g., Hochreiter and Schmidhuber 1997a; Barrett and Dherin 2020). In other words, the non-convex optimization problem is plagued with local minima but all of them are equally good in their generalization properties.

One could argue that deep neural networks seem to be self-regularizing. But, in those cases in which overfitting is found, there are a few ways to introduce extra regularization during training. Many of them can be understood as an addition of a prior term in the loss function so that one optimizes for the maximum a posteriori solution instead of the maximum likelihood. The most used ones are weight decay and dropout. Weight decay consists of forcing the weights of the neural network to be small. Large weights tend to produce neural networks that are very specialized to the training data and do not generalize well. For this reason, one typically adds an ℓ_2 (also known as Tikhonov) regularization term like the following:

$$L_{\text{regularized}} = L + \lambda |\boldsymbol{\theta}|^2. \tag{46}$$

The strength of the regularization is controlled by the hyperparameter λ. Dropout consists of randomly removing connections among neurons in the neural network with probability p. This makes the training noisier but introduces a certain regularization by sparsifying the weights. In essence, neural networks learn how to solve the problem at hand even with random perturbations to the architecture.

5.4.4 Normalization

Several techniques have been described in the literature to accelerate the training of CNNs and also to improve generalization. Batch normalization (Ioffe and Szegedy 2015) is a very convenient and easy-to-use technique that consistently produces large accelerations in the training. It works by normalizing every batch to have zero mean and unit variance. Mathematically, the input is normalized so that:

$$\begin{aligned} y_i &= \gamma \hat{x_i} + \beta \\ \hat{x_i} &= \frac{x_i - \mu}{\sqrt{\sigma^2 + \epsilon}}, \end{aligned} \tag{47}$$

where μ and σ are the mean and standard deviation of the inputs on the batch and $\epsilon = 10^{-3}$ is a small number to avoid underflow. The parameters γ and β are learnable parameters that are modified during the training. Although batch normalization can stabilize and accelerate training, it is true that it requires the usage of relatively large batches so that the statistics μ and σ are not too noisy. Other variants of normalization

have also been developed: layer normalization, instance normalization, group normalization, ...[21]

We caution against the liberal use of batch normalization in physics applications (especially for regression) without careful testing. If a feedforward network $F : \mathbf{x} \longrightarrow \mathbf{y}$ is thought of as a mapping between two dimensional quantities, the use of batch norm essentially means the units in which the inputs are provided changes for every batch.

5.4.5 Residual blocks and skip connections

Very deep networks usually saturate during training, producing higher errors than shallow networks because of difficulties during training (fundamentally produced by the vanishing gradient problem). Residual networks He et al. (2016) came to the rescue by obtaining state-of-the-art results with exceptionally deep networks without adding any extra parameters and with practically the same computational complexity. It is based on the idea that if $y = F(x)$ represents the desired effect of the block on the input x, it is much simpler for a network to learn the deviations from the input. This residual mapping works then by rewriting $y = x + R(x)$, with $R(x)$ a new neural network that describes the residual. Skip connections are specific types of residual connections in which intermediate features of the neural network are added or concatenated in later stages of the network (see the U-Net architecture of Fig. 7). They also help in propagating gradients to the initial layers of the neural network.

5.4.6 Specialized hardware

The embarrassingly parallel character of the operations to be carried out in a layer of a neural network (for instance, convolutions with different kernels can be carried out simultaneously without any dependence) has opened up the possibility of using specific hardware to accelerate the calculations. GPUs were traditionally architected for parallel graphics rendering (using fragment shaders). They are optimized for Single Instruction Multiple Data (SIMD) processing. This type of parallel programming paradigm is suited for application to large-scale scientific datasets, and for dense matrix multiplication. This means GPUs are ideal for accelerating deep neural networks, giving increases in the computation power of more than an order of magnitude with respect to general purpose CPUs. Tensor Processing Units (TPU) are even more specialized hardware that are, in essence, very fast matrix multipliers. Recently, even optics-based computation hardware has been proposed, with the promise to accelerate some computations by orders of magnitude at very reduced power consumption (Miscuglio and Sorger 2020).

It has also been verified that deep neural networks are especially tolerant to floating point errors so that they can be easily (and routinely) trained in single-precision. Even half-precision can be used, provided one does the backpropagation in single-precision. Specialized GPUs and TPUs can accelerate half-precision calculations by a large factor when compared with single-precision.

[21] See https://bit.ly/3XleCff.

6 Unsupervised deep learning

One of the weakest points of all linear methods described in the previous sections is that they rely only on the information provided by second-order statistics (correlation). Therefore, they cannot efficiently describe a dataset which is lying in a nonlinear manifold of the original high-dimensional space. We expect this to be true in general, so relying on nonlinear models has become a necessity. Several unsupervised nonlinear models were developed in the first years of the century: locally linear embedding (LLE; Roweis and Saul 2000), Isomap (Tenenbaum et al. 2000), a kernelized version of PCA (Schölkopf et al. 1998), self-organizing maps (SOM; Kohonen 2001), autoassociative neural networks (Bourlard and Kamp 1988) and t-SNE (Hinton and Roweis 2002). Only the last three methods have been used in solar physics but without much continuity. However, the landscape in recent years has changed completely thanks to the deep learning revolution. It is now possible to train excellent generative models that capture the statistical properties of a training set and we should expect this line of research to produce very interesting applications in solar physics.

6.1 Self-organizing maps

A self-organizing map[22] is a specific type of neural network that is trained unsupervisedly. A SOM is a way to project a high-dimensional dataset into a two-dimensional space by keeping, as much as possible, the topological information present in the original space. It consists of a predefined set of $N \times N$ neurons that are connected locally. The training is done by competitive learning starting from a random initial distribution of weights. Weights are updated after each observation is used by computing the neuron that is closer (typically in Euclidean distance) to the observation. The information of the neuron is then propagated to the neurons around within a predefined distance. One of the problems of this training is that it results in an unpredictable distribution of classes along the whole map. However, we point out that reproducibility can easily be solved by fixing the random seed used for training. It was used by Asensio Ramos et al. (2007a) to classify profiles of the Mn I line whose Stokes I profile is especially sensitive to the magnetic field strength. SOMs were later used by Asensio Ramos (2012) to classify profiles in IMaX (Martínez Pillet et al. 2011) observations and they also proposed them as a poor's man inversion method with reduced precision because it is fundamentally a classification-based inversion. Although self-organizing maps look promising for classification purposes, the lack of control of the output reduces their attractiveness.

6.2 t-SNE

Student-t Stochastic Neighbor Embedding (t-SNE) is a nonlinear dimensionality reduction method[23] that has had some success in recent years. The idea is to embed

[22] An implementation can be found in https://github.com/bougui505/quicksom.

[23] t-SNE is available on the `scikit-learn` Python package.

high-dimensional data for visualization in a low-dimensional space of two or three dimensions, which are especially suited for human understanding. It models each high-dimensional object by a two- or three-dimensional point in such a way that similar objects are modeled by nearby points and dissimilar objects are modeled by distant points with high probability. Thanks to the perplexity hyperparameter, one can make the mapping focus more on global or local properties of the observations. This multi-scale characteristics makes t-SNE a good candidate for exploring purposes. However, contrary to PCA, t-SNE does not set up a basis. Mapping new observations requires training the algorithm from scratch. It has been used by Panos and Kleint (2020) for the classification of Mg II line profiles. t-SNE can reliably distinguish between profiles associated with flaring regions and non-flaring regions. Additionally, it has been used by (Verma et al. 2021) for classifying Hα profiles and identifying those that are suitable for a simple inversion method based on the cloud model. Both works demonstrate that t-SNE is promising for understanding the general picture of large observations. However, as any unsupervised method, this interpretation can only be done a posteriori.

6.3 Mutual information

Panos et al. (2021) explored the use of neural networks to compute the mutual information between pairs of spectral lines observed with the IRIS satellite. Mutual information can be seen as a generalization of correlation.[24] For two random variables X and Y, the mutual information measures the difference between the joint distribution $p(x, y)$ and the product of their marginal distributions $p(x)p(y)$. Panos et al. (2021) showed that an encoder-type neural network can be trained to measure the mutual information. This training proceeds by using the same neural network to encode two spectral lines observed at the same pixel, which are samples from the joint distribution. The same neural network is used to encode two spectral lines from different pixels, which are seen as samples from the marginal distributions. By maximizing the distance between both encodings, the neural network learns how to approximate the mutual information. After training such an architecture with millions of IRIS profiles, they found that lines are weakly correlated in quiet conditions. The coupling strongly increases in flaring conditions, with Mg II and C II having the strongest coupling. Panos and Kleint (2021) used this tool to analyze in detail the full atmospheric response during flares. This tool is very promising for the study of multispectral data.

6.4 Autoencoders

Perhaps the most promising nonlinear dimensionality reduction are autoencoders[25] (AE), also known in the past as autoassociative neural networks (AANNs; Socas-Navarro 2005a). They were not very often used because of their inherent computational burden, given that one has to train a neural network for every new

[24] An implementation can be found in https://github.com/gtegner/mine-pytorch.

[25] An implementation can be found in https://github.com/dariocazzani/pytorch-AE.

type of observation that one needs to analyze. This is currently not a problem because of the availability of libraries for training neural networks and the powerful hardware to which we have access. However, even during the first decade of the 2000 s, training these neural networks was still problematic. AANNs are a special case of an encoder-decoder fully connected neural network. For the case of analyzing Stokes profiles, the input Stokes profiles are encoded by decreasing the size of the layers until a bottleneck layer of only d neurons is found. d is the expected intrinsic dimensionality of the Stokes profiles. They are again expanded in the decoder part until recovering the original size of the Stokes profiles. They are trained by forcing the network to output exactly the same profiles used as input. This way, the neural network has to compress the relevant information for each Stokes profile into only d numbers. Socas-Navarro (2005a) showed a comparison of AANNs and PCA. Given the nonlinear character of AANNs, they are able to much better reconstruct a set of Stokes profiles using a lower dimensionality.

We anticipate that, in the current era of deep learning, AE will find a central role in many fields of solar physics, especially those related with spectroscopy and spectropolarimetry, although imaging could certainly obtain gains. The projection of the observations into a latent space of reduced dimensionality introduces a strong regularization, that can be efficiently exploited by many inversion methods. The first applications of modern AEs are very recent. Sadykov et al. (2021) used them to show that the spectroscopic data of the Mg II line observed withe the NASA's IRIS satellite can be compressed by a factor of 27 without any relevant impact on the line profiles. Additionally, the authors find that the features found by the AE are interpretable. More recently, Díaz Baso et al. (2022) use an AE to compress Stokes I profiles to facilitate the computation of uncertainties during the inference process using a Bayesian framework (see Sect. 7.7.2 for more details).

6.5 Generative models

Generative models are probabilistic models, $p(\mathbf{X})$, that can approximate the distribution of objects of interest, $\mathbf{X}$, given a sufficiently large training set while being accompanied with an efficient way of sampling from $p(\mathbf{X})$. As such, they can be used as priors for $\mathbf{X}$ in any subsequent inference process. Modern generative models, especially for objects of large dimensionality like images, are either based on variational autoencoders (VAE; Kingma and Welling 2014), generative adversarial networks (GAN; Goodfellow et al. 2016), normalizing flows (NF; Dinh et al. 2014) or denoising diffusion probabilistic models (DDPM; Ho et al. 2020). A diagram with the specific architecture of each generative model is shown in Fig. 6. All of them can be seen as an instance of a latent-variable model (displayed as red blocks). In these models, we assume the existence of a hidden latent variable with a dimensionality that can be equal to or smaller than that of the signal of interest. This latent variable is often extracted from very simple probability distributions (Gaussian noise in many cases) and transformed, thanks to the action of a neural network, into samples from the distribution of interest.

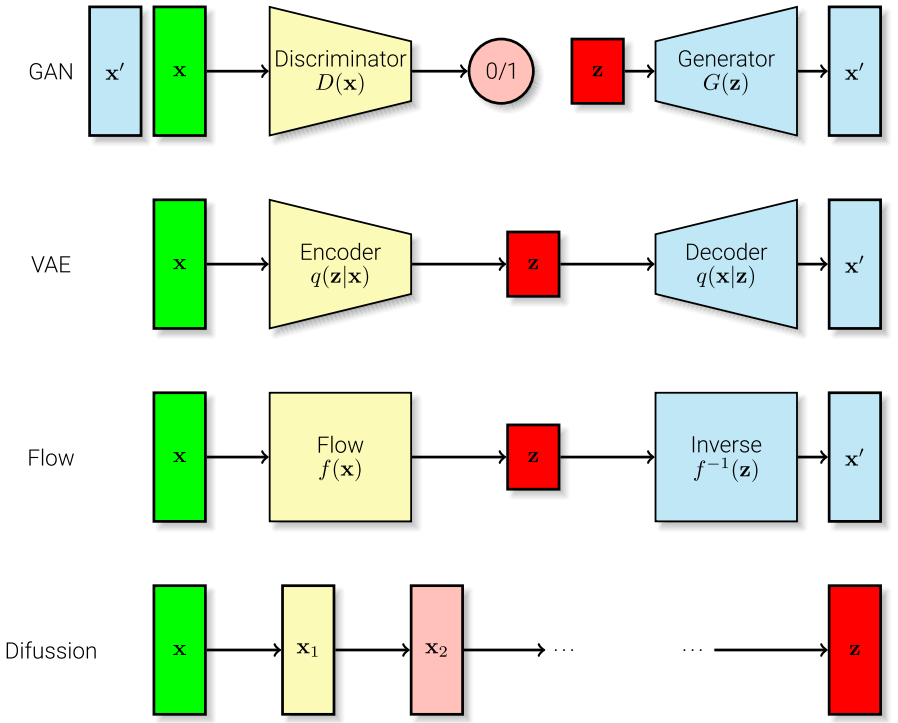

Fig. 6 Overview of the most successful nonlinear generative models for signals in high dimensions (adapted from https://lilianweng.github.io/posts/2021-07-11-diffusion-models)

6.5.1 Generative adversarial networks

Generative adversarial networks[26] (GAN) have had a huge impact on image generation, arts, language, and on some fields of research. They are based on two networks (see the upper row of Fig. 6): a generator $G(\mathbf{z})$, that maps the latent variable into the signals of interest, and a discriminator $D(\mathbf{x})$ that tells whether a sample $\mathbf{x}$ is coming from the distribution of interest or not. Both neural networks are trained simultaneously using adversarial training (Goodfellow et al. 2016).

Despite the huge impact in many fields, the impact in solar physics has been somewhat reduced. Kim et al. (2019) proposed conditional GANs for the generation of artificial magnetograms from STEREO data. The interest of such an approach is that once trained, GANs can generate artificial magnetograms on the far side of the Sun. They can be compared with current observations carried out with the Polarimetric and Helioseismic Imager (PHI) on Solar Orbiter (Solanki et al. 2020). The quality of farside magnetograms is still reduced, even after the improvements provided by Felipe and Asensio Ramos (2019) and Broock et al. (2022). Kim et al. (2019) trained the generator by using extreme UV data from AIA and magnetograms from HMI, both observed on the near side. The generated magnetograms in active

[26] An implementation can be found in https://github.com/eriklindernoren/PyTorch-GAN.

regions look very similar to the target ones while also providing very strong correlations in the total unsigned magnetic flux. More daunting is the task of correctly generating the polarity structure of the active regions, whose information is absent or barely present in the EUV images. According to Kim et al. (2019), their GAN is able to correctly produce Hale's law, purely learned from the data.

The reverse process, to produce EUV images from magnetograms, was approached by Park et al. (2019) with some success using GANs. Trained again with SDO data, the model is able to produce brightenings in all AIA filters in active regions, which compares well with the real data. In filters like 171 Å which show conspicuous loops, the GAN has a hard time reproducing them probably because the connectivity information is not present in the magnetograms.

Shin et al. (2020) developed a model that generates artificial magnetograms from Ca II K images. They improved over previous works by using a training scheme that takes into account both large-scale and small-scale properties of the images simultaneously, as proposed by Wang et al. (2017). This allows them to generate high-resolution magnetograms with sizes up to 1024×1024 pixels. Again, the polarity structure of very active regions is correctly captured by the model even though this information is probably absent from the Ca II images. The only sensible explanation for this is that this information is extracted from the statistical properties of the training set. The authors also point out that the model does a bad job on the quiet regions of the Sun.

An obvious question that arises for the image-generation models that we have discussed is what is their final purpose. It seems obvious that simply generating the images might have limited applicability, except perhaps homogenizing very long baseline datasets. On the contrary, having an efficient generative model for such complex processes will surely become key for future research. Generative models map a latent vector $\mathbf{z}$ of reduced dimensionality onto a complex and large image I. Consequently, introducing a pretrained generative model in an elaborate inference scheme is a very good prior and can strongly inform the output and lead to very efficient inference methods directly from images. For instance, one can think of data assimilation methods in which a physical simulation is set up to explain a specific observation. In this case, the physical model is very efficiently related to the observation via the latent space, which automatically avoids outliers.

6.5.2 Variational autoencoders

Standard autoencoders, as shown in Sect. 6.4, are not generative models because there is no way of sampling from the distribution. A variational autoencoder[27] (VAE; Kingma and Welling 2014) is a modification of a standard autoencoder that works as a generative model (see the second row of Fig. 6). To this end, the latent space is forced to have a fixed probability distribution during training. Once trained, sampling from this fixed distribution (often a Gaussian distribution) and passing the samples through the decoder, produces samples of the variable of interest according to the prior. A VAE was used by Panos et al. (2021) as a means of compressing Mg II

[27] An implementation can be found in https://github.com/dariocazzani/pytorch-AE

profiles. When the VAE is trained with line profiles from the so-called quiet Sun, it represents a very efficient outlier detector. Out-of-distribution profiles (i.e., flaring profiles) cannot be efficiently reproduced by the VAE. Therefore, if the difference between the reconstructed profile and the original profile is large, one can safely say that the profile is not coming from the inactive Sun.

6.5.3 Normalizing flows

Another powerful way of producing samples from the posterior distribution is via normalizing flows[28] (NF), which are a very flexible, tractable, and easy-to-sample family of generative models, that can approximate complex distributions. Simply put, an NF is a transformation of a simple probability distribution (often a multivariate standard normal distribution, with zero mean and unit covariance) into the desired probability distribution (see the third row of Fig. 6). Normalizing flows accomplish this by the application of a sequence of invertible and differentiable variable transformations. Let us assume that $\mathbf{Z}$ is a d-dimensional random variable with a simple and tractable probability distribution $q_{\mathbf{Z}}(\mathbf{z})$, with the condition that it is fairly straightforward to sample. Let $\mathbf{X} = f(\mathbf{Z})$ be a transformed variable, with a function f that is invertible. If this condition holds, then $\mathbf{Z} = g(\mathbf{X})$, where $g = f^{-1}$. The change of variables formula states that the probability distribution of the transformed variable is given by:

$$q_{\mathbf{X}}(\mathbf{x}) = q_{\mathbf{Z}}(g(\mathbf{x}))\left|\det\left(\frac{\partial g(\mathbf{x})}{\partial \mathbf{x}}\right)\right|. \tag{48}$$

The term $\partial g(\mathbf{x})/\partial \mathbf{x}$ is the Jacobian matrix and takes into account the change of probability volume during the transformation. Its role is to force the resulting distribution to be a proper probability distribution with unit integrated probability. Since the transformation is invertible, the equality $\partial g(\mathbf{x})/\partial \mathbf{x} = (\partial f(\mathbf{z})/\partial \mathbf{z})^{-1}$ holds, so that one can rewrite the previous expression as:

$$q_{\mathbf{X}}(\mathbf{x}) = q_{\mathbf{Z}}(\mathbf{z})\left|\det\left(\frac{\partial f(\mathbf{z})}{\partial \mathbf{z}}\right)\right|^{-1}. \tag{49}$$

Designing an invertible transformation that can be trained to produce generative models over complex datasets is difficult. For this reason, normalizing flows make use of the fact that the composition of invertible transformations is also invertible. Then, if $f = f_M \circ f_{M-1} \circ \cdots \circ f_1$, the transformed distribution is

$$q_{\mathbf{X}}(\mathbf{x}) = q_{\mathbf{Z}}(\mathbf{z}) \prod_{i=1}^{M}\left|\det\left(\frac{\partial f_i(\mathbf{y_i})}{\partial \mathbf{y_i}}\right)\right|^{-1}, \tag{50}$$

where $\mathbf{y}_i = f_{i-1} \circ \cdots \circ f_1(\mathbf{z})$ and $\mathbf{y}_1 = \mathbf{z}$. Compositional invertible transformations have made it possible to define very flexible normalizing flows through the use of deep neural networks.

[28] https://github.com/bayesiains/nflows.

Despite their potential as a flexible probabilistic generative model, they have not been used in solar physics for this purpose yet. We refer the reader to Sect. 7.7.2 for a discussion on how NFs have been applied for the acceleration of Bayesian inference from spectropolarimetric observations by directly fitting the posterior distribution.

