Classification of Solar Flares using Data Analysis and Clustering of Active Regions

Hanne Baeke[1], Jorge Amaya[2], and Giovanni Lapenta[3]

[1]KU Leuven
[2]ESA
[3]Univ Leuven, KU Leuven, Dept Wiskunde

January 10, 2023

Abstract

We devised a new data analysis technique to identify the threat level of solar active regions by processing a combined data set of magnetic field properties and flaring activity. The data set is composed of two elements: a reduced factorization of SHARP properties of the active regions, and information about the flaring activity at the time of measurement of the SHARP parameters. Machine learning is used to reduce the data and to subsequently classify the active regions. For this classification we used both supervised and unsupervised clustering. The following processing steps are applied to reduce and enhance the SHARP data: outlier detection, redundancy elimination with common factor analysis, addition of sparsity with autoencoders, and construction of a balanced data set with under- and over-sampling. Supervised clustering (based on K-nearest neighbors) produces very good results on the strong X- and M-flares, with TSS scores of respectively 93% and 75%. Unsupervised clustering (based on K-means and Gaussian Mixture Models) shows that non-flaring and flaring active regions can be distinguished, but there is not enough information in the data set for the technique to identify clear differences between the different flaring levels. This work shows that the SHARP database lacks information to accurately make flaring predictions: there is no clear hyperplane in the SHARP parameter space, even after a detailed cleaning procedure, that can separate active regions with different flaring activity. We propose instead, for future projects, to complement the magnetic field parameters with additional information, like images of the active regions.

Classification of Solar Flares using Data Analysis and Clustering of Active Regions

H. Baeke[1], J. Amaya[1], G. Lapenta[1]

[1]Center for mathematical Plasma Astrophysics, Department of Mathematics, KU Leuven, Celestijnenlaan 200B, 3001 Leuven, Belgium

Key Points:

- SHARP parameters of solar active regions contain redundant information that can be reduced to five parameters using Common Factor Analysis.
- Unsupervised classification allows to differentiate inactive regions, from C/M flaring active regions, and extremely active X-flare regions.
- We detect no clear boundaries in the reduced parameters between different levels of moderate flaring activity.

Corresponding author: Hanne Baeke, `hanne.baeke@kuleuven.be`

Abstract

We devised a new data analysis technique to identify the threat level of solar active regions by processing a combined data set of magnetic field properties and flaring activity. The data set is composed of two elements: a reduced factorization of SHARP properties of the active regions, and information about the flaring activity at the time of measurement of the SHARP parameters. Machine learning is used to reduce the data and to subsequently classify the active regions. For this classification we used both supervised and unsupervised clustering. The following processing steps are applied to reduce and enhance the SHARP data: outlier detection, redundancy elimination with common factor analysis, addition of sparsity with autoencoders, and construction of a balanced data set with under- and over-sampling. Supervised clustering (based on K-nearest neighbors) produces very good results on the strong X- and M-flares, with TSS scores of respectively 0.93 and 0.75. Unsupervised clustering (based on K-means and Gaussian Mixture Models) shows that non-flaring and flaring active regions can be distinguished, but there is not enough information in the data set for the technique to identify clear differences between the different flaring levels. This work shows that the SHARP database lacks information to accurately make flaring predictions: there is no clear hyperplane in the SHARP parameter space, even after a detailed cleaning procedure, that can separate active regions with different flaring activity. We propose instead, for future projects, to complement the magnetic field parameters with additional information, like images of the active regions.

Plain Language Summary

One of the main sources of space weather activity are solar active regions. In these zones the magnetic activity of the Sun is increased and can produce the two most energetic events in the solar system: flares and coronal mass ejections. We investigate the magnetic field properties of active regions, and the amount of energy they release. Our end goal is to produce an automatic model that can forecast the energy level released by a flare from solar active regions, using only their current magnetic field properties.

For this study, we used machine learning techniques that recognize patterns in data, without being explicitly told what to look for. These techniques can sometimes find patterns that escape the human intuition. The technique classifies different active regions, based on their magnetic properties, identifying those that can release large amounts of energy in the near future.

Our technique is able to discover differences between flaring and non-flaring active regions. But the data contains not enough information to predict how strong the energy releases will be. Therefore, improvement is still needed since we want to identify the strongest, most dangerous energy releases. Future research should incorporate other data types to get better results.

1 Introduction

Solar flares pose a serious threat to the near-Earth environment. They can produce streams of highly energetic particles, which can affect the Earth's magnetosphere within a few hours or minutes (Cinto et al., 2020). These particles pose radiation hazards to astronauts and spacecrafts (Mikaelian, 2009). Flares are also associated with radio communication disruptions (Knipp et al., 2016; Redmon et al., 2018), and the associated high energy particles can ionize our atmosphere at low altitudes (Liu et al., 2021). The largest flares are often accompanied by coronal mass ejections (CMEs). Kawabata et al. (2018) show that CMEs are associated with approximately all events whose X-ray flux is larger than $10^{-3.9} W m^{-2}$, which correspond to the X-flares. These CMEs can trigger geomagnetic storms, which can disable satellites (Dang et al., 2022) and even knock out electrical power grids (Pulkkinen et al., 2005). Should such a large storm happen nowadays, it would have catastrophic results, causing considerable economic damage. For example, the 1977 New York City blackout cost is estimated at \$624 million dollars (Sorkin, 1982). A similar event today would have an even higher cost. Forecasting solar energetic activity is a critical topic in space weather research.

The differentiation of solar active regions very often involves the use of sunspot classifications - Mount Wilson (Hale et al., 1919) and McIntosh (McIntosh, 1990) - which are still performed manually. These classes are based on human observations in the visible light spectrum. This leads to inference of the subjectivity of the experts. Moreover, the visible light spectrum provides very limited information regarding the critical properties of solar active regions. Today it is possible to automatize the classification of solar active regions, reducing the influence of human bias. This will allow to produce fast solar flare forecasting systems.

This work focuses on the development of an unsupervised classification of solar active regions, using machine learning, and on their relation to their (non-)flaring activity. The classification is based on the SHARP parameters, extracted from SDO HMI observations of the magnetic field of active regions. A detailed processing of the SHARP data is performed to achieve the best possible results from unsupervised classification techniques. Therefore, these processing steps are also discussed with care throughout this paper.

There have been multiple previous attempts to build an automated classification of active regions. However, most of these studies tried to automate the existing McIntosh or Mount-Wilson classifications, e.g. (Colak & Qahwaji, 2008; Maloney & Gallagher, 2018; Nguyen et al., 2006; Smith et al., 2018). These studies applied machine learning on solar images, often combined with automatic sunspot detection. The machine learning methods used in the literature include neural networks, k-nearest neighbors, Support Vector Machines (SVMs), Random Forest and layered learning. In most cases, the percentage of correct classifications depends strongly on the specific class and on the amount of data available. The results of Colak and Qahwaji (2008) for example show results with a percentage of correct classifications between $\sim 40\%$ and $\sim 85\%$.

Housseal et al. (2019) performed unsupervised classification of sunspots, however, the authors did not use the magnetic field parameters: they used instead HMI magnetogram images to look for patterns in the sunspots connected to the active regions.

Recently, multiple papers have used the SHARP magnetic field parameters to construct solar flare prediction algorithms based on machine learning, e.g. (Abduallah et al., 2020; Bobra & Couvidat, 2015; Chen et al., 2019; Ilonidis et al., 2015; Jiao et al., 2020; Jonas et al., 2018; Liu et al., 2017; Ran et al., 2022; Sinha et al., 2022; Sun et al., 2022; Wang et al., 2020; Zhang et al., 2022). The methods used include Random Forest, MLPs, extreme learning machines, LSTMs, CNNs, SVMs, etc. Ilonidis et al. (2015) used time series of the SDO magnetic field data and constructed SVMs to forecast solar flares, which yielded a True Skill Score of 91%. Bobra and Couvidat (2015) also used SVMs on SHARP data, to distinguish between flare producing active regions and non-flare producing active regions. The authors did not include C-flares, which simplified the distinction between flaring and non-flaring active regions. Sun et al. (2022) focused on the prediction of M- and X-flares versus flare-quiet instances. They discarded all C-flares and lower from their data set. Jiao et al. (2020) took a different approach and applied machine learning on the SHARP parameters to identify the flare intensity, a continuous variable, instead of the discrete solar flare types.

A number of studies have investigated the importance of each of the SHARP parameters for solar flare prediction (Ran et al., 2022; Sinha et al., 2022; Zhang et al., 2022). They found that the most influential SHARP parameters are `TOTUSJH`, `TOTUSJZ`, `MEANPOT`, `TOTPOT`, `USFLUX` and `R_VALUE`. See Table 1 for the physical meaning of these parameters.

A new data set has been created by Bobra et al. (2021), called SMARPs. These are similar to SHARPs, but constructed from the solar images taken by MDI of SOHO. It attempts to extend backwards the SHARP database to the more active Solar Cycle 23. However, the SMARPs do not include as much information as the SHARPs and the data quality is lower (Sun et al., 2022).

Some studies combined the SHARP magnetic field parameters with features that are automatically generated from the solar images with machine learning methods, e.g (Chen et al., 2019; Jonas et al., 2018). Chen et al. (2019) compared the results of LSTM models trained on the SHARP data and on autoencoder-derived features and found that they were very similar. Therefore, the autoencoder-derived features could be a viable alternative for the SHARP parameters.

The goal of the present work is to classify the flaring activity of solar active regions, based only on the SHARP parameters extracted from the SDO HMI instrument. We apply rigorous and comprehensive pre-processing techniques to extract as much useful information as possible from the SHARP database. The results will inform us if there is enough information in the data to perform flare forecasts. While many of the classification methods used in the literature are based on supervised learning, we use unsupervised clustering to allow the computer to extract patterns unknown to the human experts. We show how the unsupervised classes that we obtain correlate with the flaring activity of active regions. In this work we also try to distinguish the different levels of flaring activity, whereas most studies are limited to the prediction of binary classes, only finding differences between flaring and non-flaring data.

The paper is structured as follows. Active regions and solar flares are briefly introduced in section 2. Section 3 discusses the data used, followed by section 4, which explains the data processing methods and results. Sections 5 and 6 introduce the clustering methods and types of evaluation. The clustering results are shown in section 7, followed by the discussion in section 8. Finally, section 9 summarizes the main conclusions of the research results.

2 Active Regions and Solar Flares

Solar active regions are large areas on the Sun where the magnetic activity temporarily and locally increases. The magnetic field there is complex and intense. Magnetic fields in active regions can be a thousand times stronger than the average solar magnetic field of a few Gauss (Sheeley, N.R., 2020). The number of active regions observed in the solar disk varies over the course of the solar cycle and are most common during its peak.

A solar flare is a sudden, intense brightening of a small area on the Sun, lasting minutes to a few hours. Flares occur in the solar corona when magnetic field lines of opposite polarity are forced together, by the convective motion of their foot-points in the convection zone, or by travelling coronal pressure waves. This causes magnetic reconnection, a sudden transformation of magnetic energy into kinetic and thermal energy. Streams of highly energetic particles travel along magnetic field lines, generating high intensity electromagnetic radiation on their path and during their interaction with matter. Solar flares typically erupt from solar active regions, because their complex and intense magnetic field is the perfect locus of magnetic reconnection (Priest & Forbes, 2002).

Flares are classified according to the strength of their soft X-ray emission, as recorded by the GOES satellites located in geostationary orbit. The following is a list of the flare classes in order of exponentially increasing magnitude: A, B, C, M and X. Strong solar flares occur very infrequently, compared to weak solar flares. Therefore, solar flare data is by definition largely imbalanced. This always has to be taken into account during the processing of the data and the interpretation of the results.

3 Data Set

The open source data set of Angryk et al. (2020b) is used for this research. The authors developed a data set (henceforth called the Angryk data set), extracted from the Space Weather HMI Active Region Patch series (SHARP) (Bobra et al., 2011), integrated with information from solar flare catalogs. These SHARP patches and their magnetic field parameters are derived from solar photospheric vector magnetograms obtained by the Helioseismic and Magnetic Imager (HMI) from the Solar Dynamics Observatory (SDO). The HMI instrument provides information on the magnetic field in the solar photosphere. These observations are bundled in patches for each active region. Magnetic field parameters are extracted from these patches and integrated over the whole area. They give an indication of the magnetic activity of the complete patch.