6.5.4 Denoising diffusion probabilistic models

Denoising diffusion models[29] (DDPM; Ho et al. 2020) are based on two chains of processes (see Fig. 6). The first one adds a small amount of noise to a certain sample from the variable of interest. When this noise addition is repeated many times, the final result cannot be distinguished from pure noise and is assumed to be the latent variable. This process is, obviously, easy to simulate. The inverse process takes the latent variable and proposes a neural network that "cleans" the noise, trying to undo what the first process did to the signal. This generative model is at the base of the most recent image generative models, of enormous success when coupled with powerful language models. We still need to see applications of DDPMs as prior for solar data.

7 Applications of supervised deep learning

The vast majority of applications of nonlinear models in supervised training are based on CNNs. The models have been increasing in complexity in the last few years, motivated by the success of CNNs in learning directly from the data. In the following, we describe relevant applications to different subfields of solar physics.

7.1 Segmentation of solar images

Deep learning has produced a huge advance in the dense (per pixel) classification of solar images, of special relevance due to the large amount of synoptic solar observations that we currently have. Automatic detection and segmentation of solar structures in images could allow us to build databases for an enormous amount of images. Detecting sunspots, flares, coronal holes, and other structures are potential candidates for such applications. CNNs have recently been used for the identification of CHs. Illarionov and Tlatov (2018) proposed a U-Net architecture[30] as proposed by Ronneberger et al. (2015) (see Fig. 7) to identify CHs on solar AIA/SDO images obtained in the 193 Å wavelength. Illarionov and Tlatov (2018) trained the model with 2385 binary maps from the Kislovodsk Mountain Astronomical Station. The output of the U-Net is a binary image that tells whether the pixel belongs to a coronal hole or not. The training is carried out using the binary cross-entropy (BCE) as a loss function:

[29] https://github.com/lucidrains/denoising-diffusion-pytorch.

[30] https://lmb.informatik.uni-freiburg.de/people/ronneber/u-net.

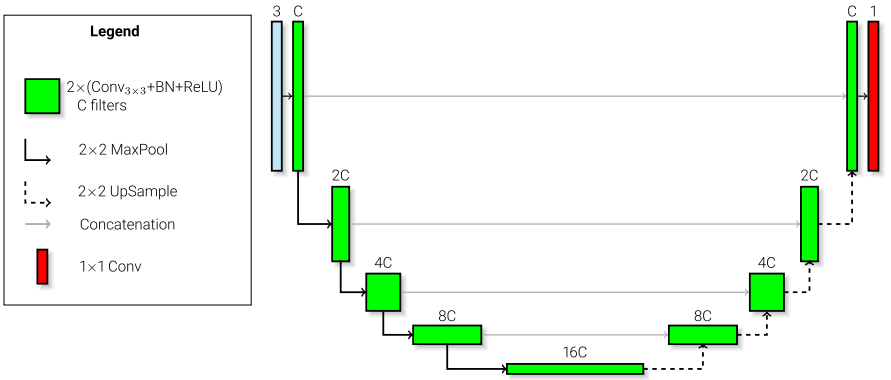

Fig. 7 Schematic drawing of the encoder-decoder U-Net architecture. In this case, the input has 3 channels, while the output contains only one channel. The number of channels after the first convolutional layer is C. Green blocks summarize the application of a convolutional layer with a 3×3 kernel, followed by batch normalization and a ReLU. These operations are repeated twice. Solid arrows correspond to the MaxPool operation, while dashed arrows refer to bilinear upsampling of the feature images. Grey arrows refer to skip connections that are simply concatenated in the decoder

$$L = -\sum_i y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i), \tag{51}$$

where y_i is the target label for the i-th pixel and $\hat{y}_i$ is the prediction of the network. The results of the CH identification were compared with feature maps of other methods, such as CHIMERA (Coronal hole identification via multi-thermal emission recognition algorithm; Garton et al. 2018) and SPoCA, from January 2017 to July 2018. One of their conclusions is that the U-Net architecture produces segmentation maps that are more consistent than those of SPoCA. By comparing the area variation of the CH, they observe that CHIMERA and U-Net show similar results and the two methods have a correlation coefficient of 0.76. In a follow-up study, Illarionov et al. (2020) extended the identification of the CH for synoptic maps and they constructed a catalogue for 2010-2020 based on the AIA/SDO 193 Å data. The Solar Corona Structures Segmentation Network (SCSS-Net) was also developed by Mackovjak et al. (2021), again inspired by the U-Net architecture, for the dense segmentation of solar images and the localization of CH and AR. U-Nets were also used by Jiang et al. (2020) to identify and track solar magnetic flux elements observed in magnetograms. This will largely facilitate tracking of small-scale magnetic elements, something that is currently done with ad-hoc techniques and large human intervention (e.g., Gošić et al. 2014). Since tracking involves some degree of time coherence on the labeling of the elements, we anticipate that taking into account the time evolution could produce a large improvement over single-frame segmentation (e.g., Ventura et al. 2019).

Jarolim et al. (2021) used a CNN to identify the boundaries of CH using the seven extreme ultraviolet (EUV) channels of AIA/SDO as input, together with the line-of-sight magnetograms provided by the HMI/SDO. Their identification method is termed Coronal Hole RecOgnition Neural Network Over multi-Spectral-data

(CHRONNOS). Their analysis of the CNN concludes that the CNN has the ability to learn directly from multi-dimensional data and can identify CH and distinguish them from prominence channels. To this end, the CNN takes advantage of the shape, structural appearance, global context information, and the multi-wavelength representation.

Later, using three different and independent wavelengths, Baek et al. (2021) implemented the Single Shot MultiBox Detector (SSD) and the Faster Region-based Convolutional Neural Network (R-CNN), for the detection of the CH, prominences and sunspots. The training set is based on full-disk data from AIA/SDO and HMI/SDO between 2010 and 2019. The data cadence is 12 hr for sunspots and CH and four hours for the prominences. The events in each observed image was manually labelled, including the bounding boxes. The total number of images with coronal holes was 5085 (from the AIA 193 Å channel), those with sunspots was 4383 (from the intensity images of HMI/SDO) and those with prominences were 2926 (from the AIA/SDO 304 Å channel). Once trained, they checked that the models do a good job in locating the CH after a direct comparison with the HEK database.

Although several methods (some of them based on CNNs) have been developed in the recent years, we are still missing an estimation of the uncertainties in the segmentation of solar images. In a recent paper, Reiss et al. (2021) analyzes these uncertainties in the detection of CH boundaries. Nine automatic methods are compared using a CH from the southern hemisphere, close to the sun center, and observed for a couple of solar rotations. Multiple EUV wavelengths and measurements of the radial component of the photospheric magnetic field from the SDO spacecraft were used as preparation of the data to be used by different methods. The compared methods are ASSA-CH, CHIMERA (Garton et al. 2018), CHORTLE, CNN193 (Illarionov and Tlatov 2018), CHRONNOS (Jarolim et al. 2021), SPoCA-CH (Verbeeck et al. 2014), and SYNCH. They also evaluated the mean CH intensity in AIA 193A, the mean signed and unsigned line-of-sight magnetic field component (B_{LOS}), the degree of unipolarity, and the net open magnetic flux (sum of B_{LOS} over the CH area). They found that different methods produce significantly different outcomes. The differences are small in the center of the CH and they start to be larger when approaching the boundary of the CH. Differences in the shape of the CH and its physical properties are also found (see Fig. 8).

As a consequence, the choice of the method has a non-negligible impact on the predicted solar wind. As a final conclusion, one of the fundamental problems to characterize the uncertainties is the absence of a well-agreed definition of a CH (or, by extension, of any feature on the solar surface). We can only compare automatic methods with a segmentation made by eye by the observer. This manual segmentation can also depend on the wavelength used for its evaluation. We urgently need a community effort toward defining an agreed training set.

Dense segmentation of photospheric images has been pursued recently by Díaz Castillo et al. (2022), with the aim of classifying granular structures. The access to high-resolution images has shown the overwhelming complexity of granulation. One can only hope to understand the physical mechanisms by first applying a semantic segmentation of the images and isolating interesting phenomena (intergranular lanes, exploding granules, ...). Although it is still work in progress, Díaz Castillo et al.

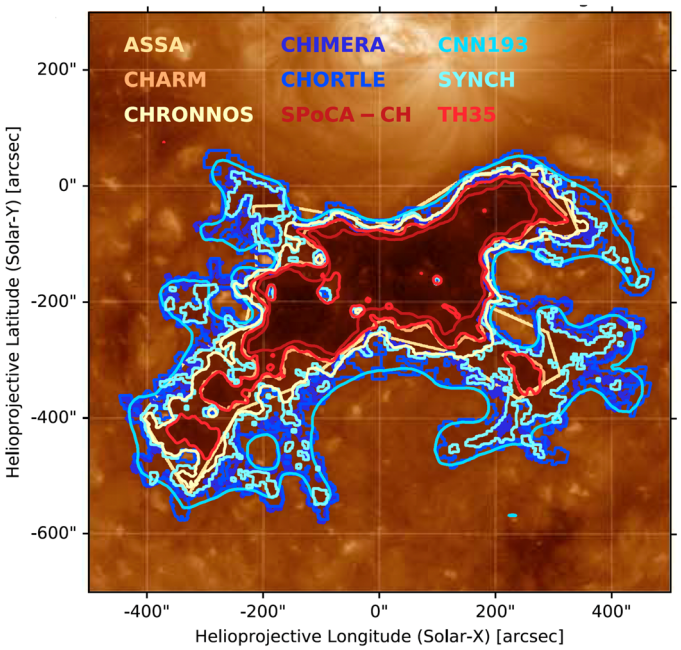

Fig. 8 A comparison of the estimated coronal hole maps from nine different automated detection schemes overlaid on the AIA 193 Å. Image reproduced with permission from Reiss et al. (2021), copyright by the author(s)

(2022) demonstrated that a U-Net is able to learn this segmentation problem and then apply it to large datasets to analyze the statistical properties of magnetoconvection.

7.2 Classification of solar images

Arguably one of the most used ways of understanding physical phenomena in the solar surface and the heliosphere has been by painfully classifying all events into different classes. This allows researchers to pinpoint typical properties of similar events and associate them with their physical properties. In the era of photographic or video images, this classification could be done by hand. However, the amount of data that we are currently generating, as well as the expected increase in data rates in the near future, is so large that we need the help of machines to classify all events. Armstrong and Fletcher (2019) used a deep neural network to automatically classify solar events in different classes: quiet Sun, prominence, filaments, sunspots, and flare ribbons. The trained neural network achieves an extremely high performance (close to 99.9%). They also demonstrate that transfer learning can be used in solar physics. Transfer learning is the idea of reusing complete, or parts of a neural network that have been previously trained and applying them to another problem. Transfer learning is part of many successful applications of deep learning in general and is based on the idea that the initial layers of a convolutional neural network are able to extract features from the images that are used by the last layers of the neural network

to carry out the classification. Armstrong and Fletcher (2019) successfully demonstrated that the first layers of a CNN trained with solar structures in one wavelength are able to extract features that can be used with images in a different wavelength.

Along the same line raised in the previous section, Armstrong and Fletcher (2019) also advocated for the generation of a huge and curated database of classified solar images. This is in parallel to similar efforts in the machine learning community like ImageNet (Deng et al. 2009), in which more than 14 million images have been labeled by hand using more than 20 thousand words describing the images. The solar ImageNet would be a large database of multiwavelength multi-instrument images ideally labeled by hand.

Apart from the pure classification of images, one of the potential applications of these neural networks is to automatically detect "interesting" events that are far from the typical cases. In the case of a classifier, a flag can be raised by the machine when it finds an event that is associated with similar probability to several classes. This means that the class is not known with certainty so it does not look like any of the examples in the training set. Another option to develop an outlier detector is by using a generative model (see Sect. 6.5) that learns how to produce images like those in the training set. If the properly trained generative model is able to correctly reproduce the input (some generative models are even able to output an estimation of the likelihood), the input is compatible with the trained data and cannot be considered an outlier. If it is not well reproduced (or the likelihood is very small), then it can be tagged as an outlier and a flag can be raised for later analysis.

Although not a whole-image classification scheme, MacBride et al. (2021) used a FCN classifier to identify and classify spectral line profiles. The aim was to rapidly cluster the profiles into different categories depending on the number of components in the emission peak and absorption dip present in the line. The model is able to determine the underlying properties of complex profiles, not only to identify the peak and dips in the profiles but also to classify sub-classes that will then be used to constrain better the fitting of a single or multiple velocities components inside the pixel. They tested the method using Ca II 8542 Å line profiles observed by the Interferometric BIdimensional Spectrometer (IBIS) using an uneven spectral sampling, with a higher density in the line core, as proof of concept of the model, and also as a benchmark for two-component atmospheric profiles studies that are commonly present in sunspot chromospheres (see Fig. 9).

7.3 Prediction of flares

Due to the consequences of an impact of a major solar flare on terrestrial space weather, there has been an increasing interest in applying statistical learning techniques for the prediction of flares (typically M-class flares and greater) and coronal mass ejections (CMEs). Since not all flares have associated CMEs (and vice versa),[31] forecasting of the two are considered related, but different problems.

[31] Sheeley et al. (1983) reported that every GOES X-ray flare (lasting six hours or longer) had an associated CME. Their data set comprised events observed between 1979 and 1981.

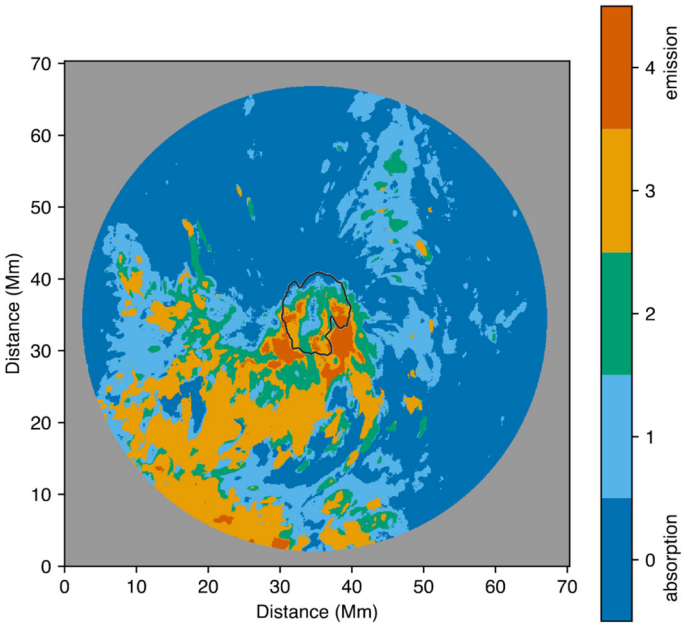

Fig. 9 SPectral classifications of an IBIS observation, where the color bar relates to the spectral shape classified, with '0' and '4' representing pure absorption and emission profiles, respectively. The umbra/penumbra boundary is highlighted using a black contour Image reproduced with permission from MacBride et al. (2021), copyright by the author(s)

In either case, the problem is usually posed as a classification problem. Given input parameters $\mathbf{x}$ sampled at time t_0, does a flare occur in the time period $t \in (t_0, t_0 + \Delta t]$, with Δt on the order of hours to days? Variations on this problem statement can include multiclass classification (e.g., whether there is an M- or X-class flare), or regression to predict the maximum soft x-ray flux (e.g., as measured by GOES XRS) in the time period of interest.

Early attempts at flare/CME prediction focused on the use of input features inspired by physics models (or heuristics) of how solar flares are thought to operate. Ample theoretical considerations, observational evidence, and numerical simulations support the commonly accepted picture that solar flares are powered by abrupt reconfigurations of the solar coronal magnetic field (see reviews by Priest and Forbes 2002; Shibata and Magara 2011) which results in the coronal magnetic field entering a lower energy state. The lowest energy state of the coronal magnetic field above an active region is the potential field configuration, which has zero current density, since $\mathbf{j} = \nabla \times \mathbf{B} = \nabla \times [-\nabla \Phi] = 0$ (e.g., see Altschuler and Newkirk 1969). Without available *free energy* (i.e., the magnetic energy in excess of the energy stored in a potential field configuration), an active region should not be flare-productive. This physical argument suggests the photospheric (since this is the layer for which magnetograms are most easily acquired) current density should have predictive power for flare prediction. A related quantity of interest is the twist parameter, which is the current density normalized by the magnetic field strength.

Falconer (2001) performed a pilot study of the association between CME-productivity and an AR's perceived non-potentiality with two quantities derived from vector magnetograms. The latter was visually assessed from the morphology of loops in Yohkoh X-ray images (e.g., the presence or absence of a sigmoid). The paper suggests that the length of the main polarity inversion line (PIL) and the net current (measured through one polarity) are quantitative indicators of CME productivity. However, only eight vector magnetograms covering three distinct ARs were available for this study. Falconer et al. (2002) extended the work using 17 vector magnetograms covering 12 ARs. In addition to the PIL length and net current, they examined a dimensionless twist parameter and reported all three are correlated with the flux content of the AR, and are correlated with CME productivity. To address the limitations imposed by the lack of regular vector magnetogram coverage available at the time, Falconer et al. (2003) developed a proxy for the main PIL length parameter using line-of-sight magnetograms from the Michelson Doppler Imager (Scherrer et al. 1995) onboard the ESA/NASA Solar & Heliospheric Observatory (SOHO; Domingo et al. 1995) mission. This work opened up the possibility to use MDI full-disk magnetograms (available at 90 min cadence) for assessing the CME productivity of ARs.

In a series of papers (Leka and Barnes 2003a, b; Barnes and Leka 2006; Leka and Barnes 2007), Leka & Barnes performed systematic analyses of how vector magnetogram-derived parameters such as current and twist are different between flaring and non-flare active regions. Of particular relevance is Leka and Barnes (2003a), in which they performed discriminant analysis on flaring and non-flaring regions. This is a linear model for binary classification. Suppose $\mathbf{x}$ is the feature vector (consisting of vector magnetogram-derived quantities) and $\overline{\mathbf{x}}^0$ and $\overline{\mathbf{x}}^1$ denotes the mean of $\mathbf{x}$ over the two separate populations (in this case, flaring and non-flaring active regions). The sign of the linear functional

$$f(\mathbf{x}) = \mathbf{x}\mathbf{C}^{-1}(\overline{\mathbf{x}}^0 - \overline{\mathbf{x}}^1) + \frac{1}{2}(\overline{\mathbf{x}}^0 - \overline{\mathbf{x}}^1)\mathbf{C}^{-1}(\overline{\mathbf{x}}^0 + \overline{\mathbf{x}}^1), \tag{52}$$

was used to classify whether an active region with feature vector $\mathbf{x}$ is in the flaring or non-flaring population. They computed discriminate functions for single variate as well as multivariate feature vectors. However, the data set available only included 24 blocks of roughly 1-hour long observations (spanning over 7 active regions and 10 C, M and X flares) from the University of Hawaii Imaging Vector Magnetograph. The data set was enough to establish that a small number of input parameters is insufficient to distinguish the two flare-active and flare-quiet active region populations with low error rates. With six input features (standard deviation of the horizontal magnetic vector, skew of vertical current density J_z, kurtosis of J_z, area of pixels with a shear angle greater than 80 deg, time rate of the change of the best-fit linear force-free parameter and the time rate of change of the mean unsigned normal flux density), they were able to construct a function $f(\mathbf{x})$ which linearly separates the two populations.

7.3.1 HMI era

The Leka and Barnes (2003a) study was limited by the quantity of data available and it is not clear how generalizable the results are when applied to other active regions. However, within the data set studied, it appears variables measuring the distribution of magnetic twist helps with discerning whether an active region is flare-quiet or productive.

The availability of regular vector magnetograms at 12 min cadence from SDO/HMI was a game changer. It enabled the curation of flare prediction datasets with hundreds and thousands of samples. The Space Weather HMI Active Region Patches (SHARPs) data product provides vector magnetograms in rectangular patches (on the CCD or on a longitude-latitude grid; see Fig. 10) following active regions as they emerge and rotate across the solar disk. The metadata for this data product includes quantities that summarize the size (in terms of flux content and area spanned), and magnetic twist (e.g., area-averaged current density) for each magnetogram in the temporal sequence (see Bobra et al. 2014, for the complete list). When used for problems like flare prediction, the SHARP parameters can be considered 'hand-engineered' features extracted by experts.

The paper by Bobra and Couvidat (2015) ushered in a new era for solar flare prediction. Their contributions to the field of flare prediction research are manyfold. This study was the first to use the HMI SHARP data set to demonstrate the potential utility of continuous vector magnetogram data for flare prediction. The paper introduced many standard data science practices to the solar physics community. This includes the practice of $n-$fold cross-validation (CV). Specifically, they evenly split the dataset into n tranches, picked the union of $n-1$ tranches to train a model, and then used the remaining tranche as a test set. They then rotated through the tranches, each time using a different tranche to be the test set. This allowed the model to be trained and tested n times. The spread of the evaluation metrics over the so-called $n-$folds provides a measure of the reliability of the metrics.

7.3.2 Evaluation metrics

As more research groups began to tackle the flare prediction problem, there emerged a need to standardize how flare prediction models are evaluated. Consider any binary classification problem. The aim is for the model to classify whether an element is in class A. For any binary classification problem (including flare prediction), consider the contingency Table 1. The contingency table completely specifies the joint probability density function (JPDF) regarding whether an event has occurred, and whether it was forecast to occur.

Metrics regarding the performance of the forecasting method are functions of the JPDF (see Table 2). For example, the recall (false alarm rate) is the conditional probability of a positive forecast, *given* the event did (not) take place. In contrast, the precision is the conditional probability that the event occurred, given a positive forecast was issued.

Which metric is the correct one to use? It depends on the goal and there is no single correct answer. For instance, the stakeholder of an operational space weather

Fig. 10 Space Weather HMI Active Region Patches (SHARPs) identified by a computer tracking algorithm. In this image, two SHARPs have been identified and are marked by rectangular bounding boxes

Table 1 Contingency table for binary classification

	Event: Yes	Event: No
Forecast: Yes	**TP**	***FP***
Forecast: No	**FN**	***TN***

We denote **TP**, ***TN***, ***FP*** and **FN** as the number of true positives, true negatives, false positives and false negatives, respectively

forecast may prioritize the need to minimize false positives, because false positives will trigger protocols (e.g., shutting down the power supply). For other stakeholders, avoidance of a false negative (i.e., a reliable *all-clear"* forecast) may be the priority (Barnes et al. 2016). If the stakeholders were solar physicists with research interests in flares and must submit their observing plans to instruments on a space-borne observatory (e.g. Hinode or IRIS) a day in advance, the cost for false alarms may be comparatively small.

When comparing different flare prediction models, one would ideally wish to compare models using identical test data sets. In practice, it is not possible without the coordination of research groups tackling the flare prediction problem due to differences in the choice of flares they include in the data sets, the choice of train/test set partitioning, etc. These choices impact the class imbalance between flaring and non-flaring events in the data sets used. Let the class imbalance be the ratio of non-events ($\boldsymbol{TN} + \boldsymbol{FP}$) to actual events ($\mathbf{TP} + \mathbf{FN}$). Evaluation metrics comprising sums

Table 2 Evaluation metrics for binary classification

Metric	Definition	Meaning	Range
Recall	$\frac{\mathbf{TP}}{\mathbf{TP}+\mathbf{FN}}$	P(Forecast:Yes|Event: Yes)	[0, 1]
Precision	$\frac{\mathbf{TP}}{\mathbf{TP}+\boldsymbol{FP}}$	P(Event:Yes|Forecast: Yes)	[0, 1]
Specificity	$\frac{\boldsymbol{TN}}{\boldsymbol{TN}+\boldsymbol{FP}}$	P(Forecast:No|Event: No)	[0, 1]
False Alarm Rate	$\frac{\boldsymbol{FP}}{\boldsymbol{TN}+\boldsymbol{FP}}$	P(Forecast:Yes|Event: No)	[0, 1]
Accuracy	$\frac{\mathbf{TP}}{\mathbf{TP}+\mathbf{FN}+\boldsymbol{TN}+\boldsymbol{FP}}$	P(Forecast: Yes & Event:Yes)	[0, 1]
Rate Correct	$\frac{\mathbf{TP}+\boldsymbol{TN}}{\mathbf{TP}+\mathbf{FN}+\boldsymbol{TN}+\boldsymbol{FP}}$	P(Forecast == Event)	[0, 1]
Critical Success Index	$\frac{\mathbf{TP}}{\mathbf{TP}+\boldsymbol{FP}+\mathbf{FN}}$	–	[0, 1]
Gilbert Skill Score	$\frac{\mathbf{TP}-\mathrm{CH}}{\mathbf{TP}+\boldsymbol{FP}+\mathbf{FN}-\mathrm{CH}}$	CSI excluding chance hits	[0, 1]
Heidke Skill Score (v1)	$\frac{\mathbf{TP}}{\mathbf{TP}+\mathbf{FN}}-\frac{\boldsymbol{FP}}{\mathbf{TP}+\mathbf{FN}}$	Recall$\times(2-$ Precision$^{-1})$	$(-\infty, 1]$
Heidke Skill Score (v2)	$\frac{\mathbf{TP}+\boldsymbol{TN}-\mathrm{E}}{\mathbf{TP}+\mathbf{FN}+\boldsymbol{TN}+\boldsymbol{FP}-\mathrm{E}}$	–	[0, 1]
True Skill Statistic	$\frac{\mathbf{TP}}{\mathbf{TP}+\mathbf{FN}}-\frac{\boldsymbol{FP}}{\boldsymbol{TN}+\boldsymbol{FP}}$	Recall - False Alarm Rate	$[-1, 1]$

Refer to Table 1 for the definition of classes. In the definition for the Gilbert Skill Score, CH (chance hits) is the Accuracy for a random forecast model. The probability that a random forecast outputs a positive is uncorrelated with the underlying probability of the event. Hence, the joint probability for the Accuracy can be factored, giving $\mathrm{CH} = P(\text{Forecast: Yes}) \times P(\text{Event:Yes}) = \frac{(\mathbf{TP}+\boldsymbol{FP})}{n}\frac{(\mathbf{TP}+\mathbf{FN})}{n}$, where $n = \mathbf{TP} + \boldsymbol{FP} + \mathbf{FN} + \boldsymbol{TN}$. In the definition for the Heidke Skill Score (v2; SWPC 2014), E refers to the Rate Correct for a random forecast model: $\mathrm{E} = \mathrm{CH} + \frac{(\boldsymbol{FP}+\boldsymbol{TN})(\mathbf{FN}+\boldsymbol{TN})}{n^2}$

or products of terms, each of which only depends on blue or red quantities (e.g., Recall, Specificity, False Alarm Rate, True Skill Statistic) do not depend on the underlying class imbalance (Fig. 11).

Bloomfield et al. (2012) and Bobra and Couvidat (2015) offer extensive discussions of the benefits of using the True Skill Statistic over the Heidke Skill Score (e.g., Barnes and Leka 2008) to measure the performance of flare prediction models. The primary reason is that the latter is sensitive to the class imbalance. Choosing metrics that are insensitive to the class imbalance is especially important when data augmentation or sampling strategies are used in the attempt to improve model performance. Since the number of X-class (M-class) flares in a solar cycle is in the dozens (hundreds) in a solar cycle, the underlying population has a high class imbalance. In order to help models train better, ML practitioners may use resampling strategies that mitigate the imbalance. Relying on metrics that are sensitive to the imbalance ratio makes it difficult to compare metrics evaluated on the training set, testing set (which may reflect the population imbalance), and across studies (see discussions by Bobra and Couvidat 2015; Barnes et al. 2016).

7.3.3 Baseline models

When evaluating flare prediction models, whether they are physics-based or purely data-driven, it is important to compare their metrics with respect to baseline models.

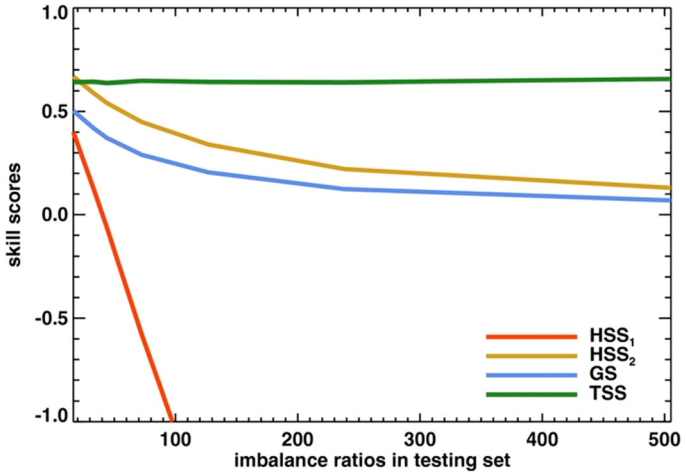

Fig. 11 Dependence of various binary classification metrics (Heidke Skill Scores, Gilbert Skill Score and True Skill Statistic; see Table 2) on the underlying class imbalance ratio $\frac{TN+FP}{TP+FN}$ Image reproduced with permission from Bobra and Couvidat (2015), copyright by AAS

A useful baseline model tends to be one that is interpretable, conceptually simple, and computationally inexpensive (relative to the models being evaluated). The purpose of a baseline model is to serve as a reference point. By evaluating a metric over predictions from the baseline model, and doing the same for a more sophisticated model, one can measure the marginal utility of the extra effort. For flare prediction, two basic baseline models are the random forecast model, and the climatological model.

Random forecast model This model provides a positive forecast with a set probability $p = \frac{TP+FP}{n}$, regardless of the input data. Since the forecast is uncorrelated with the actual event occurrence (and ignores any input features), the Accuracy and Rate Correct of this model are

$$\begin{aligned} \text{Accuracy} &= P(\text{Forecast: Yes}) \times P(\text{Event:Yes}) \\ &= p\frac{(\mathbf{TP}+\mathbf{FN})}{n}. \end{aligned} \tag{53}$$

$$\begin{aligned} \text{Rate Correct} &= (\text{Forecast: Yes}) \times P(\text{Event:Yes}) \\ &+ P(\text{Forecast: No}) \times P(\text{Event: No}) \\ &= p\frac{(\mathbf{TP}+\mathbf{FN})}{n} + (1-p)\frac{(\boldsymbol{TN}+\boldsymbol{FP})}{n}. \end{aligned} \tag{54}$$

The Accuracy and Rate Correct metrics depend on the chosen p, and the underlying event class imbalance.

In contrast, the True Skill Statistic (TSS) for a random forecast model is

$$\begin{aligned} \text{TSS} &= \text{Recall} - \text{False Alarm Rate} \\ &= P(\text{Forecast:Yes}|\text{Event: Yes}) - P(\text{Forecast:Yes}|\text{Event: No}) \\ &= p - p = 0, \end{aligned} \tag{55}$$

which is true irrespective of the class imbalance and the forecast probability (p) chosen.

Climatological forecast model This is a special case of the random forecast model, with $p = P(\text{Forecast: Yes}) = P(\text{Event:Yes})$. Note the forecast probability used here is the event rate evaluated over the population (hence the name climatological). For example, the 0.01% of calendar days has at least one X-flare, and the prediction task is to predict whether at least one X-flare occurred on a calendar day, $P(\text{Event:Yes}) = 0.01$. Suppose $p = 0.01$ for the climatological model. If the testing set used for evaluating metrics has the same class imbalance as the population, $\text{Accuracy} = p^2 = 0.0001$ and $\text{Rate Correct} = p^2 + (1-p)^2 = 0.9802$. So in terms of the former, the climatological model appears dismal, and in terms of the latter, it performs spectacularly well. In contrast, $\text{TSS} = 0$, which illustrates why this is an unbiased metric.

The Gilbert Skill Score and the Heidke Skill Score (v2; see Table 2) are both metrics that are defined relative to the random forecast model. They partially address the desire that we want to measure the marginal utility of a model against a baseline model. Nevertheless, they still suffer from dependence on class imbalance.

Our recommendation is to decouple metrics from models (as opposed to the Gilbert and Heidke Skill Scores). If possible (and desirable, depending on the stakeholder's needs), choose unbiased metrics like TSS. Then evaluate metrics for baseline and ML models alike to evaluate marginal utility (improved performance, if any). We caveat this recommendation by reiterating that the most relevant metric(s) always depends on the context and the stakeholder(s).

The choice of appropriate baseline models depends on the application. For flare prediction, the climatological model is used as a reference point by NOAA (e.g., see Barnes et al. 2016). In some contexts where predictive models (where purely ML-baed and/or physics-based) are already in common use, the State-of-the-Art (SOTA) model may be appropriate.

7.3.4 Weakly-labeled supervised training

The analysis and prediction of flares, especially when done with spectra, is an instance of weakly-labeled datasets. The observations of the IRIS satellite are of special relevance for the analysis of flares in recent times. Although each spectral observation constitutes a fundamental unit of information, the label associated with the flare (flare/no flare) cannot be put at the level of individual spectra but only at the time series level. Huwyler and Melchior (2022) approached this classification problem by using multiple instance learning (Dietterich et al. 1997), a supervised learning technique that associates labels not to individual instances but to bags of instances. They were able to detect the presence of flaring regions with tens of

minutes in advance from observation of the Mg II window with IRIS. These weakly-labeled techniques can also be of great help in segmentation problems.

7.3.5 Operational flare forecasting models

Leka et al. (2019a) provides a comprehensive review of operational flare forecasting models and, for the first time, a consistent comparison between flare models deployed at various international agencies and research institutions. While most models perform better than a no-skill baseline model, there was no single operational model that consistently outperformed others over a broad set of metrics and event distributions.

Further detailed analysis of the behaviors of operational flare models by Leka et al. (2019b) and Park et al. (2020) provides some important conclusions. Firstly, information regarding prior flare activity and active region evolution can improve forecasts. Secondly, having a human "forecaster in the loop" helps. Thirdly, performance degrades when data is restricted to near disk-center. Lastly, the use of "modern data sources" (e.g., SDO/HMI) and statistical approaches improves performance. The data used for the comparison is available from Leka et al. (2019a).

7.3.6 Deep learning for flare prediction

The widespread availability of GPUs, deep learning frameworks and open-source computer vision codebases has supercharged the adoption of computer vision methods for flare forecasting. Huang et al. (2018) applied CNNs to MDI and HMI line-of-sight magnetograms for flare forecasting. They find that the trained CNNs include intermediate spatial filters that are sensitive to magnetic polarity inversion lines. In contrast, some deep neural network flare prediction models use "hand-engineered" feature extraction (e.g., Nishizuka et al. 2018). LSTMs have also been applied to flare prediction using 25 SHARP parameters, augmented by 15 flare history parameters (Liu et al. 2019). Consistent with prior literature, this work shows the incorporation of the prior flare productivity improves prediction performance.

Given the success of deep neural models, Yi et al. (2023) proposed to train the CNN model proposed by Yi et al. (2021) using deep reinforcement learning. This model predicts the presence of a flare as a binary output from line-of-sight magnetograms. The results indicate that RL can improve the quality of the prediction when compared to more standard training schemes, especially when dealing with rare events.

Given that these deep neural models will eventually be part of operational flare forecasting strategies, it turns out important to check their biases. Liu et al. (2022) analyzed several deep neural models to look for the influence of the image resolution on the prediction abilities. They found that the models analyzed are robust to the specific image resolution. They pay more attention to global features extracted from the active regions, and pay less attention to local information in magnetograms. This points out that these models will become operational soon.

7.4 Explainable models for flare prediction

Many of the deep learning models developed for flare prediction are complex. Consequently, it is difficult to interrogate the models to understand the reasons why a model predicts the presence of a flare. For this reason, the community has recently relied on some of the techniques for explainability developed in machine learning in recent times (Barredo Arrieta et al. 2020). Yi et al. (2021) used Gradient-weighted Class Activation Mapping[32] (Grad-CAM; Selvaraju et al. 2017) to localize the regions of the solar surface that triggered the model to predict a flare. Grad-CAM can be used with any CNN-based model. It works by computing the derivative of the prediction with respect to the final convolutional layer to produce a coarse importance map. This map highlights the parts of the input images that trigger the detection of a flare in the neural network. They find that the model correctly focuses on the polarity inversion line to forecast a flare, a fact that is well known. Likewise, Panos et al. (2023) used Grad-CAM and the game-theoretic method expected gradients (Erion et al. 2021) to discover features in the spectra of Mg II for predictions of flares. They found that triplet emission, flows, broadening, and highly asymmetric spectra are features that appear before a flare. Additionally, the regions to which the neural networks pay more attention for the prediction are strongly associated with the location of the maximum UV emission of the flare.

7.5 Heliosphere and space weather

This section focuses on applications of ML to heliospheric and space weather problems using solar data as inputs. For an overview of the role of ML in space weather studies and forecasting and a gentle introduction to ML tailored for the space weather audience, we refer the reader to Camporeale (2019). Another recommended review paper is Bortnik and Camporeale (2021), which lists ten broad categories of approaches to applying ML to space science problems. We show some examples of applications, although we encourage the reader to consult these review papers focused on many aspects of heliophysics beyond solar physics.

Torres et al. (2022) used fully connected neural networks to predict the solar energetic particles (SEP) above 10 $\mathrm{cm}^{-2}s^{-1}sr^{-1}$ with energies above 10 MeV to occur from the properties of the coronal mass ejection (CME). They found that the neural approach provides consistent results, although they depend on the availability of observations of the CME. This makes this method not sufficiently reliable, since SEPs can be present even if no CME is seen. A similar approach was pursued by Lavasa et al. (2021), although they compared the neural approach with many different linear classifiers.

Upendran et al. (2020) tackled the problem of solar wind speed prediction at Lagrangian point 1 (L1) by training a DNN which takes temporal sequences of SDO/AIA EUV images to predict the solar wind velocity as available in the OMNI database. This work made use of the technique of transfer learning (TL), whereby the frontend of the DNN was imposed as a set of pretrained layers from a well-known

[32] https://github.com/jacobgil/pytorch-grad-cam.

computer vision package (in this case, GoogleNet). This frontend acts as a preprocessor for feature extraction. The output latent vector is then passed on to the remaining trainable layers of the DNN. The SDO dataset used by Upendran et al. (2020) comprised AIA images at a daily sampling frequency. By using instead a 30 min sample frequency, Brown et al. (2022) reported significant improvements in evaluation metrics (e.g., root mean squared error) for the solar wind speed prediction. Another change they made was to use an attention-based mechanism, though the improvement of model performance is largely attributed to the much higher data sampling frequency.

Bernoux et al. (2022) trained a DNN to use SDO AIA images (193 Å) as inputs to produce a probabilistic forecast of geomagnetic activity (Kp index). The model output is probabilistic in the sense that the output consists of a mean and standard deviation of Kp. Similar to the aforementioned work on solar wind speed prediction, this model uses TL and has a preprocessor feature extractor.

7.6 Solar Cycle predictions

It is well known that the Sun passes from a low magnetic activity (measured as the number of visible sunspots on the disk) to a high magnetic activity with a periodicity of roughly 11 years. Over time, researchers have tried to find correlations between the solar cycle and other observables. Among them, we find flares, CMEs, geoeffectiveness (Ap index measured at Earth), cosmic ray flux reaching the Earth's environment, and many more. The observations showed that the activity of the Sun is correlated with the amount of cosmic rays reaching the Earth (Usoskin 2023). The sunspot number (SSN) displays a high correlation with the total solar irradiance (TSI),[33] which turns out to be an important parameter for understanding of the Earth's climate. It also displays correlation with the occurrence of CMEs (Lamy et al. 2019). The periods of large activity in the Sun produce strong magnetic fields in the atmosphere, which are correlated with strong eruptive events. They can be hazardous for the Earth's environment.

In quest of a suitable solar cycle forecast method, Nandy (2021) analyzed 77 predictions made by different research groups for cycle 24 and 37 predictions for the current cycle 25. Out of the 77 models, only a couple of models managed to properly predict the observed peak of the cycle. Interestingly, none of the methods based on machine learning models was able to correctly predict the amplitude of the cycle.

Cycle 25 is not yet at its peak and the aim of some of the most recent methods based on ML is the prediction of its maximum amplitude and when it will take place. Li et al. (2021) reported two methods employing an auto-regressive neural network[34] method and a recurrent LSTM network. Using the same LSTM approach, and based on the SSN variation, three other predictions were reported. Prasad et al. (2022) predicted an increase of $\sim$20% with respect to the previous cycle, with the peak in August 2023 and a maximum of SSN of 171.9 ± 3.4. Wang et al. (2021b) reported a

[33] TSI is defined as the radiant energy emitted by the Sun at all wavelengths crossing a square meter each second outside Earth's atmosphere (Hathaway 2015)

[34] Autoregression implies predicting the future of a sequence using previously observed values.

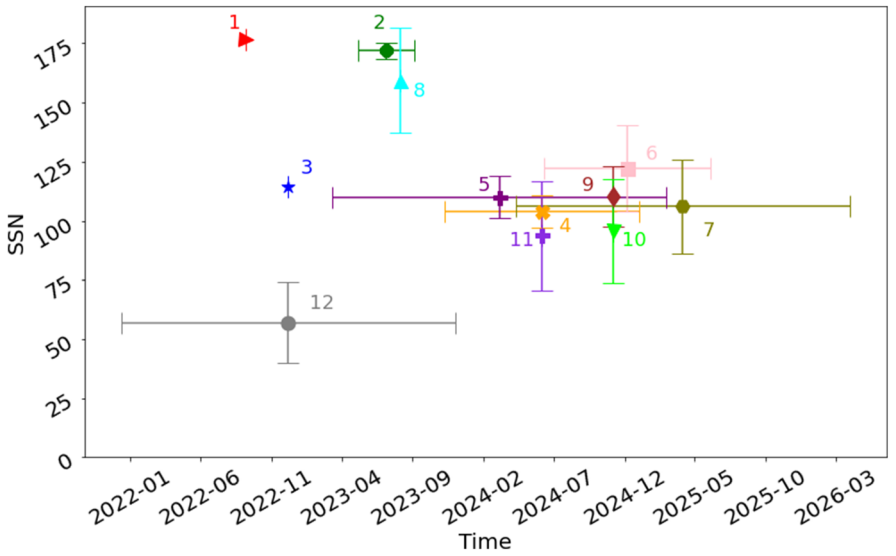

Fig. 12 Predictions of the solar cycle 25 peak time and SSN for different ML models: mark 1 with red right-triangle (Li et al. 2021), mark 2 with green octagon (Prasad et al. 2022), mark 3 with blue star (Wang et al. 2021b), mark 4 with orange x and mark 5 with purple plus (Bizzarri et al. 2022), mark 6 with pink square (Okoh et al. 2018), mark 7 with olive hexagon (Benson et al. 2020), mark 8 with cyan triangle, mark 9 with brown diamond, mark 10 with lime triangle, mark 11 with blue-violet plus (Dani and Sulistiani 2019), and mark 12 with grey circle (Covas et al. 2019)

decrease in the amplitude of cycle 25, with a predicted peak SSN of approximately 114 around 2023. Finally, Bizzarri et al. (2022) forecasted a decrease of the peak amplitude of $\sim 14\%$ with respect to cycle 24, and a maximum activity peak on cycle 25 around mid-2024.