The Angryk data set contains sixteen SHARP parameters and eight additional parameters proposed by Angryk et al. (2020a). These 24 parameters are listed in Table 1. The data set also contains parameters `BFLARE`, `CFLARE`, `MFLARE` and `XLFARE`. These express the number of flares of each flare class occurring at the time of measurement of the SHARP and therefore indicate the concurrent solar flare activity of that active region. For simplicity, in this work, each data point has been assigned to only one of four classes:

No-flare, C-flare, M-flare or X-flare. These correspond to the strongest occurring flare originating from the active region at that time. The No-flare class signifies the flare-quiet instances, but also the weakest, A- and B-class, flares. This because the A- and B-flares are hard to distinguish against the background brightness of the Sun (Chen et al., 2019). The assignment of flare types to the data points leads to the following ratio: 2 602 509 No-flares, 6717 C-flares, 680 M-flares and 47 X-flares. The data was collected between May 2010 and December 2018. This corresponds with solar cycle 24 (December 2008 - December 2019) and includes the solar maximum in April 2014. This solar cycle was an unusual quiet one, and the data set contains only few strong flares. The Angryk data set is meant to serve as a benchmark data set for testing flare prediction algorithms (Angryk et al., 2020a).

4 Data Processing

Some pre-processing of the data set was already carried out by Angryk et al. (2020a). Further processing includes outlier removal, data transformation and dimensionality reduction. These steps are explained in more detail in the following sections.

There is a large class imbalance present in the data set, with 2 602 509 No-flares, 6717 C-flares, 680 M-flares and only 47 X-flares. This class imbalance needs to be taken into account when processing the data. To reduce the impact of class imbalance, in this work the No-flare class is randomly under-sampled to 50 000 No-flares. This is done by randomly selecting 50 000 data points from the 2 602 509 No-flares, without selecting the same data point twice.

The selected number of No-flares is determined after multiple tests of the autoencoding procedure, described in section 4.3.2, the most data-intensive processing step in this work. In short, in an autoencoder a compression and decompression of the data set is performed, and the active region properties before and after the procedure should be exactly the same. We applied the procedure with different sample sizes. For each case the error is computed. When the sample size is too small, the error is large. Increasing the size of the sample reduces the error. A plot of the sample size versus the error presents an optimal inflection point, which in this work corresponds to the selected sample size: 50 000 data points are sufficient to obtain an accuracy comparable to the full 2 602 509 data points.

In section 4.4 we show how we handle additional class imbalances using over- and under-sampling techniques.

Table 1: Magnetic field parameters from Angryk et al. (2020b). Parameters with * are derived by Angryk et al. (2020a), the others are contained in SHARP. Units from Liu et al. (2017) and SDO.

Parameters	**Description**	**Formula**
`ABSNJZH` [$10\text{G}^2/\text{m}$]	Absolute net current helicity	$H_{c_{abs}} \propto \lvert \sum B_z \cdot J_z \rvert$
`EPSX`* [-10^{-1}]	Sum normalized Lorentz force (X)	$\delta F_x \propto \frac{\sum B_x B_z}{\sum B^2}$
`EPSY`* [-10^{-1}]	Sum normalized Lorentz force (Y)	$\delta F_y \propto \frac{-\sum B_y B_z}{\sum B^2}$
`EPSZ`* [-10^{-1}]	Sum normalized Lorentz force (Z)	$\delta F_z \propto \frac{\sum (B_x^2 + B_y^2 - B_z^2)}{\sum B^2}$
`MEANALP` [1/Mm]	Mean twist parameter	$\alpha_{total} \propto \frac{\sum J_z \cdot B_z}{\sum B_z^2}$
`MEANGAM` [°]	Mean inclination angle	$\overline{\gamma} = \frac{1}{N} \sum \arctan\left(\frac{B_h}{B_z}\right)$
`MEANGBH` [G/Mm]	Mean horizontal field gradient	$\overline{\nabla B_h} = \frac{1}{N} \sum \sqrt{\left(\frac{\partial B_h}{\partial x} + \frac{\partial B_h}{\partial y}\right)}$
`MEANGBT` [G/Mm]	Mean total field gradient	$\overline{\nabla B_{tot}} = \frac{1}{N} \sum \sqrt{\left(\frac{\partial B}{\partial x} + \frac{\partial B}{\partial y}\right)}$
`MEANGBZ` [G/Mm]	Mean vertical field gradient	$\overline{\nabla B_z} = \frac{1}{N} \sum \sqrt{\left(\frac{\partial B_z}{\partial x} + \frac{\partial B_z}{\partial y}\right)}$
`MEANJZD` [mA/m^2]	Mean vertical current density	$\overline{J_z} \propto \frac{1}{N} \sum \left(\frac{\partial B_y}{\partial x} - \frac{\partial B_x}{\partial y}\right)$
`MEANJZH` [G^2/m]	Mean current helicity	$\overline{H_c} \propto \frac{1}{N} \sum B_z \cdot J_z$
`MEANPOT` [$10^3\text{ergs}/\text{cm}^3$]	Mean photospheric excess magnetic energy density	$\overline{\rho} \propto \frac{1}{N} \sum (\mathbf{B}^{Obs} - \mathbf{B}^{Pot})^2$
`MEANSHR` [°]	Mean shear angle	$\overline{\Gamma} = \frac{1}{N} \sum \arccos\left(\frac{\mathbf{B}^{Obs} \cdot \mathbf{B}^{Pot}}{\lvert B^{Obs}\rvert \lvert B^{Pot}\rvert}\right)$
`R_VALUE`* [Mx]	Total unsigned flux around high gradient polarity inversion lines	$\phi = \sum \lvert B_{los}\rvert \cdot dA$ (within R mask)
`SAVNCPP` [10^{12}A]	Summed absolute value of net current per polarity	$J_{\Sigma z} \propto \left\lvert \sum^{B_z^+} J_z dA \right\rvert + \left\lvert \sum^{B_z^-} J_z dA \right\rvert$
`SHRGT45` [%]	Area with shear angle > 45°	$\frac{\text{Area with Shear} > 45°}{\text{Total Area}}$
`TOTBSQ`* [10^{10}G^2]	Total magnitude of Lorentz force	$F \propto \sum B^2$
`TOTFX`* [-10^{23}dyne]	Sum X-component of Lorentz force	$F_x \propto \sum B_x B_z dA$
`TOTFY`* [-10^{23}dyne]	Sum Y-component of Lorentz force	$F_y \propto \sum B_y B_z dA$
`TOTFZ`* [-10^{23}dyne]	Sum Z-component of Lorentz force	$F_z \propto \sum \left(B_x^2 + B_y^2 - B_z^2\right) dA$
`TOTPOT` [$10^{23}\text{ergs}/\text{cm}^3$]	Total photospheric magnetic energy density	$\rho_{tot} \propto \sum \left(\overrightarrow{\mathbf{B}^{Obs}} - \overrightarrow{\mathbf{B}^{Pot}}\right)^2 dA$
`TOTUSJH` [$10^2\text{G}^2/\text{m}$]	Total unsigned current helicity	$H_{c_{total}} \propto \sum B_z \cdot J_z$
`TOTUSJZ` [10^{12}A]	Total unsigned vertical current	$J_{z_{total}} = \sum \lvert J_z\rvert dA$
`USFLUX` [10^{21}Mx]	Total unsigned flux	$\phi = \sum \lvert B_z\rvert dA$

4.1 Outlier Removal

Multiple entries in the data set contain one or more empty properties (NaN values). We eliminate from the original data set every entry where at least one of the properties was empty. We also perform a detection and elimination of outliers. These were identified using the hierarchical clustering algorithm HDBSCAN. This method is able to automatically choose the optimal clustering of a cloud of points in an N-dimensional space. The points that are detached from the core cloud of points are identified as outliers. A more detailed explanation of HDBSCAN can be found in Campello et al. (2013).

With this technique 586 outliers were found. About 20% of the outliers come from HMI magnetogram images taken during rotation or re-positioning of the SDO spacecraft, causing distortions in the data.

In addition, 36 outliers were identified and removed by hand. Thirty-three of these additional outliers were due to the same parameter, `MEANPOT`. The other three were due to the parameter `TOTFZ`. The fact that they were missed by HDBSCAN is probably due to a combination of the standardization and some extreme outliers. The standardization transforms the data to zero mean and to unit variance. If there are a few extreme outliers, this will shift the majority of the data to very small values. Because this is not the case for the other parameters, there is a difference of $\sim 2-3$ orders of magnitude, which hinders HDBSCAN to detect all outliers.

4.2 Data Transformation

To be able to differentiate groups of points in the parameter space, it is necessary to identify high concentrations of points that can be separated by a hyper-plane. An initial visual inspection of the distribution function of each one of the parameters can show if there are peaks and valleys in the distribution that clearly separate active regions with different properties. Some of the parameters have a very small spread of values among all the active regions. Unsupervised clustering techniques have difficulties identifying multiple clusters in unimodal distributed parameters, since this would only lead to one cluster. We applied transformations to some of the parameters to perform a rebinning of the data distributions. This is one of the procedures known in machine learning as 'feature engineering'. The transformations used are listed in Table 2.

Figure 1 shows the difference a good transformation can make, and how this can improve clustering. After a logarithmic transformation two peaks are visible, while before there is only one very large one.

Table 2: Data transformations used to expand some very narrow distributions.

Parameter (Table 1)	**Transformation**
TOTUSJH	$\ln(x + \lvert\min(x)\rvert + 0.01)$
TOTBSQ	$\ln(x + \lvert\min(x)\rvert + 0.01)$
TOTPOT	$\ln(x + \lvert\min(x)\rvert + 0.01)$
TOTUSJZ	$\ln(x + \lvert\min(x)\rvert + 0.01)$
ABSNJZH	$\ln(x + \lvert\min(x)\rvert + 0.01)$
SAVNCPP	$\ln(x + \lvert\min(x)\rvert + 0.01)$
USFLUX	$\ln(x + \lvert\min(x)\rvert + 0.01)$
MEANPOT	$\ln(x + \lvert\min(x)\rvert + 0.0001)$
TOTFZ	$\ln(-x + \lvert\max(x)\rvert + 0.01)$
TOTFY	$\ln(\lvert x\rvert)$
TOTFX	$\ln(\lvert x\rvert)$

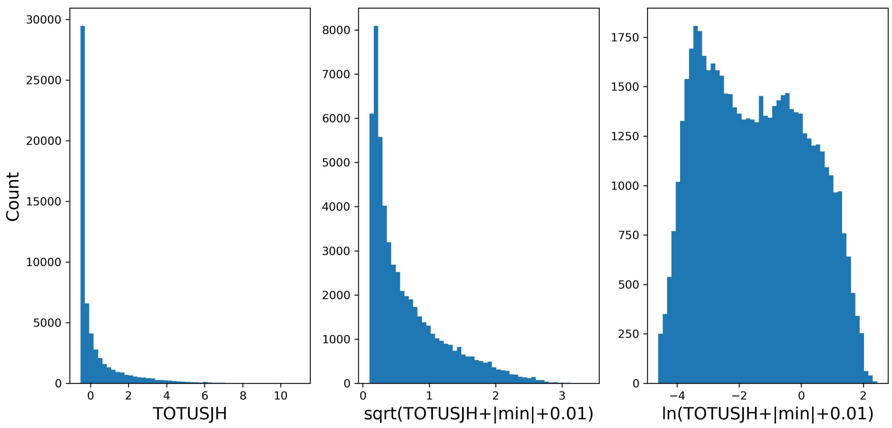

Figure 1: Example of two transformations of the parameter `TOTUSJH` (left). While the root squared transformation produces a better coverage of the distribution (centre), the transformation of the bins with the natural logarithm (right) yields a distribution more useful for clustering.

4.3 Dimensionality Reduction

High-dimensional data is computationally expensive to process. If possible, it is important to reduce the number of dimensions. In addition, clustering methods and other techniques based on the calculation of distances in an Eulerian space are subject to the 'curse of dimensionality': in high dimensions every point tends to be equidistant to each other point. Moreover, we want to reduce high correlations by removing redundant features. Figure 2 (left) illustrates the presence of correlations between the magnetic field parameters. This is not surprising, since they often depend on the same magnetic coefficients, e.g. $\mathbf{B}_z$ and $\mathbf{J}_z$ (see Table 1). These redundant features do not add any relevant information and may hinder the learning algorithm, possibly causing overfitting (Yu & Liu, 2004). To mitigate this problem, we applied Common Factor Analysis (Spearman, 1904) (CFA) to our data set.

4.3.1 Common Factor Analysis

Common Factor Analysis (CFA) is a technique which searches for latent, unobserved variables, called factors, from a set of observed variables. The package `FactorAnalyzer` of (Biggs, 2019) is used. The number of factors is determined with the help of *Horn's Parallel Analysis* (Horn, 1965). Figure 2 (right) shows the resulting factor loadings, a measure of how much a factor explains the associated magnetic field parameters. The first factor has high explanatory power for multiple magnetic field parameters, which confirms that many of these parameters are inter-correlated. Calculation of the covariance of the selected five factors confirms that they show zero covariance with each other.