Okoh et al. (2018) used a hybrid of regression and a neural network to provide a prediction. The regression method is used to derive characteristics of the solar cycle, which was used afterward as input for a neural network. They predicted a maximum amplitude for cycle 25 of 112.1 ± 17.2, to happen in January 2025 (±6 months).

Benson et al. (2020) predicted a weaker cycle 25 when compared with cycle 24, with a maximum SSN of 106 ± 19.75. Their estimations are based on the WaveNet (Oord et al. 2016) and LSTM architectures. WaveNet is a DNN based on an autoregressive generative model. It learns to model the probability distribution of a given time-series conditioned on the past. To this end, it uses dilated causal convolutional layers (see Oord et al. 2016, for more details), which allows the model to capture time dependencies of very long baselines. They predicted a peak SSN of 106 ± 19.75 for cycle 25. Using four machine learning techniques, Dani and Sulistiani (2019) obtained four different predictions for the strength of cycle 25. Based on a feed-forward artificial neural network implementation, Covas et al. (2019) predicted the lowest amplitude for cycle 25. A linear regression predicts the maximum to occur in September 2023 (with an amplitude of 159.4 ± 22.3). A random forest (RF) and a radial basis function (RBF) method predicts the same time for peak, happening in December 2024 but with two different amplitudes: 110.2 ±

12.8 for the RF and 95.5 ± 21.9 for the RBF. Finally, a SVM method predicts a peak around July 2024 with a peak SSN of 93.7 ± 23.2. All predictions, showing the dispersion, are summarized in Fig. 12.

All the proposed methods show very good testing and prediction capabilities for the past solar cycles. However, it is uncertain whether this is true in future cycles. Predicting a nonlinear process like the solar cycle is a delicate task, and it remains to be tested that the statistical properties of previous solar cycles contain enough information to predict the future.

7.7 Inversion of Stokes profiles

7.7.1 Accelerating inversions

The application of neural networks for the inversion of Stokes profiles goes back in time to Carroll and Staude (2001), who proved that multi-layer FCN could be used for estimating model parameters from the observations. Carroll and Staude (2001) proposed their use for simple Milne-Eddington inversions and concluded that they were able to obtain physical parameters without any optimization once the neural networks were trained. As additional advantages, they showed that the neural networks provided an increase in speed, noise tolerance, and stability. This was later verified by other works (Socas-Navarro 2003, 2005b). Inspired by these advances, Asensio Ramos and Socas-Navarro (2005) also applied neural networks for the acceleration of the solution of chemical equilibrium. Solving chemical equilibrium with a large set of species turns out to be slow and can dominate the computation time in inversion codes. That is precisely the reason why the inversion code NICOLE (Socas-Navarro et al. 2015) has the neural solution as an option.

Carroll and Kopf (2008) later expanded their original work to use FCNs to infer the depth stratification in a geometrical height scale of the temperature, velocity, and magnetic field vector. The network was trained using stratifications and synthetic Stokes profiles from an MHD simulation of the quiet Sun (Vögler et al. 2005). The application of the neural network in a pixel-by-pixel manner allowed them to recover a tomographic view of the FOV by recombining all individual line-of-sight stratifications.

After an impasse of more than a decade, the neural inversion of Stokes profiles is again gaining momentum, driven by modern DNN. Asensio Ramos and Díaz Baso (2019) proposed SICON,[35] a CNN that is trained with MHD simulations and opens up the possibility of carrying out extremely fast inversions of 2D maps for observations of the Hinode satellite. As an example, a map of 512×512 pixels can be inverted in an off-the-shelf GPU in merely 200 ms. The authors proposed two different architectures, both of them displaying consistent results. Apart from the enormous speed of the inversion, the CNN's have other advantages. One of them is that the inferred physical properties are not affected by the Hinode PSF, so it essentially deconvolves the data while inverting. This can only be achieved for space-born observatories because the PSF is well known and constant with time.

[35] https://github.com/aasensio/sicon.

Another advantage is that the networks can provide estimations of quantities that are very difficult to obtain with classical inversion methods. This is the case of gas pressure and the Wilson depression. The main reason why these CNNs can do this job is because they exploit correlations in the training data.

Higgins et al. (2021) also used a CNN (a U-Net in this case) to accelerate the production of vector magnetograms from HMI/SDO. Instead of training with simulations, they trained the CNN with inversions carried out with the standard pipeline. They also viewed the inference as a classification problem with a large number of bins for each variable of interest, instead of a regression problem. Despite the inherent binning error, this allowed them to easily compute uncertainties in the output. Higgins et al. (2022) expanded their previous work by training the U-Net with inversions of the same field of view and at the same time carried out with the Hinode/SOT-SP instrument. They developed SynthIA (Synthetic Inversion Approximation), which works under the assumption that the information encoded in the Hinode/SOT-SP observations (and the ensuing inversions) is also present in HMI/SDO (potentially spread over multiplet pixels). This assumption is non-trivial since HMI/SDO observes only the Fe I spectral line at 617.3 nm at low spectral resolution, while Hinode/SOT-SP observes the pair of lines at 630 nm at high spectral resolution. They showed that SynthIA can indeed extract this information and produce full-disk inversions with a quality similar to that of Hinode/SOT-SP. A similar approach has been pursued by Jiang et al. (2022) to generate vector magnetograms for the Michelson Doppler Imager (MDI) onboard the Solar and Heliospheric Observatory (SOHO). MDI/SOHO was observing the Sun between 1996 and 2010, but it was only recording the longitudinal component of the magnetic field. Jiang et al. (2022) combined this information with Hα observations collected with the Big Bear Solar Observatory (BBSO) to train a CNN to produce maps of the components of the magnetic field in the plane of the sky. The trained CNN produces good vector magnetograms, extending the period in which vector magnetograms are available for the Sun from 1996 to the present day. Despite the success of these approaches, we caution that using data from different instruments should be done with care since small data alignment problems might affect the results (Fouhey et al. 2022).

Milic and Gafeira (2020) showed that a relatively simple 1D CNN can output temperatures, velocities, and magnetic fields at three optical depth heights in the atmosphere directly from the Stokes profiles. They train the CNN with the aid of MHD simulations of the quiet Sun. However, in order to more closely mimic standard inversion codes, they do not train directly with the data from the simulation. They first invert the data with SNAPI (Milić and van Noort 2018) and use these results as a training set. The output of the network shows a very good correlation with the original data while accelerating the inversion by a factor $\sim 10^5$.

Along a different line, we find studies of applying DNNs to provide initial solutions to standard gradient-based inversion codes (Gafeira et al. 2021). These methods can greatly accelerate the convergence of inversion codes because the initialization is close to the expected solution. One of the sub-products of starting close to the solution is that the Levenberg-Marquardt algorithm often used in these

inversion codes can be made to work close to the Gauss-Newton regime from the very beginning, which has an almost quadratic convergence rate.

The inversion of lines affected by departures from local thermodynamic equilibrium (non-LTE) is computationally demanding. The reason is that one needs to self-consistently solve the statistical equilibrium equations for the atomic/molecular species producing the observed spectral lines and the radiative transfer equation (see the review by de la Cruz Rodríguez and van Noort 2017). Accelerating this process has been recently tackled with two different approaches. The first one uses CNNs (Chappell and Pereira 2022) to map the populations in LTE to the populations in non-LTE (the ratio between the populations in non-LTE and those in LTE is known as the departure coefficients) for a hydrogen model atom. Since populations in LTE can be obtained from the local physical properties, the trained mapping avoids the solution of the time consuming radiative transfer problem. Another approach has been recently presented by Vicente Arévalo et al. (2022) based on graph neural networks and specifically tailored to accelerate inversions of chromospheric lines of Ca II. This approach predicts the departure coefficients as a function of the height in the atmosphere, producing a speedup of a factor 10^3 without a significant impact in the synthetic spectral lines. This allows inversions of chromospheric lines, even those dominated by partial redistribution effects like Ca II H & K, to be carried out as fast as lines formed in LTE.

7.7.2 Uncertainty characterization

Since inversion problems are ill-defined in general, providing a single point estimate of the physical parameters as output is not optimal. In principle, one should provide full posterior distributions, which encode the uncertainties and correlations among all model parameters (Asensio Ramos et al. 2007b). A deep learning approach to this has been pursued by Osborne et al. (2019) based on the concept of invertible neural networks (INNs; Ardizzone et al. 2018). The idea of INNs is to learn the forward and inverse mappings simultaneously. The forward mapping, $y = f(x)$ goes from model parameters x to observations. The inverse mapping, $x = g(y, z)$ is augmented with a latent vector z that is assumed to be extracted from a known distribution. This latent vector takes into account all information lost during the forward pass, which precisely makes the inverse problem ill-defined. Once the INN is trained, an approximation to the posterior distribution can be obtained by sampling the latent vector. Osborne et al. (2019) were able to derive temperatures, electron number densities and velocities in flaring regions from the interpretation of the Hα and Ca II 8542 Å line using a RADYN model (Carlsson and Stein 1992, 1995, 1997; Allred et al. 2015).

Normalizing flows can also be utilized to characterize uncertainties. If the NF is conditioned on the observations, the normalizing flow can be trained to return Bayesian posterior probability estimates of the model parameters for any arbitrary observation. This amortized posterior estimation is time consuming to train but can then be applied very fast to observations, opening up the possibility of doing Bayesian inference in large fields of view. Díaz Baso et al. (2022) showed how this

can be applied to the inversion of Fe I and Ca II data and diagnose the stratification of the solar photosphere and chromosphere. They obtained the most probable value of the temperature, bulk velocity, and microturbulent velocity in a very large field of view, together with their uncertainties and correlations.

7.8 3D reconstruction of the solar corona

Rahman et al. (2023) has recently shown that one can use GANs to build a mapping from photospheric magnetograms to electron density maps at different heights in the atmosphere. The model is trained with simulations from the Magnetohydrodynamic Algorithm outside a Sphere (MAS) method, that solves the time-dependent resistive magnetohydrodynamic equations (MHD) in 3D, including coronal heating, thermal conduction and radiative losses (Lionello et al. 2008; Riley et al. 2015).

Recent works have demonstrated the potential of using fully connected neural networks for the description of continuous fields (scalar, vector,...) as a function of the position in space (e.g., Mildenhall et al. 2020). To this end, neural networks, usually termed implicit neural representations (INR), coordinate-based representations (CBR), or neural fields (NeF), are used to map coordinates on the space (or space-time) to coordinate-dependent field quantities. NeFs have many desirable properties. They are very efficient in terms of the number of free parameters. They produce continuous and differentiable fields, which can then be seamlessly part of complex models. Finally, they have a strong implicit bias, favoring specific signals. An NeF is given by the following simple, but flexible, fully-connected neural network:

$$\begin{aligned} \log N_e(\mathbf{x}) &= \phi_n \circ \phi_{n-1} \cdots \circ \phi_0(\mathbf{x}), \\ \phi_i &= \sigma(\mathbf{W}_i \mathbf{x}_i + \mathbf{b}_i), \end{aligned} \tag{56}$$

where $\mathbf{W}_i$ are weight matrices, $\mathbf{b}_i$ are bias terms, and σ is an activation function.

NeFs, as defined by Eq. (56), are known to suffer from the so-called spectral bias (Rahaman et al. 2019; Wang et al. 2021c), which prevents them from learning high-frequency functions. This problem has been empirically alleviated by first passing the input coordinates through a Fourier feature mapping, which allows the INR to correctly generate high spatial frequencies (Tancik et al. 2020). Recently, Sitzmann et al. (2020) proposed `SIRENs`,[36] which uses periodic functions (sines) as activation functions so that the electron density can be written as:

$$\begin{aligned} \log N_e(\mathbf{x}) = S_\psi(\mathbf{x}) &= \mathbf{W}_n(\phi_{n-1} \circ \phi_{n-2} \cdots \circ \phi_0) + \mathbf{b}_n \\ \phi_i &= \sin(\omega_i(\mathbf{W}_i \mathbf{x}_i + \mathbf{b}_i)), \end{aligned} \tag{57}$$

where ψ summarize all tunable parameters of the `SIREN`. Thanks to a specific initialization procedure, a `SIREN` can efficiently reproduce both low and high spatial frequencies. In some sense, a `SIREN` can be seen as a nonlinear extension of a Fourier series.

[36] https://github.com/vsitzmann/siren.

NeFs have been recently introduced in solar physics by Jarolim et al. (2022) for the description of the magnetic field in the solar corona. Jarolim et al. (2022) carry out the extrapolation of the photospheric magnetic field by describing it with a NeF. They optimize the neural network by imposing the force-free and the solenoidal conditions:

$$\begin{aligned} L_{\mathrm{ff}} &= \frac{\|(\nabla \times \mathbf{B}) \times \mathbf{B}\|^2}{\|\mathbf{B}\|^2 + \epsilon} \\ L_{\mathrm{div}} &= \|\nabla \mathbf{B}\|^2. \end{aligned} \tag{58}$$

The spatial derivatives are easy to compute for a NeF using automatic differentiation. These losses are optimized by simultaneously fulfilling the boundary conditions.

Later, Bintsi et al. (2022) used NeFs to show that it is possible to infer the emission properties in the whole 3D corona from a set of observations from the ecliptic (with latitudes below 7°). To this end, NeFs are used to describe the local emission properties in the 3D volume. The training requires accumulating the emission along rays using ray tracing and optimizing a loss that compares the synthetic images and AIA/SDO observations at 193 Å. The simulations demonstrate that 32 observations are enough to obtain a very accurate description of the corona, even in the polar regions. Extending the procedure to infer physical properties like temperature and electron density (from which the local emission properties are computed) is just one step ahead.

The force-free extrapolation of magnetic fields in the corona, especially above photospheric magnetic field concentrations, improves significantly if one has additional information about the 3D geometrical structure of coronal loops observed in the UV. A reliable inference of this 3D structure requires the use of triangulation techniques with the aid of stereoscopic observations. Since having these observations is not the case in many occasions, Chifu and Gafeira (2021) used a CNN to extract the Z component of the loop based only on the 2D shape extracted from a EUV single image. The model obtained a very high accuracy for short loops with no complex shapes and lower performance in very complex and twisted shapes.

7.9 Image deconvolution

Observing any other astronomical object through the Earth's atmosphere introduces perturbations that are difficult to correct. The obvious solution of moving to space is not always possible or feasible. Even if adaptive optics systems are working properly, some residual wavefront perturbations are still present in the images, and the diffraction limit of the telescope is not reached. A posteriori correction techniques based on phase diversity (Paxman et al. 1992; Löfdahl and Scharmer 1994; Löfdahl et al. 1998) and multi-object multi-frame blind deconvolution (MOMFBD; Löfdahl et al. 2002; van Noort et al. 2005) have been developed. The main disadvantage of these methods is their large computational requirements. For this reason, deep learning has been applied recently by Asensio Ramos et al. (2018) to accelerate the deconvolution process. The method is based on a fully convolutional deep neural network that was trained supervisedly with images previously corrected with the help

of MOMFBD. Once trained, this method can deconvolve bursts of 1k×1k images containing 7 short-exposure images in $\sim$ 5 ms with an appropriate GPU. This opens up the possibility of, for instance, doing the deconvolution online while analyzing the data.[37]

Although a step forward in terms of speed, the neural approach developed by Asensio Ramos et al. (2018) has two main problems. The first one is that it is trained with supervision, so one needs to use the MOMFBD algorithm to build the training set. Though not a major obstacle, a method that does not need this previous step would be preferable. The second issue is that it only produces deconvolved images. No estimation of the wavefront in each individual frame is produced. Estimating the wavefronts can be helpful in checking the performance of the telescope and instrument and understanding the performance of the adaptive optics. For this reason, Asensio Ramos and Olspert (2021) improved the approach by showing how the training can be done in a fully unsupervised manner, while also producing an estimation of the wavefront for each observed frame. Given the lack of supervision, the method can be generally applied to any type of object, once a sufficient amount of training data is available.[38]

7.10 Image-to-image models

Arguably the most powerful property of deep learning models is their ability to deal with high-dimensionality data like images. Models are powerful enough to produce very high-resolution natural images. This fact has also been exploited in solar physics in different applications that we summarize in the following.

7.10.1 Synthetic generation of solar data

The multi-wavelength and multi-layer coverage of the solar atmosphere by SDO instruments provide opportunities to explore the synthetic generation of solar data. In this context, synthetic data generation includes the translation of data from one instrument into proxy data for another instrument (or even the same instrument), as well as the generation of data that follow the underlying distribution of real observed data, but which is not necessarily instantiated on the real Sun.

One example of data translation is the generation of proxy EUV spectral irradiance data. Using the SDOML dataset (Galvez et al. 2019b), Szenicer et al. (2019) trained a CNN to translate multi-channel AIA images into disk-integrated line (and band) irradiance data observed by EVE. This was done by using data captured by both instruments before mid 2014, when EVE MEGS-A was still operational. The errors from the CNN model prediction are smaller than from a physics model based on differential emission measure (DEM) inversions. After the model was trained, a rearrangement of the network components allowed for the generation of synthetic line irradiance images (Fig. 13).

[37] https://github.com/aasensio/learned_mfbd.

[38] https://github.com/aasensio/unsupervisedMFBD.

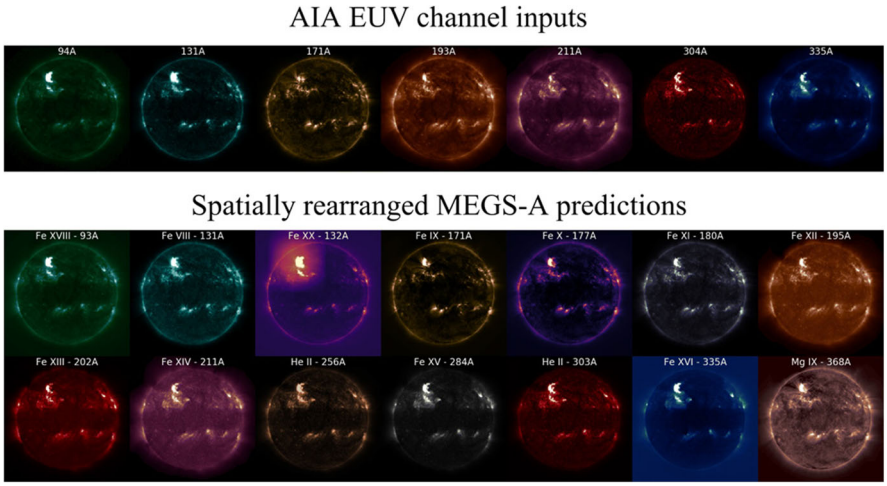

Fig. 13 Synthetic EUV line irradiance images generated by a CNN trained to map AIA images to EVE disk-integrated line irradiance data Image reproduced with permission from Szenicer et al. (2019), copyright by the author(s)

Conditional Generative Adversarial Networks (cGANs) have been trained to generate synthetic magnetograms from EUV/UV images (Kim et al. 2019). Theoretical considerations would suggest EUV/UV intensity data would not encode magnetic global polarity information (Liu et al. 2021). Specifically, the expectation is that the thermodynamic structure of the solar atmosphere is symmetric under the operation $\mathbf{B} \rightarrow -\mathbf{B}$. However, the synthetic magnetograms from Kim et al. (2019) do have bipolar active regions that resemble real active regions with Hale polarity rules consistent with solar cycle 24. Detailed comparisons with actual observed active regions reveal big differences in the morphology. So these synthetically generated magnetograms would not be useful for AR-scale studies. Whether they are suitable for use for downstream heliospheric predictions remains to be seen.

Another image translation problem is the synthetic generation of EUV images from other EUV images of different wavelengths. This problem was posed by Salvatelli et al. (2019) in the context of potentially reducing the number of physical channels needed in future EUV telescopes. The approach was to use three AIA input channels to generate another AIA channel using U-Nets. This problem was further extensively explored by Lim et al. (2021) using cGANs, who considered image translation from single, double, and triple input channels with cross-correlation (CC) coefficient between prediction and ground truth as the performance metric. Salvatelli et al. (2022) further explored the problem by considering other metrics, including commonly used computer vision metrics like structural similarity index measures. Salvatelli et al. (2022) showed that the CC metric may not be the ideal performance metric, and also showed how various metrics degraded when the trained model was applied during flaring conditions.