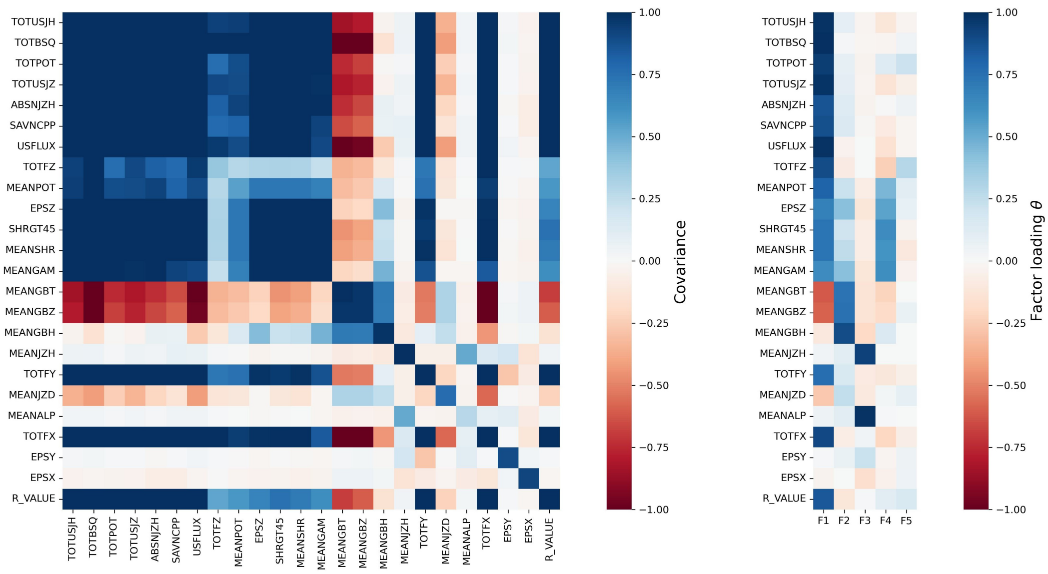

Figure 2: Left: Covariance matrix of the data set before applying CFA on it. A lot of the parameters are strongly correlated with each other. Right: Heatmap of factor loadings of CFA.

4.3.2 Sparse Autoencoders

Makhzani and Frey (2014) shows improvement in classification tasks when sparse data representations are used. To improve sparsity in our data set, we applied an additional data processing step. Sparse autoencoders are able to transform the data into a higher dimensional space, where it is possible to create hyperplanes that allow to separate different clusters of points.

Sparse autoencoders are a special kind of unsupervised neural networks. For an explanation on neural networks, we refer the reader to the notes of Ng et al. (2011). The underlying mathematics of autoencoders are the same as for neural networks. The special property of autoencoders is that the target values ($\hat{X}$) are set equal to the input values (X) (Hinton & Salakhutdinov, 2006): $f : X \rightarrow \hat{X}$, where $X \approx \hat{X}$. The model learns an approximation of the identity function. This may seem like a trivial task, but by placing constraints on the network interesting structures can be discovered.

In a basic (vanilla) autoencoder, also called encoder-decoder, $AE = \{f, f'\}$, the applied constraint consists to limit the number of nodes in an intermediary hidden layer to less than the number of input features of the model: the autoencoder functions are defined as $f : X \in \mathbb{R}^n \rightarrow Z \in \mathbb{R}^m$, followed by $f' : Z \in \mathbb{R}^m \rightarrow \hat{X} \in \mathbb{R}^n$, where $n > m$. A second autoencoder category corresponds to sparse autoencoders (Jiang et al., 2015), where the constraint is applied by forcing sparsity in the intermediary hidden layer. In this case the dimension of the hidden layer does not have to be smaller than the input layer. This sparsity constraint ensures that only a few hidden nodes are allowed to be active at the same time, i.e. most of the hidden nodes will have a value of zero. Sparse autoencoders provide an information bottleneck without having to reduce the number of nodes. This also means that low dimensional data sets can be projected into a higher dimension where sparsity is encouraged, allowing for a better differentiation between different classes.

4.3.2.1 Implementation Details The sparse autoencoder is implemented using Python, together with libraries `Tensorflow` (Abadi et al., 2015) and `Keras` (Chollet et al., 2015). Any kind of neural network learns by minimizing a cost, or loss function, obtained by comparing the output of the model with the expected output. The loss function, Eq. 1, consists of two terms: (1) a reconstruction error and (2) a sparsity penalty. As reconstruction error the mean squared error is used. The sparsity penalty is a regularization acting on the outputs of individual neural network nodes in the hidden layer. It penalizes the activation of the hidden nodes, $a_i^{(h)} \in Z$, using the L1-norm. In the sparsity term of Eq. 1, λ is the pre-factor that determines the influence of the sparse regularization.

$$L = \frac{1}{n} \sum_i (X_i - \hat{X}_i)^2 + \lambda \sum_i \left| a_i^{(h)} \right| \tag{1}$$

The autoencoder is optimized following the traditional error minimization techniques used in classical neural networks. The optimization algorithm that we selected is the `Adam` (Kingma & Ba, 2015) technique. This is an extension to stochastic gradient descent that maintains separate learning rates for each parameter.

To determine the accuracy of the output the R-squared metric, Eq. 2 is used:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(X_i - \hat{X}_i)^2}{\sum_{i=1}^{N}(X_i - \overline{X_i})^2} \text{ with } \overline{X_i} = \frac{1}{N}\sum_{j=1}^{N} X_j \tag{2}$$

To reduce the influence of the class imbalance, different weights have been assigned to the data samples corresponding to different flare classes. A weight of respectively 1, 4, 16 and 64 has been assigned to classes No-flare, C-flare, M-flare and X-flare.

In the Adam optimization algorithm one of the hyperparameters is the learning rate. This hyperparameter influences the speed at which the model converges towards the minimum loss. The optimal learning rate is determined using the method introduced by Smith (2017). This method trains a network starting with a low learning rate, which is exponentially increased throughout the epochs (training cycles). The optimal learning rate corresponds to the fastest decrease in loss throughout the training. An additional method to determine the optimal learning rate is to run the algorithm for multiple values of the learning rate for a limited number of epochs, and to select one with the lowest validation loss. In our work, the combination of these two optimization methods yields an optimal learning rate of 0.0005.

Our data set is split into three sub-groups: 60% training, 20% validation and 20% testing data. The split is performed using stratification, which means that in each data portion the percentage of each flare type is preserved.

4.3.2.2 Architecture Optimization To find the optimal autoencoder architecture, three parameters need to be optimized: (1) the magnitude λ of the sparsity constraint, (2) the number of hidden nodes and (3) the activation function.

If the sparsity pre-factor is too high, all hidden nodes will tend to produce values of zero; if this parameter is too small, no sparsity will be introduced. The optimal value of λ is obtained by finding a balance between the level of sparsity and the activity on the hidden nodes. The pre-factor needs to be set to ensure that only part of the nodes (less than the number of input nodes) are active at the same time, without leaving inactive nodes. This balance is found for $\lambda = 0.1$.

The most adequate architecture is selected by comparing the loss function between the training and the validation set. The optimal architecture contains one hidden layer with seven hidden nodes and uses SELU (Klambauer et al., 2017) activation function.

4.3.2.3 Resulting Distributions The resulting optimal sparse autoencoder is used to increase the dimensionaliy, generating sparsity in the data set. The R-squared metric returns a value of 0.9942, indicating that the model is able to nearly perfectly mimic the original distributions. A two-dimensional projection of the distribution of each pair of parameters in the final data set is shown in Figure 3. This higher dimensional encoding of the data will be used for clustering in later sections.

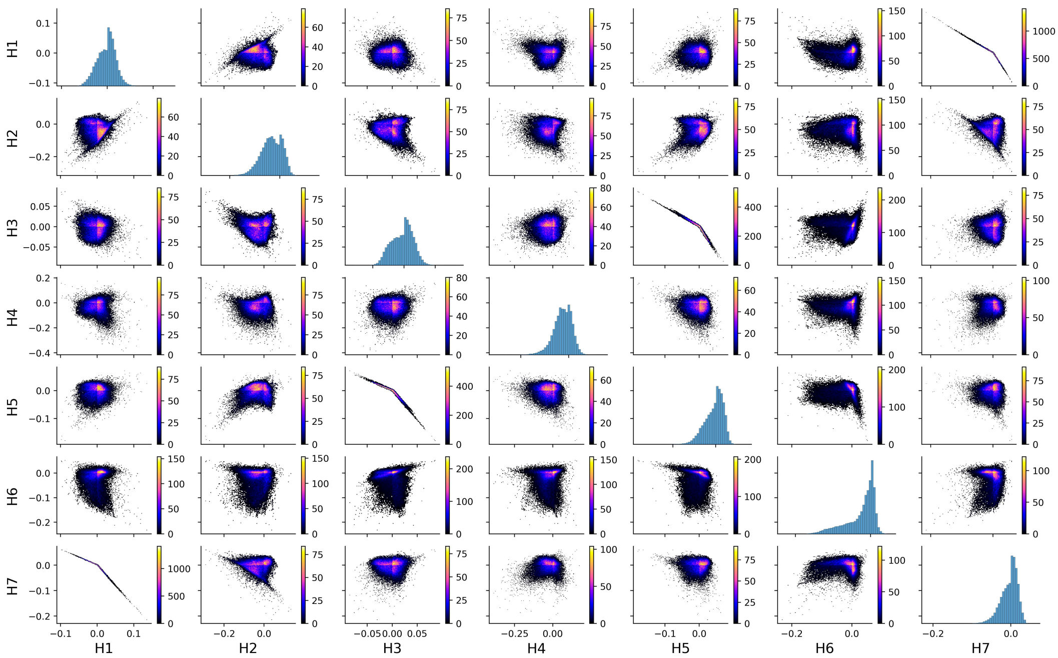

Figure 3: Distributions of the encoded data produced by the hidden layer of the sparse autoencoder. The autoencoder includes one hidden layer, with seven neurons, and SELU activation functions. The pre-factor λ for the activity regularization is set to 0.1.

4.4 Data Sampling

Solar flare data is by definition largely imbalanced, since strong solar flares are scarce, affecting the classification results. Machine learning methods tend to favor the dominant class, which in our case corresponds to the non-flaring active regions. The four different flare activity classes are either over-sampled or under-sampled to construct a balanced data set with a similar amount of data points per flare class. A random under-sampling of the No-flares was already presented in section 4, but the imbalance among flare classes is still large.

4.4.1 Random Sampling

Random sampling can be applied to either under-sample or over-sample data. The methods `RandomUnderSampler` and `RandomOverSampler` of the package `imbalanced-learn` (Lemaître et al., 2017) are used. Random under-sampling picks samples from the majority classes without replacement, while over-sampling picks samples from the minority classes with replacement. However, random over-sampling of the minority class can lead to duplication, which might lead to overfitting. Therefore an alternative over-sampling method is used.

4.4.2 SMOTE Sampling

The alternative Synthetic Minority Over-sampling TEchnique (SMOTE) (Chawla et al., 2002) technique is also included in the `imbalanced-learn` package. SMOTE does not duplicate any samples, but generates new data points by randomly selecting a minority class instance (a), and then finding its k nearest neighbors. Subsequently, one of those k neighbors (b) is chosen at random and a synthetic example is created at a random point on the line segment between the instance (a) and its selected neighbor (b).

4.4.3 Resulting Data Set

It has been shown by Chawla et al. (2002) that the combination of SMOTE and under-sampling performs better than plain under-sampling. In our work the majority classes, No-flare and C-flare, are randomly under-sampled, while the minority classes, M-flare and X-flare, are over-sampled with SMOTE. Every class is sampled to 6000 samples, making the data set balanced.

5 Clustering

We tested multiple clustering algorithms on the data set to classify the solar active regions based on their processed magnetic field parameters and found common aspects among the corresponding active regions.

Clustering is a machine learning method which groups data in subgroups that share similar properties (in our case, similar reduced magnetic field parameters). A good clustering method minimizes the intra-cluster distances, while maximizing inter-cluster distances (Zhang & Tsai, 2005). The implementation and the way clusters are defined differ from method to method. Every method that is considered here is implemented with the `scikit-learn` package.

5.1 k-Nearest Neighbors (supervised)

k-Nearest Neighbors (KNN), explained in e.g. Cunningham and Delany (2007), is a supervised and instance-based clustering algorithm. It assumes similar objects exist in close proximity to the evaluated data point. The class of a data point is determined based on the most frequent class among its k nearest neighbors.