7.10.2 Estimation of velocities

Motions in the solar photosphere are fundamentally controlled by convection in a magnetized plasma. Remotely sensing these three-dimensional velocities is important for the analysis of solar events. The component along the line of sight (LOS) of the velocity can be extracted from spectroscopic observations thanks to the Doppler effect. However, the components of the velocity field in the plane perpendicular to the LOS cannot be diagnosed spectroscopically. Different algorithms have been used to trace horizontal flows at the solar surface by estimating the optical flow from consecutive images. The most widespread is the method of local correlation tracking (LCT; November and Simon 1988). This method suffers from problems when dealing with events of a short time duration or with reduced physical size. To alleviate this, Asensio Ramos et al. (2017) developed DeepVel,[39] an end-to-end deep learning approach for the estimation of horizontal velocity fields in the solar atmosphere based on a deep fully convolutional neural network. The neural network was trained on a set of velocity fields obtained from simulations of the quiet Sun. DeepVel is very fast, uses only two consecutive frames, and returns the velocity field in every pixel and for every time step. DeepVel opened up the possibility of identifying small-scale vortices in the solar atmosphere that last for a few minutes and with sizes of the order of a few hundred kilometers, something impossible with methods based on local correlation tracking

DeepVel was later retrained by Tremblay et al. (2018) to carry out an exhaustive comparison with classical local correlation methods and check their ability in extracting transverse plasma motions at a large scale from SDO/HMI observations. They concluded that DeepVel was able to beat classical methods by a large margin in small scales, those of the granulation while being very similar to local correlation methods in larger scales. It is encouraging that, when applied to simulations, DeepVel is able to nicely recover the kinetic energy density from the simulation.

A new model, DeepVelU,[40] based on the U-Net architecture, has been proposed by Tremblay and Attie (2020). This model displays several improvements with respect to the original DeepVel network. The U-Nets analyze the inputs in a multiscale fashion, which turns out to be interesting to capture horizontal velocities at different scales, from granular to the supergranular scales. Additionally, Tremblay and Attie (2020) trained DeepVelU in simulations of the quiet Sun and active regions. They checked that DeepVelU is able to capture the transverse velocities from simulations with much improved correlation, especially when dealing with large spatial scales.

7.10.3 Superresolution

Instruments are limited by optics to provide a certain spatial resolution on the solar surface. However, the recent field of research on compressed sensing, which is founded on the idea of sparsity and compressibility, has demonstrated that one can

[39] https://github.com/aasensio/deepvel.

[40] https://github.com/tremblaybenoit/DeepVel_DeepVelU.

enhance the spatial resolution of images under certain conditions. It is clear that the presence of spatial correlation in the images of the Sun suggests that one can enhance current observations to provide a certain degree of superresolution. Díaz Baso and Asensio Ramos (2018) proposed `Enhance`,[41] a deep CNN that provides superresolved continuum and magnetograms for SDO/HMI. The nominal pixel size of HMI of 0.5″ is transformed into 0.25″. These images are compared, as a cross-check, with images obtained from the Hinode satellite (correctly degraded to provide a resolution of 0.25″ per pixel). The superresolved images provide a very good representation of the small scales, enhancing the contrast in the continuum in the quiet Sun by almost a factor 2. Magnetograms are also properly superresolved although the fact that this is a signed quantity can produce small artifacts. All-in-all, `Enhance` is a very good tool to provide a better picture of the environment around regions of interest.

More recently, Dou et al. (2022)[42] have used generative adversarial network (GAN) to produce high-fidelity and photorealistic super-resolved images of Michelson Doppler Imager (MDI) in order to match the Helioseismic and Magnetic Imager (HMI) resolution. First, a GAN model is designed to downscale the HMI data to MDI resolution to transfer the characterization of the HMI data to the MDI scale. Then a second supervised GAN model was developed to produce the superresolved magnetograms based on the MDI data. We caution the reader to be very critical when using superresolved data for data analysis since the presence of artifacts and ambiguities can surely affect the physics inferred from them.

7.10.4 Denoising

Although exquisite detail is put on the design of the instruments developed to observe the Sun, they are always affected by noise. A posteriori methods can be used to denoise the data by exploting the regularity of the solar structures, helped by the fact that noise has reduced spatial and temporal correlation. As already discussed in Sect. 3.1.1, linear methods have been used with this purpose. However, new, more powerful nonlinear methods are appearing in the literature, and they are being used for denoising different solar observations. In particular, Díaz Baso et al. (2019) got inspiration from the Noise2Noise approach of Lehtinen et al. (2018). This is a method that supervisedly trains a relatively simple denoising neural network by only having pairs of the same solar structure with two different realizations of the noise. In contrast, the standard supervised approach needs pairs of noisy and clean images, which are only possible using synthetic data. It is obvious that obtaining training examples for the Noise2Noise approach is much easier than for the standard supervised case. This was indeed demonstrated by Díaz Baso et al. (2019), who used pairs of images taken with the CRISP instrument mounted on the Solar Swedish Telescope (SST) at the same wavelength but at different times, making sure that the time separation was small. The denoising results are great, with special relevance on filterpolarimeters like CRISP, which show conspicuous (preferentially when

[41] https://github.com/cdiazbas/enhance.

[42] https://github.com/dfpdl/SPSR.

analyzing polarimetric signals) systematic artifacts on the observed field, produced either by the instrument or by the data reduction process.

Later, Park et al. (2020) also approached the denoising problem of SDO/HMI solar magnetograms using a deep convolutional conditional GAN, leading to a reduction in the average noise level of more than a factor 2.5. The GAN is trained so that it maps single magnetograms to the average of the 21 magnetograms centered on the one of interest (including 10 before and 10 after). Potentially, this could lead to a reduction in the noise standard deviation of a factor ~ 4.6, although it can also lead to a slight blurriness of the generated images produced by motions in the solar surface. The generator network is conditioned on the noisy magnetogram and its purpose is to produce a denoised version of the magnetogram. Following the standard GAN paradigm, a second discriminator network is in charge of distinguishing the magnetogram produced by the generator and the real ones from the training set. The equilibrium of the two networks is produced when the generator produces images indistinguishable from the training set so that the discriminator is fooled roughly 50% of the time.

7.10.5 Image desaturation

A very interesting application of deep learning is the desaturation of SDO/AIA data. These synoptic observations frequently suffer from saturation effects mainly as a consequence of the occurrence of solar flares. Correcting the saturated regions of the image is an instance of image inpainting. The aim is to fill the (irregular) holes by leveraging statistical information from the rest of the image and from the training set. To this end, Yu et al. (2022) developed a model using a GAN. The generator is based on a U-Net that uses partial convolution layers (Liu et al. 2022) instead of standard convolutional layers. These partial convolution layers are specifically suited for inpainting tasks. The discriminator is based on a PatchGAN architecture (Isola et al. 2017; Wang et al. 2021a). The results show a promising avenue to provide continuous synoptic observations even when energetic events happen in the Sun.

7.10.6 Farside imaging

Predictions of the active regions currently on the hidden side of the Sun (known as farside) are routinely computed using helioseismic measurements. They are obtained by solving the inversion problem known as helioseismic holography (Lindsey and Braun 1997), which uses time series of waves on the visible surface (nearside) and map them back to the far side. Given the dispersive character of the mapping between the nearside and the farside, the resulting images are quite diffuse. Machine learning has great potential for the improvement of these inversions. Kim et al. (2019) trained generative models to produce farside magnetograms from STEREO extreme ultraviolet (EUV) images. Since the polarity of the magnetic field is not directly encoded on the EUV images, it is noteworthy that the correct polarity can be recovered. Felipe and Asensio Ramos (2019) gave the more conservative step of proposing a CNN (FarNet) that associates the farside maps obtained with helioseismic holography with probability maps obtained from magnetograms

acquired half a rotation later. As a consequence, the aim is to estimate the presence of active regions, neglecting the polarity, with nearside data. The neural approach is able to detect much weaker active regions than those that are detected with the standard technique. Improvements on this approach will probably require deep architectures directly trained with Doppler maps. Later, Broock et al. (2021) analyzed the statistical properties of FarNet and concluded that for equivalent false positive ratios when compared with the standard method, it produces $\sim 47\%$ more true detections. Additionally, it is able to detect much weaker active regions. A significant improvement (FarNet-II) was also recently published by Broock et al. (2022), by including attention mechanisms and convolutional recurrent layers based on the ConvLSTM approach (Shi et al. 2015). Using temporal information provides a much improved time consistency of the predicted active regions, also allowing for a better prediction in the case of weak active regions.[43]

8 Outlook for the future

Machine learning has been routinely used in solar physics. However, the recent deep learning revolution is producing a panoply of new applications that were never envisioned a few years back, permeating in many subfields of research inside solar physics. The availability of increasingly larger observational material is making solar physics transition to the powerful collection of methods that advanced ML offers to help us understand what we see. We frankly think ML will become an intrinsic part of our research in the future.

Currently, many applications in solar physics consider the ML model as a very convenient way of parameterizing a very flexible mapping that carries out the inverse problem directly. This is interesting because we need to accelerate certain complex operations that cannot be carried out otherwise, especially with the current and future solar telescopes. However, we are also starting to witness a huge revolution in solar physics in which deep learning models are informed with physical models. A good synergy can be obtained if the physical laws that we currently use to interpret our observations are used together with neural networks to approximate the most complex parts of the process. We will surely see new methods for the inversion of the Stokes profiles, new methods for the extrapolation of magnetic fields, new methods to accelerate MHD simulations, new methods to understand synoptic observations, and many more. All of them will use deep learning as a key ingredient.

Acknowledgements AAR acknowledges financial support from the Spanish Ministerio de Ciencia, Innovación y Universidades through project PGC2018-102108-B-I00 and FEDER funds. I.C. acknowledges the support of the Coronagraphic German and US Solar Probe Plus Survey (CGAUSS) project for WISPR by the German Aerospace Center (DLR) under grant 50OL1901. This research has made use of NASA's Astrophysics Data System Bibliographic Services. R.G. acknowledges to Fundação para a Ciência e a Tecnologia (FCT) the support through the research grants UIDB/04434/2020 and UIDP/04434/2020.

[43] https://github.com/EBroock/FarNet-II.

References

Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, Corrado GS, Davis A, Dean J, Devin M, Ghemawat S, Goodfellow I, Harp A, Irving G, Isard M, Jia Y, Jozefowicz R, Kaiser L, Kudlur M, Levenberg J, Mané D, Monga R, Moore S, Murray D, Olah C, Schuster M, Shlens J, Steiner B, Sutskever I, Talwar K, Tucker P, Vanhoucke V, Vasudevan V, Viégas F, Vinyals O, Warden P, Wattenberg M, Wicke M, Yu Y, Zheng X (2015) TensorFlow: large-scale machine learning on heterogeneous systems. https://www.tensorflow.org/

Allred JC, Kowalski AF, Carlsson M (2015) A unified computational model for solar and stellar flares. Astrophys J 809(1):104. https://doi.org/10.1088/0004-637X/809/1/104. arXiv:1507.04375 [astro-ph.SR]

Altschuler MD, Newkirk G (1969) Magnetic fields and the structure of the solar corona. I: methods of calculating coronal fields. Sol Phys 9(1):131–149. https://doi.org/10.1007/BF00145734

Ardizzone L, Kruse J, Wirkert S, Rahner D, Pellegrini EW, Klessen RS, Maier-Hein L, Rother C, Köthe U (2018) Analyzing inverse problems with invertible neural networks. arXiv e-prints arXiv:1808.04730 [cs.LG]

Armstrong JA, Fletcher L (2019) Fast solar image classification using deep learning and its importance for automation in solar physics. Sol Phys 294(6):80. https://doi.org/10.1007/s11207-019-1473-z. arXiv:1905.13575 [astro-ph.SR]

Asensio Ramos A (2010) Compressed sensing for next generation instruments. Astron Nachr 331(6):652. https://doi.org/10.1002/asna.201011394

Asensio Ramos A (2012) Extracting information from the data flood of new solar telescopes: brainstorming. In: Rimmele TR, Tritschler A, Wöger F, Collados Vera M, Socas-Navarro H, Schlichenmaier R, Carlsson M, Berger T, Cadavid A, Gilbert PR, Goode PR, Knölker M (eds) Second ATST-EAST meeting: magnetic fields from the photosphere to the corona. ASP conference series, vol 463. Astronomical Society of the Pacific, p 215

Asensio Ramos A (2016) Random sub-Nyquist polarimetric modulator. Appl Opt 55(6):1324. https://doi.org/10.1364/AO.55.001324. arXiv:1601.05211 [astro-ph.IM]

Asensio Ramos A, de la Cruz Rodríguez J (2015) Sparse inversion of stokes profiles. I. Two-dimensional Milne-Eddington inversions. Astron Astrophys 577:A140. https://doi.org/10.1051/0004-6361/201425508. arXiv:1503.07666 [astro-ph.SR]

Asensio Ramos A, Díaz Baso CJ (2019) Stokes inversion based on convolutional neural networks. Astron Astrophys 626:A102. https://doi.org/10.1051/0004-6361/201935628. arXiv:1904.03714 [astro-ph.SR]

Asensio Ramos A, López Ariste A (2010) Compressive sensing for spectroscopy and polarimetry. Astron Astrophys 509:A49 arXiv:0909.4439

Asensio Ramos A, Manso Sainz R (2012) Signal detection for spectroscopy and polarimetry. Astron Astrophys 547:A113. https://doi.org/10.1051/0004-6361/201220124. arXiv:1209.6455 [astro-ph.SR]

Asensio Ramos A, Olspert N (2021) Learning to do multiframe wavefront sensing unsupervised: applications to blind deconvolution. Astron Astrophys 646:A100. https://doi.org/10.1051/0004-6361/202038552. arXiv:2006.01438 [astro-ph.IM]

Asensio Ramos A, Socas-Navarro H (2005) An artificial neural network approach to the solution of molecular chemical equilibrium. Astron Astrophys 438:1021–1028. https://doi.org/10.1051/0004-6361:20052865. arXiv:astro-ph/0505322

Asensio Ramos A, Martínez González MJ, López Ariste A, Trujillo Bueno J, Collados M (2007a) A Near-infrared line of Mn I as a diagnostic tool of the average magnetic energy in the solar photosphere. Astrophys J 659(1):829–847. https://doi.org/10.1086/511951. arXiv:astro-ph/0612389 [astro-ph]

Asensio Ramos A, Martínez González MJ, Rubiño-Martín JA (2007b) Bayesian inversion of Stokes profiles. Astron Astrophys 476(2):959–970. https://doi.org/10.1051/0004-6361:20078107. arXiv:0709.0596 [astro-ph]

Asensio Ramos A, Socas-Navarro H, López Ariste A, Martínez González MJ (2007c) The intrinsic dimensionality of spectropolarimetric data. Astrophys J 660(2):1690–1699. https://doi.org/10.1086/513069. arXiv:astro-ph/0701604 [astro-ph]

Asensio Ramos A, de la Cruz Rodríguez J, Martínez González MJ, Pastor Yabar A (2016) Inversion of Stokes profiles with systematic effects. Astron Astrophys 590:A87. https://doi.org/10.1051/0004-6361/201628387. arXiv:1604.05470 [astro-ph.SR]

Asensio Ramos A, Requerey IS, Vitas N (2017) DeepVel: Deep learning for the estimation of horizontal velocities at the solar surface. Astron Astrophys 604:A11. https://doi.org/10.1051/0004-6361/201730783. arXiv:1703.05128 [astro-ph.SR]

Asensio Ramos A, de la Cruz Rodríguez J, Pastor Yabar A (2018) Real-time, multiframe, blind deconvolution of solar images. Astron Astrophys 620:A73. https://doi.org/10.1051/0004-6361/201833648. arXiv:1806.07150 [astro-ph.SR]

Baek JH, Kim S, Choi S, Park J, Kim J, Jo W, Kim D (2021) Solar event detection using deep-learning-based object detection methods. Sol Phys 296(11):160. https://doi.org/10.1007/s11207-021-01902-5

Baraniuk R (2007) Compressive sensing. IEEE Signal Process Mag 24:118–121

Barnes G, Leka KD (2006) Photospheric magnetic field properties of flaring versus flare-quiet active regions. III. Magnetic charge topology models. Astrophys J 646(2):1303–1318. https://doi.org/10.1086/504960

Barnes G, Leka KD (2008) Evaluating the performance of solar flare forecasting methods. Astrophys J 688 (2):L107. https://doi.org/10.1086/595550

Barnes G, Leka KD, Schrijver CJ, Colak T, Qahwaji R, Ashamari OW, Yuan Y, Zhang J, McAteer RTJ, Bloomfield DS, Higgins PA, Gallagher PT, Falconer DA, Georgoulis MK, Wheatland MS, Balch C, Dunn T, Wagner EL (2016) A comparison of flare forecasting methods. I. Results from the "all-clear" workshop. Astrophys J 829(2):89. https://doi.org/10.3847/0004-637X/829/2/89. arXiv:1608.06319 [astro-ph.SR]

Barra V, Delouille V, Hochedez JF (2008) Segmentation of extreme ultraviolet solar images via multichannel fuzzy clustering. Adv Space Res 42(5):917–925. https://doi.org/10.1016/j.asr.2007.10.021

Barra V, Delouille V, Kretzschmar M, Hochedez JF (2009) Fast and robust segmentation of solar EUV images: algorithm and results for solar cycle 23. Astron Astrophys 505(1):361–371. https://doi.org/10.1051/0004-6361/200811416

Barredo Arrieta A, Díaz-Rodríguez N, Del Ser J, Bennetot A, Tabik S, Barbado A, Garcia S, Gil-Lopez S, Molina D, Benjamins R, Chatila R, Herrera F (2020) Explainable artificial intelligence (XAI): concepts, taxonomies, opportunities and challenges toward responsible AI. Inform Fusion 58:82–115. https://doi.org/10.1016/j.inffus.2019.12.012

Barrett DGT, Dherin B (2020) Implicit gradient regularization. arXiv e-prints arXiv:2009.11162 [cs.LG]

Benson B, Pan WD, Prasad A, Gary GA, Hu Q (2020) Forecasting solar cycle 25 using deep neural networks. Sol Phys 295(5):65. https://doi.org/10.1007/s11207-020-01634-y. arXiv:2005.12406 [astro-ph.SR]

Benvenuto F, Piana M, Campi C, Massone AM (2018) A hybrid supervised/unsupervised machine learning approach to solar flare prediction. Astrophys J 853(1):90. https://doi.org/10.3847/1538-4357/aaa23c. arXiv:1706.07103 [astro-ph.SR]

Bernoux G, Brunet A, Buchlin É, Janvier M, Sicard A (2022) Forecasting the geomagnetic activity several days in advance using neural networks driven by solar EUV imaging. J Geophys Res 127(10): e2022JA030868. https://doi.org/10.1029/2022JA030868

Bintsi KM, Jarolim R, Tremblay B, Santos M, Jungbluth A, Mason JP, Sundaresan S, Vourlidas A, Downs C, Caplan RM, Muñoz Jaramillo A (2022) SuNeRF: validation of a 3D global reconstruction of the solar corona using simulated EUV images. arXiv e-prints arXiv:2211.14879 [astro-ph.SR]

Bishop CM (1996) Neural networks for pattern recognition. Oxford University Press, Oxford

Bizzarri I, Barghini D, Mancuso S, Alessio S, Rubinetti S, Taricco C (2022) Forecasting the solar cycle 25 using a multistep Bayesian neural network. MNRAS 515(4):5062–5070. https://doi.org/10.1093/mnras/stac2013

Bloomfield DS, Higgins PA, McAteer RTJ, Gallagher PT (2012) Toward reliable benchmarking of solar flare forecasting methods. Astrophys J 747(2):L41. https://doi.org/10.1088/2041-8205/747/2/l41

Bobra MG, Couvidat S (2015) Solar flare prediction using SDO/HMI vector magnetic field data with a machine-learning algorithm. Astrophys J 798(2):135. https://doi.org/10.1088/0004-637X/798/2/135. arXiv:1411.1405 [astro-ph.SR]

Bobra MG, Sun X, Hoeksema JT, Turmon M, Liu Y, Hayashi K, Barnes G, Leka KD (2014) The Helioseismic and Magnetic Imager (HMI) vector magnetic field pipeline: SHARPs – space-weather HMI active region oatches. Sol Phys 289(9):3549–3578. https://doi.org/10.1007/s11207-014-0529-3. arXiv:1404.1879 [astro-ph.SR]

Borrero JM, Asensio Ramos A, Collados M, Schlichenmaier R, Balthasar H, Franz M, Rezaei R, Kiess C, Orozco Suárez D, Pastor A, Berkefeld T, von der Lühe O, Schmidt D, Schmidt W, Sigwarth M, Soltau D, Volkmer R, Waldmann T, Denker C, Hofmann A, Staude J, Strassmeier KG, Feller A, Lagg A, Solanki SK, Sobotka M, Nicklas H (2016) Deep probing of the photospheric sunspot penumbra: no evidence of field-free gaps. Astron Astrophys 596:A2. https://doi.org/10.1051/0004-6361/201628313. arXiv:1607.08165 [astro-ph.SR]

Borrero JM, Franz M, Schlichenmaier R, Collados M, Asensio Ramos A (2017) Penumbral thermal structure below the visible surface. Astron Astrophys 601:L8. https://doi.org/10.1051/0004-6361/201730753. arXiv:1705.02832 [astro-ph.SR]

Bortnik J, Camporeale E (2021) Ten ways to apply machine learning in Earth and space sciences. Eos. https://doi.org/10.1029/2021EO160257