The optimal number of neighbors k is the one that minimizes the error, the percentage of wrong predictions, while maintaining the ability to make accurate predictions on new data. The method minimizes the loss on the validation data, without overfitting on the training data. In general, lower k makes the predictions less stable. Increasing the number of neighbors makes the predictions more stable due to averaging and therefore more likely to produce reliable results. We selected the optimal k by performing the KNN algorithm for a range of k-values, fitting a fourth order polynomial to the corresponding error values and selecting the k corresponding to the minimum error.

5.2 K-means (unsupervised)

K-means (Lloyd, 1982; MacQueen, 1967) is an unsupervised, centroid-based clustering method and assumes that the clusters are spherical and equally sized. The method works best when the clusters are equally dense and not too contaminated by noise or outliers. The clustering is achieved by iteratively assigning each data point to its nearest centroid and creating new centroids by computing the mean of each cluster.

The optimal number of clusters is determined by a *scree* plot (Cattell, 1966), where the 'knee' point is associated to the optimum value, and corresponds to the inflection point of the curve. The position of this 'knee' is determined through the *Kneedle algorithm* (Satopaa et al., 2011). The scree plot is configured by computing the error for different runs for a range of different number of clusters. A line is plotted between the first and last point of the curve and the distances between each point and the line are computed. The point with maximal distance between the two lines marks the maximum of curvature, i.e. the elbow.

5.3 Gaussian Mixture Models (unsupervised)

Gaussian Mixture Models (GMM) assume that all data points are generated from a mixture of Gaussian distributions and identifies for each data point the probabilities of belonging to each of the Gaussian distributions. This method allows the detection of more elongated clusters. The Gaussian distributions are approximated by the Expectation-Maximization method (Dempster et al., 1977). The GMM is a probabilistic method.

To determine the number of clusters for GMM, several methods can be used. We chose to use the gradient of the Bayesian Information Criterion (BIC). BIC (Schwarz, 1978) gives an estimation on how accurately the model represents the existing data, with lower BIC value indicating a better estimation. BIC is defined in Eq. 3, with k the number of unknown model parameters (mean and variance for each cluster), n the number of samples and $\hat{L}$ the maximum likelihood.

$$BIC = k \ln n - 2 \ln \hat{L} \tag{3}$$

A high number of clusters corresponds to low BIC scores, but the error curve shows an inflection point. This point can be found by checking the gradient of BIC. The optimal number of clusters is the point where the gradient no longer changes, i.e. when the second derivative is zero (Lavorini, 2018).

6 Evaluation Methods

To determine the quality of a clustering method a good evaluation method is essential. An Area Under the Curve Receiver Operating Characteristics (AUC-ROC) plot (Fawcett, 2006) is a good evaluation technique for supervised classification methods, when the data is severely imbalanced (Brownlee, 2020).

ROC curves are in general used in binary classifications, but can be extended to multi-class data by using one-vs-rest for each class, which provides one ROC curve per class. The macro-average can be computed by taking the average of all ROC curves, treating all classes equally.

The ROC curve is a visual measure of the predictive quality of the model, that visualizes the trade-off between sensitivity and specificity. The plot of a ROC curve displays the True Positive Rate (TPR), see equation 4, on the y-axis and the False Positive Rate (FPR), see equation 5, on the x-axis. These rates are computed for different thresholds. The threshold is the lowest probability necessary to be assigned to the positive cluster.

$$TPR = \frac{TP}{TP + FN} \tag{4}$$

$$FPR = \frac{FP}{TN + FP} \tag{5}$$

An AUC score can be computed from the ROC, by computing the area under the curve. AUC is a measure of the ability of a classifier to distinguish between classes, where e.g. 0.7 means that in 70% of the cases the model is able to distinguish between the positive and the negative class (Narkhede, 2018).

In addition, the True Skill Statistic, also called the Hanssen score (Hanssen & Kuipers, 1965), will be computed for the supervised clustering, see equation 6. The value of TSS lies between -1 and 1, with a higher value indicating a better forecast. This is one of the most used evaluation metrics to assess solar flare forecasts.

$$TSS = \frac{TP}{TP + FN} - \frac{FP}{FP + TN} = \frac{TP}{P} - \frac{FP}{N} \tag{6}$$

It is a lot harder to assess whether unsupervised clustering methods perform well, because no labels are present. A viable alternative are validation methods that check whether there is a high separation between clusters and a high cohesion within the clusters. Examples of such metrics are the Calinsky-Harabasz (CH) coefficient (Caliński & Harabasz, 1974) and the Silhouette coefficient (SC) (Rousseeuw, 1987). The Calinski-Harabasz coefficient is defined as the ratio between the within-cluster dispersion and the between-cluster dispersion. This coefficient should be maximized. The Silhouette coefficient is computed, for each sample, using: (a) the mean inter-cluster distance, and (b) the mean nearest-cluster distance. The formula is given in equation 7. The final Silhouette score is found by computing the mean over all samples. The best value is 1, the worst is -1 and values near 0 indicate that the clusters overlap. If the value is negative it is generally an indication that samples are assigned to the wrong cluster, as it is found that a different cluster is more similar.

$$SC = \frac{b - a}{max(a, b)} \tag{7}$$

7 Results

Figure 4 shows the mean value and standard deviation of each of the seven reduced parameters, for each flare class. In general, the parameters are very similar for all flaring active regions (C, M and X-flares). X-flare classes present only slight differences with respect to the other flaring classes. Parameters H2, H5 and H6 have a larger absolute mean value for these stronger flare classes. The mean value of the data without flares

(No) is clearly different. It can be expected that flaring active regions will be distinguishable from non-flaring active regions, while distinguishing between the different flare classes may be more challenging with the available data.