Bourlard H, Kamp Y (1988) Auto-association by multilayer perceptrons and singular value decomposition. Biol Cybern 59(4):291–294

Bradbury J, Frostig R, Hawkins P, Johnson MJ, Leary C, Maclaurin D, Necula G, Paszke A, VanderPlas J, Wanderman-Milne S, Zhang Q (2018) JAX: composable transformations of Python+NumPy programs. http://github.com/google/jax

Broock EG, Felipe T, Asensio Ramos A (2021) Performance of solar far-side active region neural detection. Astron Astrophys 652:A132. https://doi.org/10.1051/0004-6361/202141006. arXiv:2106.09365 [astro-ph.SR]

Broock EG, Asensio Ramos A, Felipe T (2022) FarNet-II: An improved solar far-side active region detection method. Astron Astrophys 667:A132. https://doi.org/10.1051/0004-6361/202244206

Brown EJE, Svoboda F, Meredith NP, Lane N, Horne RB (2022) Attention-based machine vision models and techniques for solar wind speed forecasting using solar EUV images. Space Weather 20(3): e2021SW002976. https://doi.org/10.1029/2021SW002976

Camporeale E (2019) The challenge of machine learning in Space Weather: Nowcasting and forecasting. Space Weather 17:1166–1207. https://doi.org/10.1029/2018SW002061. arXiv:1903.05192 [physics.space-ph]

Candès E, Romberg J, Tao T (2006) Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. IEEE Trans Inf Theory 52:489

Candès E, Romberg J, Tao T (2006) Stable signal recovery from incomplete and inaccurate measurements. Comm Pure Appl Math 59:1207

Candès EJ, Wakin MB (2008) An introduction to compressive sampling. IEEE Signal Process Mag 25:21–30

Carlsson M, Stein RF (1992) Non-LTE radiating acoustic shocks and CA II K2V bright points. Astrophys J 397:L59. https://doi.org/10.1086/186544

Carlsson M, Stein RF (1995) Does a nonmagnetic solar chromosphere exist? Astrophys J 440:L29. https://doi.org/10.1086/187753. arXiv:astro-ph/9411036 [astro-ph]

Carlsson M, Stein RF (1997) Formation of solar calcium H and K bright grains. Astrophys J 481(1):500–514. https://doi.org/10.1086/304043

Carroll TA, Kopf M (2008) Zeeman-tomography of the solar photosphere. Three-dimensional surface structures retrieved from Hinode observations. Astron Astrophys 481:L37–L40. https://doi.org/10.1051/0004-6361:20079197. arXiv:0803.1048 [astro-ph]

Carroll TA, Staude J (2001) The inversion of Stokes profiles with artificial neural networks. Astron Astrophys 378:316–326. https://doi.org/10.1051/0004-6361:20011167

Casini R, Li W (2019) Removal of spectro-polarimetric fringes by two-dimensional principal component analysis. Astrophys J 872(2):173. https://doi.org/10.3847/1538-4357/ab0023

Casini R, López Ariste A, Tomczyk S, Lites BW (2003) Magnetic maps of prominences from full Stokes analysis of the He I D3 line. Astrophys J 598(1):L67–L70. https://doi.org/10.1086/380496

Casini R, Bevilacqua R, López Ariste A (2005) Principal component analysis of the He I D_3 polarization profiles from solar prominences. Astrophys J 622(2):1265–1274. https://doi.org/10.1086/428283

Casini R, López Ariste A, Paletou F, Léger L (2009) Multi-line Stokes inversion for prominence magnetic-field diagnostics. Astrophys J 703(1):114–120. https://doi.org/10.1088/0004-637X/703/1/114. arXiv:0906.2144 [astro-ph.IM]

Casini R, Asensio Ramos A, Lites BW, López Ariste A (2013) Improved search of principal component analysis databases for spectro-polarimetric inversion. Astrophys J 773(2):180

Chappell BA, Pereira TMD (2022) SunnyNet: a neural network approach to 3D non-LTE radiative transfer. Astron Astrophys 658:A182. https://doi.org/10.1051/0004-6361/202142625. arXiv:2112.13852 [astro-ph.SR]

Cheung MCM, Boerner P, Schrijver CJ, Testa P, Chen F, Peter H, Malanushenko A (2015) Thermal diagnostics with the atmospheric imaging assembly on board the solar dynamics observatory: a validated method for differential emission measure inversions. Astrophys J 807(2):143. https://doi.org/10.1088/0004-637X/807/2/143. arXiv:1504.03258 [astro-ph.SR]

Cheung MCM, De Pontieu B, Martínez-Sykora J, Testa P, Winebarger AR, Daw A, Hansteen V, Antolin P, Tarbell TD, Wuelser JP, Young P, MUSE Team (2019) Multi-component decomposition of astronomical spectra by compressed sensing. Astrophys J 882(1):13. https://doi.org/10.3847/1538-4357/ab263d. arXiv:1902.03890 [astro-ph.SR]

Chifu I, Gafeira R (2021) 3d solar coronal loop reconstructions with machine learning. Astrophys J 910(1):L10. https://doi.org/10.3847/2041-8213/abed53

Clevert DA, Unterthiner T, Hochreiter S (2015) Fast and accurate deep network learning by exponential linear units (ELUs). arXiv e-prints arXiv:1511.07289 [cs.LG]

Colak T, Qahwaji R (2013) Prediction of Extreme ultraviolet Variability Experiment (EVE)/Extreme ultraviolet Spectro-Photometer (ESP) irradiance from Solar Dynamics Observatory (SDO)/Atmospheric Imaging Assembly (AIA) images using fuzzy image processing and machine learning. Sol Phys 283(1):143–156. https://doi.org/10.1007/s11207-011-9880-9

Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn, pp 273–297

Covas E, Peixinho N, Fernandes J (2019) Neural network forecast of the sunspot butterfly diagram. Sol Phys 294(3):24. https://doi.org/10.1007/s11207-019-1412-z. arXiv:1801.04435 [astro-ph.SR]

Cranmer SR (2009) Coronal Holes. Living Rev Sol Phys 6:3. https://doi.org/10.12942/lrsp-2009-3. arXiv:0909.2847 [astro-ph.SR]

Cybenko G (1988) Approximation by superpositions of a sigmoidal function. Tech. rep., insttuftscs

Dani T, Sulistiani S (2019) Prediction of maximum amplitude of solar cycle 25 using machine learning. J Phys Conf Ser 1231:012022. https://doi.org/10.1088/1742-6596/1231/1/012022

de la Cruz Rodríguez J, van Noort M (2017) Radiative diagnostics in the solar photosphere and chromosphere. Space Sci Rev 210(1–4):109–143. https://doi.org/10.1007/s11214-016-0294-8. arXiv:1609.08324 [astro-ph.SR]

de la Cruz Rodríguez J, Leenaarts J, Danilovic S, Uitenbroek H (2019) STiC: a multiatom non-LTE PRD inversion code for full-Stokes solar observations. Astron Astrophys 623:A74. https://doi.org/10.1051/0004-6361/201834464. arXiv:1810.08441 [astro-ph.SR]

De Pontieu B, Title AM, Lemen JR, Kushner GD, Akin DJ, Allard B, Berger T, Boerner P, Cheung M, Chou C (2014) The interface region imaging spectrograph (IRIS). Sol Phys 289(7):2733–2779. https://doi.org/10.1007/s11207-014-0485-y. arXiv:1401.2491 [astro-ph.SR]

De Pontieu B, Martínez-Sykora J, Testa P, Winebarger AR, Daw A, Hansteen V, Cheung MCM, Antolin P (2020) The multi-slit approach to coronal spectroscopy with the multi-slit solar explorer (MUSE). Astrophys J 888(1):3. https://doi.org/10.3847/1538-4357/ab5b03. arXiv:1909.08818 [astro-ph.IM]

del Toro Iniesta JC, López Ariste A (2003) An orthonormal set of Stokes profiles. Astron Astrophys 412:875–878

del Toro Iniesta JC, Ruiz Cobo B (2016) Inversion of the radiative transfer equation for polarized light. Living Rev Sol Phys 13:4. https://doi.org/10.1007/s41116-016-0005-2. arXiv:1610.10039 [astro-ph.SR]

Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L (2009) Imagenet: a large-scale hierarchical image database. In: CVPR09

Díaz Baso CJ, Asensio Ramos A (2018) Enhancing SDO/HMI images using deep learning. Astron Astrophys 614:A5. https://doi.org/10.1051/0004-6361/201731344. arXiv:1706.02933 [astro-ph.SR]

Díaz Baso CJ, de la Cruz Rodríguez J, Danilovic S (2019) Solar image denoising with convolutional neural networks. Astron Astrophys 629:A99. https://doi.org/10.1051/0004-6361/201936069. arXiv:1908.02815 [astro-ph.SR]

Díaz Baso CJ, Asensio Ramos A, de la Cruz Rodríguez J (2022) Bayesian Stokes inversion with normalizing flows. Astron Astrophys 659:A165. https://doi.org/10.1051/0004-6361/202142018. arXiv:2108.07089 [astro-ph.SR]

Díaz Castillo SM, Asensio Ramos A, Fischer CE, Berdyugina SV (2022) Towards the identification and classification of solar granulation structures using semantic segmentation. Front Astron Space Sci 9:896632. https://doi.org/10.3389/fspas.2022.896632

Dietterich TG, Lathrop RH, Lozano-Pérez T (1997) Solving the multiple instance problem with axis-parallel rectangles. Artif Intell 89(1):31–71. https://doi.org/10.1016/S0004-3702(96)00034-3

Dinh L, Krueger D, Bengio Y (2014) NICE: non-linear independent components estimation. arXiv e-prints arXiv:1410.8516 [cs.LG]

Domingo V, Fleck B, Poland AI (1995) SOHO: the solar and heliospheric observatory. Space Sci Rev 72 (1–2):81–84. https://doi.org/10.1007/BF00768758

Donoho D (2006) Compressed sensing. IEEE Trans Inf Theory 52:1289

Dou F, Xu L, Ren Z, Zhao D, Zhang X (2022) Super-resolution of solar magnetograms using deep learning. Res Astron Astrophys 22(8):085018. https://doi.org/10.1088/1674-4527/ac78ce

Erion G, Janizek J, Sturmfels P, Lundberg S, Lee SI (2021) Improving performance of deep learning models with axiomatic attribution priors and expected gradients. Nature Mach Intell 3:1–12. https://doi.org/10.1038/s42256-021-00343-w

Ervin T, Bortnik J, Downs C (2021) Coronal hole detection using machine learning techniques. UCLA Library https://escholarship.org/uc/item/5qm499f2

Ester M, Kriegel HP, Sander J, Xu X (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: KDD-96 Proceedings. AAAI Press, pp 226–231

Falconer DA (2001) A prospective method for predicting coronal mass ejections from vector magnetograms. J Geophys Res 106(A11):25185–25190. https://doi.org/10.1029/2000JA004005

Falconer DA, Moore RL, Gary GA (2002) Correlation of the coronal mass ejection productivity of solar active regions with measures of their global nonpotentiality from vector magnetograms: baseline results. Astrophys J 569(2):1016–1025. https://doi.org/10.1086/339161

Falconer DA, Moore RL, Gary GA (2003) A measure from line-of-sight magnetograms for prediction of coronal mass ejections. J Geophys Res 108(A10):1380. https://doi.org/10.1029/2003JA010030

Felipe T, Asensio Ramos A (2019) Improved detection of far-side solar active regions using deep learning. Astron Astrophys 632:A82. https://doi.org/10.1051/0004-6361/201936838. arXiv:1911.01099 [astro-ph.SR]

Felipe T, Collados M, Khomenko E, Kuckein C, Asensio Ramos A, Balthasar H, Berkefeld T, Denker C, Feller A, Franz M, Hofmann A, Joshi J, Kiess C, Lagg A, Nicklas H, Orozco Suárez D, Pastor Yabar A, Rezaei R, Schlichenmaier R, Schmidt D, Schmidt W, Sigwarth M, Sobotka M, Solanki SK, Soltau D, Staude J, Strassmeier KG, Volkmer R, von der Lühe O, Waldmann T (2016) Three-dimensional structure of a sunspot light bridge. Astron Astrophys 596:A59. https://doi.org/10.1051/0004-6361/201629586. arXiv:1611.04803 [astro-ph.SR]

Fouhey DF, Higgins REL, Antiochos SK, Barnes G, DeRosa ML, Hoeksema JT, Leka KD, Liu Y, Schuck PW, Gombosi TI (2022) Large-scale spatial cross-calibration of hinode/SOT-SP and SDO/HMI. arXiv e-prints arXiv:2209.15036 [astro-ph.SR]

Gafeira R, Orozco Suárez D, Milić I, Quintero Noda C, Ruiz Cobo B, Uitenbroek H (2021) Machine learning initialization to accelerate Stokes profile inversions. Astron Astrophys 651:A31. https://doi.org/10.1051/0004-6361/201936910. arXiv:2103.09651 [astro-ph.IM]

Galvez R, Fouhey DF, Jin M, Szenicer A, Muñoz-Jaramillo A, Cheung MCM, Wright PJ, Bobra MG, Liu Y, Mason J, Thomas R (2019) A machine-learning data set prepared from the NASA solar dynamics observatory mission. Astrophys J Suppl Ser 242(1):7. https://doi.org/10.3847/1538-4365/ab1005. arXiv:1903.04538 [astro-ph.SR]

Galvez R, Fouhey DF, Jin M, Szenicer A, Muñoz-Jaramillo A, Cheung MCM, Wright PJ, Bobra MG, Liu Y, Mason J, Thomas R (2019) A machine-learning data set prepared from the NASA solar dynamics observatory mission. Astrophys J Suppl Ser 242(1):7. https://doi.org/10.3847/1538-4365/ab1005. arXiv:1903.04538 [astro-ph.SR]

Garton TM, Gallagher PT, Murray SA (2018) Automated coronal hole identification via multi-thermal intensity segmentation. J Space Weather Space Clim 8:A02. https://doi.org/10.1051/swsc/2017039. arXiv:1711.11476 [astro-ph.SR]

Gary GA (2001) Plasma beta above a solar active region: rethinking the paradigm. Sol Phys 203(1):71–86. https://doi.org/10.1023/A:1012722021820

Gavish M, Donoho DL (2014) The optimal hard threshold for singular values is 4/sqrt(3). arXiv:1305.5870 [stat.ME]

Glorot X, Bengio Y (2010) Understanding the difficulty of training deep feedforward neural networks. In: Teh YW, Titterington M (eds) Proceedings of the thirteenth international conference on artificial intelligence and statistics. Proceedings of machine learning research, vol 9. PMLR, Chia Laguna Resort, Sardinia, Italy, pp 249–256. http://proceedings.mlr.press/v9/glorot10a.html

Golub L, Cheimets P, DeLuca EE, Madsen CA, Reeves KK, Samra J, Savage S, Winebarger A, Bruccoleri AR (2020) EUV imaging and spectroscopy for improved space weather forecasting. J Space Weather Space Clim 10:37. https://doi.org/10.1051/swsc/2020040

Goodfellow I, Bengio Y, Courville A (2016) Deep Learning. MIT Press. http://www.deeplearningbook.org

Gošić M, Bellot Rubio LR, Orozco Suárez D, Katsukawa Y, del Toro Iniesta JC (2014) The Solar Internetwork. I. Contribution to the network magnetic flux. Astrophys J 797(1):49. https://doi.org/10.1088/0004-637X/797/1/49. arXiv:1408.2369 [astro-ph.SR]

Grossmann-Doerth U, Schüssler M, Solanki SK (1988) Unshifted, asymmetric Stokes V-profiles—possible solution of a riddle. Astron Astrophys 206(2):L37–L39

Hastie T, Tibshirani R, Friedman J (2009) The elements of statistical learning, 2nd edn. Springer Series in Statistics, Springer, New York. https://doi.org/10.1007/978-0-387-84858-7

Hathaway DH (2015) The solar cycle. Living Rev Sol Phys 12:4. https://doi.org/10.1007/lrsp-2015-4. arXiv:1502.07020 [astro-ph.SR]

He K, Zhang X, Ren S, Sun J (2015) Delving deep into rectifiers: surpassing human-level performance on ImageNet classification. arXiv e-prints arXiv:1502.01852 [cs.CV]

He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. pp 770–778. https://doi.org/10.1109/CVPR.2016.90

Henney CJ, Harvey JW (2005) Automated coronal hole detection using He 1083 nm spectroheliograms and photospheric magnetograms. In: Sankarasubramanian K, Penn M, Pevtsov A (eds) Large-scale structures and their role in solar activity. ASP Conference Series, vol 346. Astronomical Society of the Pacific, p 261. arXiv:astro-ph/0701122 [astro-ph]

Higgins REL, Fouhey DF, Zhang D, Antiochos SK, Barnes G, Hoeksema JT, Leka KD, Liu Y, Schuck PW, Gombosi TI (2021) Fast and accurate emulation of the SDO/HMI Stokes inversion with uncertainty quantification. Astron Astrophys 911(2):130. https://doi.org/10.3847/1538-4357/abd7fe

Higgins REL, Fouhey DF, Antiochos SK, Barnes G, Cheung MCM, Hoeksema JT, Leka KD, Liu Y, Schuck PW, Gombosi TI (2022) SynthIA: a synthetic inversion approximation for the stokes vector fusing SDO and Hinode into a virtual observatory. Astrophys J Suppl Ser 259(1):24. https://doi.org/10.3847/1538-4365/ac42d5

Hinton GE, Roweis S (2002) Stochastic neighbor embedding. In: Becker S, Thrun S, Obermayer K (eds) Advances in neural information processing systems. vol 15. MIT Press. https://proceedings.neurips.cc/paper_files/paper/2002/file/6150ccc6069bea6b5716254057a194ef-Paper.pdf

Ho J, Jain A, Abbeel P (2020) Denoising diffusion probabilistic models. In: Larochelle H, Ranzato M, Hadsell R, Balcan M, Lin H (eds) Advances in Neural Information Processing Systems. vol 33. Curran Associates, pp 6840–6851

Hochreiter S, Schmidhuber J (1997) Flat minima. Neural Comput 9(1):1–42. https://doi.org/10.1162/neco.1997.9.1.1

Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Huang X, Wang H, Xu L, Liu J, Li R, Dai X (2018) Deep learning based solar flare forecasting model. I. results for line-of-sight magnetograms. Astrophys J 856(1):7. https://doi.org/10.3847/1538-4357/aaae00

Hurlburt N, Cheung M, Schrijver C, Chang L, Freeland S, Green S, Heck C, Jaffey A, Kobashi A, Schiff D, Serafin J, Seguin R, Slater G, Somani A, Timmons R (2012) Heliophysics event knowledgebase for the solar dynamics observatory (SDO) and beyond. Sol Phys 275(1–2):67–78. https://doi.org/10.1007/s11207-010-9624-2. arXiv:1008.1291 [astro-ph.IM]

Huwyler C, Melchior M (2022) Using multiple instance learning for explainable solar flare prediction. Astron Comput 41:100668. https://doi.org/10.1016/j.ascom.2022.100668

Illarionov E, Kosovichev A, Tlatov A (2020) Machine-learning approach to identification of coronal holes in solar disk images and synoptic maps. Astrophys J 903(2):115. https://doi.org/10.3847/1538-4357/abb94d. arXiv:2006.08529 [astro-ph.SR]

Illarionov EA, Tlatov AG (2018) Segmentation of coronal holes in solar disc images with a convolutional neural network. MNRAS 481(4):5014–5021. https://doi.org/10.1093/mnras/sty2628. arXiv:1809.05748 [astro-ph.SR]

Inceoglu F, Shprits YY, Heinemann SG, Bianco S (2022) Identification of coronal holes on AIA/SDO images using unsupervised machine learning. Astrophys J 930(2):118. https://doi.org/10.3847/1538-4357/ac5f43. arXiv:2203.10491 [astro-ph.SR]

Ioffe S, Szegedy C (2015) Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: Blei D, Bach F (eds) Proceedings of the 32nd international conference on machine learning (ICML-15). JMLR workshop and conference proceedings, pp 448–456. http://jmlr.org/proceedings/papers/v37/ioffe15.pdf

Isola P, Zhu JY, Zhou T, Efros AA (2017) Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). pp 1125–1134. arXiv:1611.07004 [cs.CV]

Jarolim R, Veronig AM, Hofmeister S, Heinemann SG, Temmer M, Podladchikova T, Dissauer K (2021) Multi-channel coronal hole detection with convolutional neural networks. Astron Astrophys 652:A13. https://doi.org/10.1051/0004-6361/202140640. arXiv:2104.14313 [astro-ph.SR]

Jarolim R, Thalmann J, Veronig A, Podladchikova T (2022) Probing the solar coronal magnetic field with physics-informed neural networks. https://doi.org/10.21203/rs.3.rs-1415262/v1

Jiang H, Wang J, Liu C, Jing J, Liu H, Wang JTL, Wang H (2020) Identifying and tracking solar magnetic flux elements with deep learning. Astrophys J Suppl Ser 250(1):5. https://doi.org/10.3847/1538-4365/aba4aa. arXiv:2008.12080 [astro-ph.SR]

Jiang H, Li Q, Hu Z, Liu N, Abduallah Y, Jing J, Zhang G, Xu Y, Hsu W, Wang JTL, Wang H (2022) A deep learning approach to generating photospheric vector magnetograms of solar active regions for SOHO/MDI Using SDO/HMI and BBSO Data. arXiv e-prints arXiv:2211.02278 [astro-ph.SR]

Jurčák J, Štěpán J, Trujillo Bueno J, Bianda M (2018) Comparison of theoretical and observed Ca II 8542 Stokes profiles in quiet regions at the centre of the solar disc. Astron Astrophys 619:A60. https://doi.org/10.1051/0004-6361/201732265. arXiv:1808.09470 [astro-ph.SR]

Kasper JC, Klein KG, Lichko E, Huang J, Chen CHK, Badman ST, Bonnell J, Whittlesey PL, Livi R, Larson D, Pulupa M, Rahmati A, Stansby D, Korreck KE, Stevens M, Case AW, Bale SD, Maksimovic M, Moncuquet M, Goetz K, Halekas JS, Malaspina D, Raouafi NE, Szabo A, MacDowall R, Velli M, Dudok de Wit T, Zank GP (2021) Parker solar probe enters the magnetically dominated solar corona. Phys Rev Lett 127(25):255101. https://doi.org/10.1103/PhysRevLett.127.255101

Kim T, Park E, Lee H, Moon YJ, Bae SH, Lim D, Jang S, Kim L, Cho IH, Choi M, Cho KS (2019) Solar farside magnetograms from deep learning analysis of STEREO/EUVI data. Nature Astron 3:397–400. https://doi.org/10.1038/s41550-019-0711-5

Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. ArXiv e-prints arXiv:1412.6980 [cs.LG]

Kingma DP, Welling M (2014) Auto-encoding variational bayes. In: 2nd International conference on learning representations, ICLR 2014, Banff, AB, Canada, April 14–16, 2014, Conference track proceedings

Kleint L, Battaglia M, Reardon K, Sainz Dalda A, Young PR, Krucker S (2015) The fast filament eruption leading to the X-flare on 2014 March 29. Astrophys J 806(1):9. https://doi.org/10.1088/0004-637X/806/1/9. arXiv:1504.00515 [astro-ph.SR]

Kohonen T (2001) Self-organizing maps. Springer, Berlin. https://doi.org/10.1007/978-3-642-56927-2

Kolen JF, Kremer SC (2001) gradient flow in recurrent nets: the difficulty of learning longterm dependencies, Wiley-IEEE Press, pp 237–243. https://doi.org/10.1109/9780470544037.ch14

Lamy PL, Floyd O, Boclet B, Wojak J, Gilardy H, Barlyaeva T (2019) Coronal mass ejections over solar cycles 23 and 24. Space Sci Rev 215(5):39. https://doi.org/10.1007/s11214-019-0605-y

Landi Degl'Innocenti E, Landolfi M (2004) Polarization in spectral lines. Astrophysics and Space Science Library, vol 307. Kluwer Academic Publishers, Dordrecht. https://doi.org/10.1007/1-4020-2415-0

Lavasa E, Giannopoulos G, Papaioannou A, Anastasiadis A, Daglis IA, Aran A, Pacheco D, Sanahuja B (2021) Assessing the predictability of solar energetic particles with the use of machine learning techniques. Sol Phys 296(7):107. https://doi.org/10.1007/s11207-021-01837-x

LeCun Y, Bengio Y (1998) Convolutional networks for images, speech, and time series. In: Arbib MA (ed) The handbook of brain theory and neural networks. MIT Press, Cambridge, pp 255–258

LeCun Y, Bottou L, Orr GB, Müller KR (1998) Efficient backprop. In: Montavon G, Orr GB, Müller KR (eds) Neural networks: tricks of the trade. Lecture notes in computer science, vol 1524. Springer, Berlin, pp 9–50. https://doi.org/10.1007/3-540-49430-8_2

Lehtinen J, Munkberg J, Hasselgren J, Laine S, Karras T, Aittala M, Aila T (2018) Noise2Noise: learning image restoration without clean data. arXiv e-prints arXiv:1803.04189 [cs.CV]

Leka KD, Barnes G (2003) Photospheric magnetic field properties of flaring versus flare-quiet active regions. I. Data, general approach, and sample results. Astrophys J 595(2):1277–1295. https://doi.org/10.1086/377511

Leka KD, Barnes G (2003) Photospheric magnetic field properties of flaring versus flare-quiet active regions. II. Discriminant analysis. Astrophys J 595(2):1296–1306. https://doi.org/10.1086/377512

Leka KD, Barnes G (2007) Photospheric magnetic field properties of flaring versus flare-quiet active regions. IV. A statistically significant sample. Astrophys J 656(2):1173–1186. https://doi.org/10.1086/510282

Leka KD, Park SH, Kusano K, Andries J, Barnes G, Bingham S, Bloomfield DS, McCloskey AE, Delouille V, Falconer D, Gallagher PT, Georgoulis MK, Kubo Y, Lee K, Lee S, Lobzin V, Mun J, Murray SA, Nageem TAMH, Qahwaji R, Sharpe M, Steenburgh RA, Steward G, Terkildsen M (2019a) A comparison of flare forecasting methods. II. Benchmarks, metrics, and performance results for operational solar flare forecasting systems. Astrophys J Suppl Ser 243(2):36. https://doi.org/10.3847/1538-4365/ab2e12

Leka KD, Park SH, Kusano K, Andries J, Barnes G, Bingham S, Bloomfield DS, McCloskey AE, Delouille V, Falconer D, Gallagher PT, Georgoulis MK, Kubo Y, Lee K, Lee S, Lobzin V, Mun J, Murray SA, Nageem TAMH, Qahwaji R, Sharpe M, Steenburgh RA, Steward G, Terkildsen M (2019b) A comparison of flare forecasting methods. III. Systematic behaviors of operational solar flare forecasting systems. Astrophys J 881(2):101. https://doi.org/10.3847/1538-4357/ab2e11

Lemen JR, Title AM, Akin DJ, Boerner PF, Chou C, Drake JF, Duncan DW, Edwards CG, Friedlaender FM, Heyman GF, Hurlburt NE, Katz NL, Kushner GD, Levay M, Lindgren RW, Mathur DP, McFeaters EL, Mitchell S, Rehse RA, Schrijver CJ, Springer LA, Stern RA, Tarbell TD, Wuelser JP, Wolfson CJ, Yanari C, Bookbinder JA, Cheimets PN, Caldwell D, Deluca EE, Gates R, Golub L, Park S, Podgorski WA, Bush RI, Scherrer PH, Gummin MA, Smith P, Auker G, Jerram P, Pool P, Soufli R, Windt DL, Beardsley S, Clapp M, Lang J, Waltham N (2012) The Atmospheric Imaging Assembly (AIA) on the Solar Dynamics Observatory (SDO). Sol Phys 275:17–40. https://doi.org/10.1007/s11207-011-9776-8

Levina E, Bickel PJ (2005) Maximum likelihood estimation of intrinsic dimension. In: Advances in neural information processing systems 17 (NIPS 2004)

Li Q, Wan M, Zeng SG, Zheng S, Deng LH (2021) Predicting the 25th solar cycle using deep learning methods based on sunspot area data. Res Astron Astrophys 21(7):184. https://doi.org/10.1088/1674-4527/21/7/184

Lim D, Moon YJ, Park E, Lee JY (2021) Selection of three (extreme) ultraviolet channels for solar satellite missions by deep learning. ApJL 915(2):L31. https://doi.org/10.3847/2041-8213/ac0d54

Lindsey C, Braun DC (1997) Helioseismic Holography. ApJ 485(2):895–903. https://doi.org/10.1086/304445

Linker JA, Caplan RM, Downs C, Riley P, Mikic Z, Lionello R, Henney CJ, Arge CN, Liu Y, Derosa ML, Yeates A, Owens MJ (2017) The open flux problem. Astrophys J 848(1):70. https://doi.org/10.3847/1538-4357/aa8a70. arXiv:1708.02342 [astro-ph.SR]

Lionello R, Linker JA, Mikić Z (2008) Multispectral emission of the sun during the first whole sun month: magnetohydrodynamic simulations. Astrophys J 690(1):902–912. https://doi.org/10.1088/0004-637x/690/1/902

Lites BW, Akin DL, Card G, Cruz T, Duncan DW, Edwards CG, Elmore DF, Hoffmann C, Katsukawa Y, Katz N, Kubo M, Ichimoto K, Shimizu T, Shine RA, Streander KV, Suematsu A, Tarbell TD, Title AM, Tsuneta S (2013) The hinode spectro-polarimeter. Sol Phys 283(2):579–599. https://doi.org/10.1007/s11207-012-0206-3

Liu G, Dundar A, Shih KJ, Wang TC, Reda FA, Sapra K, Yu Z, Yang X, Tao A, Catanzaro B (2022) Partial convolution for padding, inpainting, and image synthesis. IEEE Trans Pattern Anal Mach Intell. https://doi.org/10.1109/TPAMI.2022.3209702

Liu H, Liu C, Wang JTL, Wang H (2019) Predicting solar flares using a long short-term memory network. Astrophys J 877(2):121. https://doi.org/10.3847/1538-4357/ab1b3c. arXiv:1905.07095 [astro-ph.SR]

Liu J, Wang Y, Huang X, Korsós MB, Jiang Y, Wang Y, Erdélyi R (2021) Reliability of AI-generated magnetograms from only EUV images. Nature Astron 5(2):108–110. https://doi.org/10.1038/s41550-021-01310-6

Liu S, Xu L, Zhao Z, Erdélyi R, Korsós MB, Huang X (2022) Deep learning based solar flare forecasting model. II. Influence of image resolution. Astrophys J 941(1):20. https://doi.org/10.3847/1538-4357/ac99dc

Loève M (1955) Probability theory. D. Van Nostrand, New York

Löfdahl MG, Scharmer GB (1994) Wavefront sensing and image restoration from focused and defocused solar images. Astron Astrophys Suppl 107:243–264

Löfdahl MG, Berger TE, Shine RS, Title AM (1998) Preparation of a dual wavelength sequence of high-resolution solar photospheric images using phase diversity. Astrophys J 495:965

Löfdahl MG, Bones PJ, Fiddy MA, Millane RP (2002) Multi-frame blind deconvolution with linear equality constraints. In: Image reconstruction from incomplete data, vol 4792. pp 146–155. https://doi.org/10.1117/12.451791. arXiv:physics/0209004 [physics.optics]

López Ariste A (2014) Pattern recognition techniques in polarimetry. Proc IAU 10(S305):207–215. https://doi.org/10.1017/S1743921315004792

López Ariste A, Casini R (2002) Magnetic fields in prominences: inversion techniques for spectropolarimetric data of the He I D_3 Line. Astrophys J 575(1):529–541. https://doi.org/10.1086/341260

López Ariste A, Casini R (2003) Improved estimate of the magnetic field in a prominence. Astrophys J 582(1):L51–L54. https://doi.org/10.1086/367600

López Ariste A, Casini R (2005) Inference of the magnetic field in spicules from spectropolarimetry of He I D3. Astron Astrophys 436(1):325–331. https://doi.org/10.1051/0004-6361:20042214

López Ariste A, Le Men C, Gelly B, Asensio Ramos A (2010) Double-pass spectro-imaging: TUNIS. Astron Nachr 331(6):658. https://doi.org/10.1002/asna.201011396

López Ariste A, Le Men C, Gelly B (2011) Double-pass spectroimaging with spectral multiplexing: TUNIS. Contrib Astron Obs Skalnate Pleso 41(2):99–105

MacBride CD, Jess DB, Grant SDT, Khomenko E, Keys PH, Stangalini M (2021) Accurately constraining velocity information from spectral imaging observations using machine learning techniques. Philos Trans R Soc A 379(2190):20200171. https://doi.org/10.1098/rsta.2020.0171

Mackovjak Š, Harman M, Maslej-Krešňáková V, Butka P (2021) SCSS-Net: solar corona structures segmentation by deep learning. Mon Not R Astron Soc 508(3):3111–3124. https://doi.org/10.1093/mnras/stab2536. arXiv:2109.10834 [astro-ph.SR]

MacQueen J (1967) Some methods for classification and analysis of multivariate observations. In: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, volume 1: statistics. University of California Press, Berkeley, Calif., pp 281–297. https://projecteuclid.org/euclid.bsmsp/1200512992

Martínez González MJ, Pastor Yabar A, Lagg A, Asensio Ramos A, Collados M, Solanki SK, Balthasar H, Berkefeld T, Denker C, Doerr HP, Feller A, Franz M, González Manrique SJ, Hofmann A, Kneer F, Kuckein C, Louis R, von der Lühe O, Nicklas H, Orozco D, Rezaei R, Schlichenmaier R, Schmidt D, Schmidt W, Sigwarth M, Sobotka M, Soltau D, Staude J, Strassmeier KG, Verma M, Waldman T, Volkmer R (2016) Inference of magnetic fields in the very quiet Sun. Astron Astrophys 596:A5. https://doi.org/10.1051/0004-6361/201628449. arXiv:1804.10089 [astro-ph.SR]

Martínez Pillet V, Del Toro Iniesta JC, Álvarez-Herrero A, Domingo V, Bonet JA, González Fernández L, López Jiménez A, Pastor C, Gasent Blesa JL, Mellado P, Piqueras J, Aparicio B, Balaguer M, Ballesteros E, Belenguer T, Bellot Rubio LR, Berkefeld T, Collados M, Deutsch W, Feller A, Girela F, Grauf B, Heredero RL, Herranz M, Jerónimo JM, Laguna H, Meller R, Menéndez M, Morales R, Orozco Suárez D, Ramos G, Reina M, Ramos JL, Rodríguez P, Sánchez A, Uribe-Patarroyo N, Barthol P, Gandorfer A, Knoelker M, Schmidt W, Solanki SK, Vargas Domínguez S (2011) The imaging magnetograph experiment (IMaX) for the sunrise Balloon–Borne solar observatory. Sol Phys 268:57–102. https://doi.org/10.1007/s11207-010-9644-y

McCulloch WS, Pitts WA (1943) A logical calculus of the ideas immanent in nervous activity. Bull Math Biophys 5:115–133

McKay MD, Beckman RJ, Conover WJ (1979) A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics 21(2):239–245

Medsker LR, Jain LC (2021) Recurrent neural network: design and applications. CRC Press

Mildenhall B, Srinivasan PP, Tancik M, Barron JT, Ramamoorthi R, Ng R (2020) NeRF: Representing scenes as neural radiance fields for view synthesis. In: Vedaldi A, Bischof H, Brox T, Frahm JM (eds)

Computer vision—ECCV 2020. Lecture notes in computer science, vol 12346. Springer, Cham, pp 405–421. https://doi.org/10.1007/978-3-030-58452-8_24

Milić I, Gafeira R (2020) Mimicking spectropolarimetric inversions using convolutional neural networks. Astron Astrophys 644:A129. https://doi.org/10.1051/0004-6361/201936537. arXiv:2006.02005 [astro-ph.SR]

Milić I, van Noort M (2018) Spectropolarimetric NLTE inversion code SNAPI. Astron Astrophys 617: A24. https://doi.org/10.1051/0004-6361/201833382. arXiv:1806.08134 [astro-ph.SR]

Miscuglio M, Sorger VJ (2020) Photonic tensor cores for machine learning. Appl Phys Rev 7(3):031404. https://doi.org/10.1063/5.0001942

Molnar M, Reardon K, Osborne C, Milić I (2020) Spectral deconvolution with deep learning: removing the effects of spectral PSF broadening. Front Astron Space Sci 7:29. https://doi.org/10.3389/fspas.2020.00029. arXiv:2005.05529 [astro-ph.SR]

Müller D, Nicula B, Felix S, Verstringe F, Bourgoignie B, Csillaghy A, Berghmans D, Jiggens P, García-Ortiz JP, Ireland J, Zahniy S, Fleck B (2017) JHelioviewer. Time-dependent 3D visualisation of solar and heliospheric data. Astron Astrophys 606:A10. https://doi.org/10.1051/0004-6361/201730893, arXiv:1705.07628 [astro-ph.SR]

Nair V, Hinton GE (2010) Rectified linear units improve restricted boltzmann machines. In: Proceedings of the 27th international conference on machine learning (ICML-10), June 21-24, 2010, Haifa, Israel. pp 807–814. http://www.icml2010.org/papers/432.pdf

Naitzat G, Zhitnikov A, Lim LH (2020) Topology of deep neural networks. arXiv e-prints arXiv:2004.06093 [cs.LG]

Nandy D (2021) Progress in solar cycle predictions: sunspot cycles 24–25 in perspective. Sol Phys 296 (3):54. https://doi.org/10.1007/s11207-021-01797-2. arXiv:2009.01908 [astro-ph.SR]

Nise NS (2000) Control systems engineering, 3rd edn. Wiley, New York

Nishizuka N, Sugiura K, Kubo Y, Den M, Ishii M (2018) Deep flare net (DeFN) model for solar flare prediction. Astrophys J 858(2):113. https://doi.org/10.3847/1538-4357/aab9a7

Nousiainen J, Rajani C, Kasper M, Helin T, Haffert SY, Vérinaud C, Males JR, Van Gorkom K, Close LM, Long JD, Hedglen AD, Guyon O, Schatz L, Kautz M, Lumbres J, Rodack A, Knight JM, Miller K (2022) Toward on-sky adaptive optics control using reinforcement learning. Model-based policy optimization for adaptive optics. Astron Astrophys 664:A71. https://doi.org/10.1051/0004-6361/202243311. arXiv:2205.07554 [astro-ph.IM]

November LJ, Simon GW (1988) Precise proper-motion measurement of solar granulation. Astrophys J 333:427. https://doi.org/10.1086/166758

Okoh DI, Seemala GK, Rabiu AB, Uwamahoro J, Habarulema JB, Aggarwal M (2018) A hybrid regression-neural network (HR-NN) method for forecasting the solar activity. Space Weather 16 (9):1424–1436. https://doi.org/10.1029/2018SW001907

Oord Avd, Dieleman S, Zen H, Simonyan K, Vinyals O, Graves A, Kalchbrenner N, Senior A, Kavukcuoglu K (2016) WaveNet: a generative model for raw audio. arXiv e-prints. arXiv:1609.03499

Osborne CMJ, Armstrong JA, Fletcher L (2019) RADYNVERSION: learning to invert a solar flare atmosphere with invertible neural networks. Astrophys J 873:128. https://doi.org/10.3847/1538-4357/ab07b. arXiv:1901.08626 [astro-ph.SR]

Panos B, Kleint L (2020) Real-time flare prediction based on distinctions between flaring and non-flaring active region spectra. Astrophys J 891(1):17. https://doi.org/10.3847/1538-4357/ab700b. arXiv:1911.12621 [astro-ph.SR]

Panos B, Kleint L (2021) Exploring mutual information between IRIS spectral lines. II. Calculating the most probable response in all spectral windows. Astrophys J 915(2):77. https://doi.org/10.3847/1538-4357/ac00c0. arXiv:2106.03463 [astro-ph.SR]

Panos B, Kleint L, Huwyler C, Krucker S, Melchior M, Ullmann D, Voloshynovskiy S (2018) Identifying typical Mg II flare spectra using machine learning. Astrophys J 861(1):62. https://doi.org/10.3847/1538-4357/aac779. arXiv:1805.10494 [astro-ph.SR]

Panos B, Kleint L, Voloshynovskiy S (2021) Exploring mutual information between IRIS spectral lines. I. Correlations between spectral lines during solar flares and within the quiet sun. Astrophys J 912 (2):121. https://doi.org/10.3847/1538-4357/abf11b. arXiv:2104.12161 [astro-ph.SR]

Panos B, Kleint L, Zbinden J (2023) Identifying preflare spectral features using explainable artificial intelligence. Astron Astrophys 671:A73. https://doi.org/10.1051/0004-6361/202244835. arXiv:2301.01560 [astro-ph.SR]

Park E, Moon YJ, Lee JY, Kim RS, Lee H, Lim D, Shin G, Kim T (2019) Generation of solar UV and EUV images from SDO/HMI magnetograms by deep learning. Astrophys J Lett 884(1):L23. https://doi.org/10.3847/2041-8213/ab46bb

Park E, Moon YJ, Lim D, Lee H (2020) De-noising SDO/HMI solar magnetograms by image translation method based on deep learning. Astrophys J Lett 891(1):L4. https://doi.org/10.3847/2041-8213/ab74d2

Park SH, Leka KD, Kusano K, Andries J, Barnes G, Bingham S, Bloomfield DS, McCloskey AE, Delouille V, Falconer D, Gallagher PT, Georgoulis MK, Kubo Y, Lee K, Lee S, Lobzin V, Mun J, Murray SA, Nageem TAMH, Qahwaji R, Sharpe M, Steenburgh RA, Steward G, Terkildsen M (2020) A comparison of flare forecasting methods. IV. Evaluating consecutive-day forecasting patterns. Astrophys J 890(2):124. https://doi.org/10.3847/1538-4357/ab65f0

Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, Desmaison A, Kopf A, Yang E, DeVito Z, Raison M, Tejani A, Chilamkurthy S, Steiner B, Fang L, Bai J, Chintala S (2019) Pytorch: an imperative style, high-performance deep learning library. In: Wallach H, Larochelle H, Beygelzimer A, d'Alché Buc F, Fox E, Garnett R (eds) Advances in Neural Information Processing Systems 32. Curran Associates, Inc., pp 8026–8037. http://papers.nips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

Paxman RG, Schulz TJ, Fienup JR (1992) Joint estimation of object and aberrations by using phase diversity. J Opt Soc Am A 9:1072–1085

Pesnell WD, Thompson BJ, Chamberlin PC (2012) The Solar Dynamics Observatory (SDO). Sol Phys 275:3–15. https://doi.org/10.1007/s11207-011-9841-3

Peyrard C, Mamalet F, Garcia C (2015) A comparison between multi-layer perceptrons and convolutional neural networks for text image super-resolution. In: Braz J, Battiato S, Imai FH (eds) VISAPP (1). SciTePress, pp 84–91

Prasad A, Roy S, Sarkar A, Chandra Panja S, Narayan Patra S (2022) Prediction of solar cycle 25 using deep learning based long short-term memory forecasting technique. Adv Space Res 69(1):798–813. https://doi.org/10.1016/j.asr.2021.10.047

Press WH, Teukolsky SA, Vetterling WT, Flannery BP (1986) Numerical recipes: the art of scientific computing. Cambridge University Press, Cambridge

Priest ER, Forbes TG (2002) The magnetic nature of solar flares. Astron Astrophys Rev 10(4):313–377. https://doi.org/10.1007/s001590100013

Querfeld CW, Smartt RN, Bommier V, Landi Degl'Innocenti E, House LL (1985) Vector magnetic fields in prominences: part two He I D3 Stokes profiles analysis for two quiescent prominences. Sol Phys 96(2):277–292. https://doi.org/10.1007/BF00149684

Quintero Noda C, Asensio Ramos A, Orozco Suárez D, Ruiz Cobo B (2015) Spatial deconvolution of spectropolarimetric data: an application to quiet Sun magnetic elements. Astron Astrophys 579:A3. https://doi.org/10.1051/0004-6361/201425414. arXiv:1505.03219 [astro-ph.SR]

Quintero Noda C, Shimizu T, Ruiz Cobo B, Suematsu Y, Katsukawa Y, Ichimoto K (2016) Analysis of a spatially deconvolved solar pore. Mon Not R Astron Soc 460(2):1476–1485. https://doi.org/10.1093/mnras/stw1068. arXiv:1605.01796 [astro-ph.SR]

Quintero Noda C, Suematsu Y, Ruiz Cobo B, Shimizu T, Asensio Ramos A (2016) Analysis of spatially deconvolved polar faculae. Mon Not R Astron Soc 460(1):956–965. https://doi.org/10.1093/mnras/stw1050. arXiv:1605.00330 [astro-ph.SR]

Rahaman N, Baratin A, Arpit D, Draxler F, Lin M, Hamprecht F, Bengio Y, Courville A (2019) On the spectral bias of neural networks. In: Chaudhuri K, Salakhutdinov R (eds) Proceedings of the 36th international conference on machine learning. Proceedings of machine learning research, vol 97. PMLR, pp 5301–5310. https://proceedings.mlr.press/v97/rahaman19a.html

Rahman S, Shin S, Hj Jeong, Siddique A, Moon YJ, Park E, Kang J, Bae SH (2023) Fast reconstruction of 3D density distribution around the sun based on the MAS by deep learning. Astrophys J 948(1):21. https://doi.org/10.3847/1538-4357/acbd3c

Rees DE, López Ariste A, Thatcher J, Semel M (2000) Fast inversion of spectral lines using principal component analysis. I. Fundamentals. Astron Astrophys 355:759–768

Reiss MA, Hofmeister SJ, De Visscher R, Temmer M, Veronig AM, Delouille V, Mampaey B, Ahammer H (2015) Improvements on coronal hole detection in SDO/AIA images using supervised classification. J Space Weather Space Clim 5:A23. https://doi.org/10.1051/swsc/2015025. arXiv:1506.06623 [astro-ph.SR]

Reiss MA, Muglach K, Möstl C, Arge CN, Bailey R, Delouille V, Garton TM, Hamada A, Hofmeister S, Illarionov E, Jarolim R, Kirk MSF, Kosovichev A, Krista L, Lee S, Lowder C, MacNeice PJ, Veronig

A, Cospar Iswat Coronal Hole Boundary Working Team (2021) The observational uncertainty of coronal hole boundaries in automated detection schemes. Astrophys J 913(1):28. https://doi.org/10.3847/1538-4357/abf2c8. arXiv:2103.14403 [astro-ph.SR]

Riley P, Lionello R, Linker JA, Cliver E, Balogh A, Beer J, Charbonneau P, Crooker N, DeRosa M, Lockwood M, Owens M, McCracken K, Usoskin I, Koutchmy S (2015) Inferring the structure of the solar corona and inner heliosphere during the Maunder minimum using global thermodynamic magnetohydrodynamic simulations. Astrophys J 802(2):105. https://doi.org/10.1088/0004-637X/802/2/105

Romberg J (2008) Imaging via compressive sampling. IEEE Signal Process Mag 25:14

Ronneberger O, Fischer P, Brox T (2015) U-Net: convolutional networks for biomedical image segmentation. arXiv e-prints arXiv:1505.04597 [cs.CV]

Rosenblatt F (1958) The perceptron: a probabilistic model for information storage and organization in the brain. Psychol Rev, pp 65–386

Roweis S, Saul LK (2000) Nonlinear dimensionality reduction by locally linear embedding. Science 290:2323. https://doi.org/10.1126/science.290.5500.2323

Ruiz Cobo B, Asensio Ramos A (2013) Returning magnetic flux in sunspot penumbrae. Astron Astrophys 549:L4. https://doi.org/10.1051/0004-6361/201220373. arXiv:1211.6335 [astro-ph.SR]

Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. Nature 323(6088):533–536. https://doi.org/10.1038/323533a0

Rumelhart DE, Hinton GE, Williams RJ (1988) Learning representations by back-propagating errors. In: Anderson JA, Rosenfeld E (eds) Neurocomputing: foundations of research. MIT Press, Cambridge, pp 696–699

Russell S, Norvig P (2009) Artificial intelligence: a modern approach, 3rd edn. Prentice Hall, Hoboken

Sadykov VM, Kitiashvili IN, Dalda AS, Oria V, Kosovichev AG, Illarionov E (2021) Compression of solar spectroscopic observations: a case study of Mg II k spectral line profiles observed by NASA's IRIS Satellite. In: 18th international conference on content-based multimedia indexing, CBMI 2021, Lille, France, June 28-30, 2021. IEEE, pp 1–6. https://doi.org/10.1109/CBMI50038.2021.9461879

Sainz Dalda A, de la Cruz Rodríguez J, De Pontieu B, Gošić M (2019) Recovering thermodynamics from spectral profiles observed by IRIS: a machine and deep learning approach. Astrophys J 875(2):L18. https://doi.org/10.3847/2041-8213/ab15d9. arXiv:1904.08390 [astro-ph.SR]

Salvatelli V, Bose S, Neuberg B, dos Santos LFG, Cheung M, Janvier M, Gunes Baydin A, Gal Y, Jin M (2019) Using U-nets to create high-fidelity virtual observations of the solar corona. arXiv e-prints arXiv:1911.04006 [astro-ph.SR]

Salvatelli V, dos Santos LFG, Bose S, Neuberg B, Cheung MCM, Janvier M, Jin M, Gal Y, Güneş Baydin A (2022) Exploring the limits of synthetic creation of solar EUV images via image-to-image translation. Astrophys J 937(2):100. https://doi.org/10.3847/1538-4357/ac867b. arXiv:2208.09512 [astro-ph.SR]

Scherrer PH, Bogart RS, Bush RI, Hoeksema JT, Kosovichev AG, Schou J, Rosenberg W, Springer L, Tarbell TD, Title A, Wolfson CJ, Zayer I, MDI Engineering Team (1995) The Solar Oscillations Investigation (SOI) uses the Michelson Doppler Imager (MDI). Sol Phys 162(1–2):129–188. https://doi.org/10.1007/BF00733429

Schmidhuber J (2014) Deep learning in neural networks: an overview. ArXiv e-prints arXiv:1404.7828

Schölkopf B, Smola AJ, Müller KR (1998) Nonlinear component analysis as a kernel eigenvalue problem. Neural Comput 10:1299. https://doi.org/10.1162/089976698300017467

Schwarz GE (1978) Estimating the dimension of a model. Ann Stat 6:461. https://doi.org/10.1214/aos/1176344136

Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D (2017) Grad-CAM: visual explanations from deep networks via gradient-based localization. In: 2017 IEEE international conference on computer vision (ICCV). pp 618–626. https://doi.org/10.1109/ICCV.2017.74

Semel M (1981) Magnetic fields observed in a sunspot and faculae using 12 lines simultaneously. Astron Astrophys 97(1):75–78

Sheeley JNR, Howard RA, Koomen MJ, Michels DJ (1983) Associations between coronal mass ejections and soft X-ray events. Astrophys J 272:349–354. https://doi.org/10.1086/161298

Shi X, Chen Z, Wang H, Yeung DY, Wong Wk, WOO Wc (2015) Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In: Cortes C, Lawrence N, Lee D, Sugiyama M, Garnett R (eds) Advances in neural information processing systems, vol 28. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2015/file/07563a3fe3bbe7e3ba84431ad9d055af-Paper.pdf

Shibata K, Magara T (2011) Solar flares: magnetohydrodynamic processes. Living Rev Sol Phys 8:6. https://doi.org/10.12942/lrsp-2011-6

Shin G, Moon YJ, Park E, Jeong H, Lee H, Bae SH (2020) Generation of high-resolution solar pseudo-magnetograms from Ca II K images by deep learning. Astrophys J Lett 895(1):L16. https://doi.org/10.3847/2041-8213/ab9085

Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. ArXiv e-prints arXiv:1409.1556 [cs.CV]

Sitzmann V, Martel J, Bergman A, Lindell D, Wetzstein G (2020) Implicit neural representations with periodic activation functions. Adv Neural Inform Process Syst 33:7462–7473

Skumanich A, López Ariste A (2002) The physical content of the leading orders of principal component analysis of spectral profiles. Astrophys J 570(1):379–386. https://doi.org/10.1086/339503

Socas-Navarro H (2003) Measuring solar magnetic fields with artificial neural networks. Neural Netw 16:355

Socas-Navarro H (2004) A simple procedure for optimizing the height resolution in spectral line inversions. Astrophys J 614:457

Socas-Navarro H (2005) Feature extraction techniques for the analysis of spectral polarization profiles. Astrophys J 620(1):517–522. https://doi.org/10.1086/426811. arXiv:astro-ph/0410565 [astro-ph]

Socas-Navarro H (2005) Strategies for spectral profile inversion using artificial neural networks. Astrophys J 621:545–553. https://doi.org/10.1086/427431. arXiv:astro-ph/0410567 [astro-ph]

Socas-Navarro H, de la Cruz Rodríguez J, Asensio Ramos A, Trujillo Bueno J, Ruiz Cobo B (2015) An open-source, massively parallel code for non-LTE synthesis and inversion of spectral lines and Zeeman-induced Stokes profiles. Astron Astrophys 577:A7. https://doi.org/10.1051/0004-6361/201424860. arXiv:1408.6101 [astro-ph.SR]

Solanki SK, del Toro Iniesta JC, Woch J, Gandorfer A, Hirzberger J, Alvarez-Herrero A, Appourchaux T, Martínez Pillet V, Pérez-Grande I, Sanchis Kilders E, Schmidt W, Gómez Cama JM, Michalik H, Deutsch W, Fernandez-Rico G, Grauf B, Gizon L, Heerlein K, Kolleck M, Lagg A, Meller R, Müller R, Schühle U, Staub J, Albert K, Alvarez Copano M, Beckmann U, Bischoff J, Busse D, Enge R, Frahm S, Germerott D, Guerrero L, Löptien B, Meierdierks T, Oberdorfer D, Papagiannaki I, Ramanath S, Schou J, Werner S, Yang D, Zerr A, Bergmann M, Bochmann J, Heinrichs J, Meyer S, Monecke M, Müller MF, Sperling M, Álvarez García D, Aparicio B, Balaguer Jiménez M, Bellot Rubio LR, Cobos Carracosa JP, Girela F, Hernández Expósito D, Herranz M, Labrousse P, López Jiménez A, Orozco Suárez D, Ramos JL, Barandiarán J, Bastide L, Campuzano C, Cebollero M, Dávila B, Fernández-Medina A, García Parejo P, Garranzo-García D, Laguna H, Martín JA, Navarro R, Núñez Peral A, Royo M, Sánchez A, Silva-López M, Vera I, Villanueva J, Fourmond JJ, de Galarreta CR, Bouzit M, Hervier V, Le Clec'h JC, Szwec N, Chaigneau M, Buttice V, Dominguez-Tagle C, Philippon A, Boumier P, Le Cocguen R, Baranjuk G, Bell A, Berkefeld T, Baumgartner J, Heidecke F, Maue T, Nakai E, Scheiffelen T, Sigwarth M, Soltau D, Volkmer R, Blanco Rodríguez J, Domingo V, Ferreres Sabater A, Gasent Blesa JL, Rodríguez Martínez P, Osorno Caudel D, Bosch J, Casas A, Carmona M, Herms A, Roma D, Alonso G, Gómez-Sanjuan A, Piqueras J, Torralbo I, Fiethe B, Guan Y, Lange T, Michel H, Bonet JA, Fahmy S, Müller D, Zouganelis I (2020) The Polarimetric and Helioseismic Imager on Solar Orbiter. Astron Astrophys 642:A11. https://doi.org/10.1051/0004-6361/201935325. arXiv:1903.11061 [astro-ph.IM]

Su Y, Veronig AM, Hannah IG, Cheung MCM, Dennis BR, Holman GD, Gan W, Li Y (2018) Determination of differential emission measure from solar extreme ultraviolet images. Astrophys J 856(1):L17. https://doi.org/10.3847/2041-8213/aab436

Sutton R, Barto A (1998) Reinforcement learning. MIT Press, Cambridge

SWPC NOAA (2014) Forecast verification glossary. https://www.swpc.noaa.gov/content/forecast-verification

Szenicer A, Fouhey DF, Muñoz-Jaramillo A, Wright PJ, Thomas R, Galvez R, Jin M, Cheung MCM (2019) A deep learning virtual instrument for monitoring extreme UV solar spectral irradiance. Sci Adv 5(10):eaaw6548. https://doi.org/10.1126/sciadv.aaw6548

Tancik M, Srinivasan P, Mildenhall B, Fridovich-Keil S, Raghavan N, Singhal U, Ramamoorthi R, Barron J, Ng R (2020) Fourier features let networks learn high frequency functions in low dimensional domains. Adv Neural Inform Proc Syst 33:7537–7547

Teh YW, Jordan MI (2010) Hierarchical bayesian nonparametric models with applications. In: Hjort NL, Holmes C, Müller P, Walker SG (eds) Bayesian nonparametrics. Cambridge series in statistical and probabilistic mathematics. Cambridge University Press, p 158-207. https://doi.org/10.1017/CBO9780511802478.006

Tenenbaum JB, de Silva V, Langford JC (2000) A global geometric framework for nonlinear dimensionality reduction. Science 290:2319. https://doi.org/10.1126/science.290.5500.2319

Tieleman T, Hinton G (2012) Lecture 6.5-rmsprop: divide the gradient by a running average of its recent magnitude. In: COURSERA: neural networks for machine learning. https://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf

Tipping ME (2000) The Relevance Vector Machine. In: Solla, Leen TK, Müller KR (eds) Advances in neural information processing systems vol 12. p 652

Torres J, Zhao L, Chan PK, Zhang M (2022) A machine learning approach to predicting SEP events using properties of coronal mass ejections. Space Weather 20(7):e2021SW002797. https://doi.org/10.1029/2021SW002797

Tremblay B, Attie R (2020) Inferring plasma flows at granular and supergranular scales with a new architecture for the deepvel neural network. Frontiers Astron Space Sci 7:25. https://doi.org/10.3389/fspas.2020.00025

Tremblay B, Roudier T, Rieutord M, Vincent A (2018) Reconstruction of horizontal plasma motions at the photosphere from intensitygrams: a comparison between DeepVel, LCT, FLCT, and CST. Sol Phys 293(4):57

Tsurutani BT, Gonzalez WD, Gonzalez ALC, Guarnieri FL, Gopalswamy N, Grande M, Kamide Y, Kasahara Y, Lu G, Mann I, McPherron R, Soraas F, Vasyliunas V (2006) Corotating solar wind streams and recurrent geomagnetic activity: a review. J Geophys Res 111(A7):A07S01. https://doi.org/10.1029/2005JA011273

Upendran V, Cheung MCM, Hanasoge S, Krishnamurthi G (2020) Solar wind prediction using deep learning. Space Weather 18(9):e02478. https://doi.org/10.1029/2020SW002478. arXiv:2006.05825 [astro-ph.SR]

Usoskin IG (2023) A history of solar activity over millennia. Living Rev Sol Phys 20:2. https://doi.org/10.1007/s41116-023-00036-z

van Noort M, Rouppe van der Voort L, Löfdahl MG (2005) Solar image restoration by use of multi-frame blind de-convolution with multiple objects and phase diversity. Sol Phys 228:191–215. https://doi.org/10.1007/s11207-005-5782-z

Vapnik VN (1995) The nature of statistical learning theory. Springer, New York. https://doi.org/10.1007/978-1-4757-3264-1

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser u, Polosukhin I (2017) Attention is all you need. In: Proceedings of the 31st international conference on neural information processing systems. NIPS'17. pp 6000–6010. arXiv:1706.03762 [cs.CL]

Ventura C, Bellver M, Girbau A, Salvador A, Marques F, Giro-i Nieto X (2019) RVOS: End-to-end recurrent network for video object segmentation. In: The IEEE conference on computer vision and pattern recognition (CVPR). arXiv:1903.05612 [cs.CV]

Verbeeck C, Delouille V, Mampaey B, De Visscher R (2014) The SPoCA-suite: software for extraction, characterization, and tracking of active regions and coronal holes on EUV images. Astron Astrophys 561:A29. https://doi.org/10.1051/0004-6361/201321243

Verma M, Matijevič G, Denker C, Diercke A, Dineva E, Balthasar H, Kamlah R, Kontogiannis I, Kuckein C, Pal PS (2021) Classification of high-resolution solar Hα spectra using t-distributed stochastic neighbor embedding. Astrophys J 907(1):54. https://doi.org/10.3847/1538-4357/abcd95. arXiv:2011.13214 [astro-ph.SR]

Vicente Arévalo A, Asensio Ramos A, Esteban Pozuelo S (2022) Accelerating non-LTE synthesis and inversions with graph networks. Astrophys J 928(2):101. https://doi.org/10.3847/1538-4357/ac53b3

Viticchié B, Sánchez Almeida J (2011) Asymmetries of the Stokes V profiles observed by HINODE SOT/SP in the quiet Sun. Astron Astrophys 530:A14. https://doi.org/10.1051/0004-6361/201016096. arXiv:1103.1987 [astro-ph.SR]

Vögler A, Shelyag S, Schüssler M, Cattaneo F, Emonet T, Linde T (2005) Simulations of magneto-convection in the solar photosphere. Equations, methods, and results of the MURaM code. Astron Astrophys 429:335–351. https://doi.org/10.1051/0004-6361:20041507

Wang N, Zhang Y, Zhang L (2021) Dynamic selection network for image inpainting. IEEE Trans Image Proc 30:1784–1798. https://doi.org/10.1109/TIP.2020.3048629

Wang QJ, Li JC, Guo LQ (2021) Solar cycle prediction using a long short-term memory deep learning model. Res Astron Astrophys 21(1):012. https://doi.org/10.1088/1674-4527/21/1/12

Wang S, Wang H, Perdikaris P (2021) On the eigenvector bias of Fourier feature networks: From regression to solving multi-scale PDEs with physics-informed neural networks. Comput Meth Appl Mech Eng 384:113938. https://doi.org/10.1016/j.cma.2021.113938

Wang TC, Liu MY, Zhu JY, Tao A, Kautz J, Catanzaro B (2017) High-resolution image synthesis and semantic manipulation with conditional GANs. arXiv e-prints arXiv:1711.11585 [cs.CV]

Winebarger AR, Weber M, Bethge C, Downs C, Golub L, DeLuca E, Savage S, del Zanna G, Samra J, Madsen C, Ashraf A, Carter C (2019) Unfolding overlapped slitless imaging spectrometer data for extended sources. Astrophys J 882(1):12. https://doi.org/10.3847/1538-4357/ab21db. arXiv:1811.08329 [astro-ph.SR]

Yi K, Moon YJ, Lim D, Park E, Lee H (2021) Visual explanation of a deep learning solar flare forecast model and its relationship to physical parameters. Astrophys J 910(1):8. https://doi.org/10.3847/1538-4357/abdebe

Yi K, Moon YJ, Jeong HJ (2023) Application of deep reinforcement learning to major solar flare forecasting. Astrophys J Suppl Ser 265(2):34. https://doi.org/10.3847/1538-4365/acb76d

Yu X, Xu L, Ren Z, Zhao D, Sun W (2022) Image desaturation for SDO/AIA using mixed convolution network. Res Astron Astrophys 22(6):065009. https://doi.org/10.1088/1674-4527/ac69b7