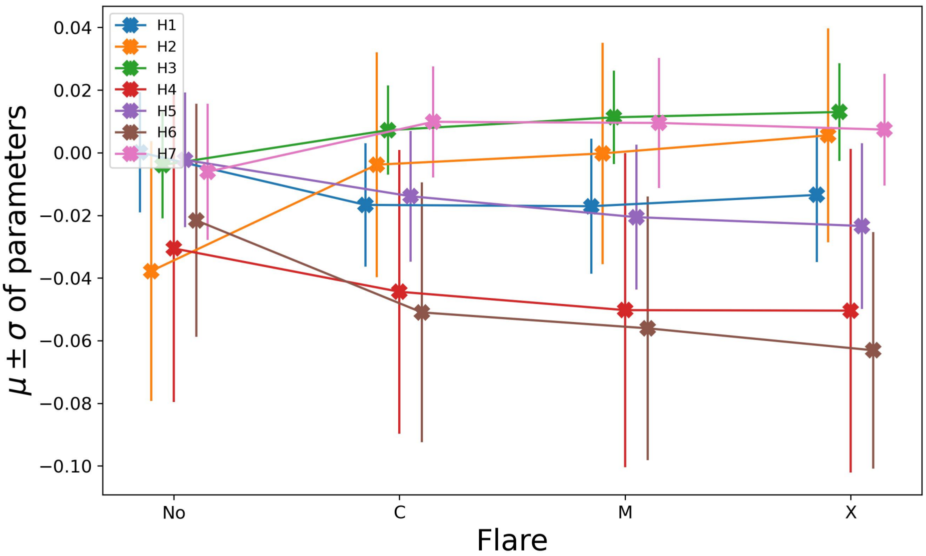

Figure 4: Mean and standard deviation of the features resulting from the sparse autoencoder, per flare label. The flaring data looks very similar, while the non-flaring data has distinct parameter values.

7.1 Supervised (KNN)

In our work the hyperparameter selection for KNN was based on the data set before the sampling procedure used in section 4.4, to avoid using under-/over-sampled data points. Performing the hyperparameter selection on the sampled data yields an optimal number of neighbors of one, which leads to unstable results. By applying the hyperparameter selection on the data set before sampling, we find an optimal number of neighbors of ten. To validate this selection method, the KNN clustering is conducted multiple times, testing the use of one, three, six and ten nearest neighbors. The resulting ROC curves are shown in Figure 5. These figures show that when more neighbours are taken into account for the clustering, the results improve, producing a higher value for the area-under-the-curve. This is the case for the macro-average along the whole data set, as well as for the individual flare types. This shows that taking only one neighbor into account would not have been optimal. The differences between the results with three, six and ten neighbors are not too large.

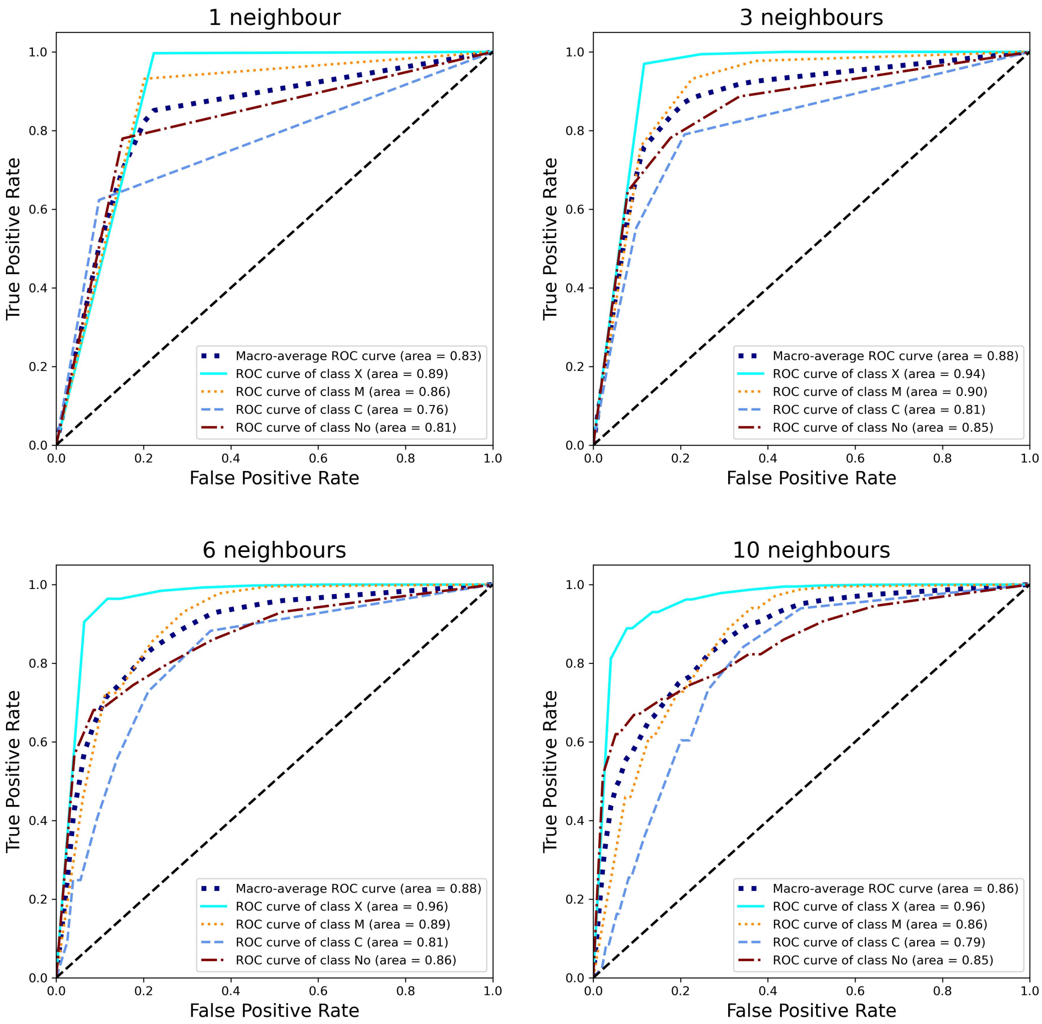

Figure 5: AUC ROC plot of the results of KNN, performed on the sampled data set, for varying number of neighbors.

Figure 6 shows the normalized confusion matrices for the clustering of KNN. On the x-axis the figure shows the predictions and on the y-axis the true classes. On the left panel we present the results using one nearest neighbor, and on the right panel the result when ten neighbors are considered. The largest difference is observed in the number of C-flares that are classified correctly. When more neighbors are taken into account, the C-flares are more often misclassified as larger M- and X-flares. On the other hand, when more neighbors are taken into account, C-flares are less often misclassified as non-flaring. The fact that the C-flares are more often misclassified as stronger flares is not necessarily a bad thing. For flare prediction, we are most interested in recognising the strongest flares. Therefore, it could be considered better to have a prediction method that is more likely to overestimate the strength of a flare, than to underestimate the strength of a flare. However, false warnings will lessen the trust of the industry in flare predictions, so ideally we want to minimize both the false positives and the false negatives.

The percentage of true positives for each flare type is higher when only one neighbor is taken into account versus when ten neighbors are taken into account. While the results with one neighbor might look better on this figure, they are unstable and more influenced by the artificial data introduced by the sampling.

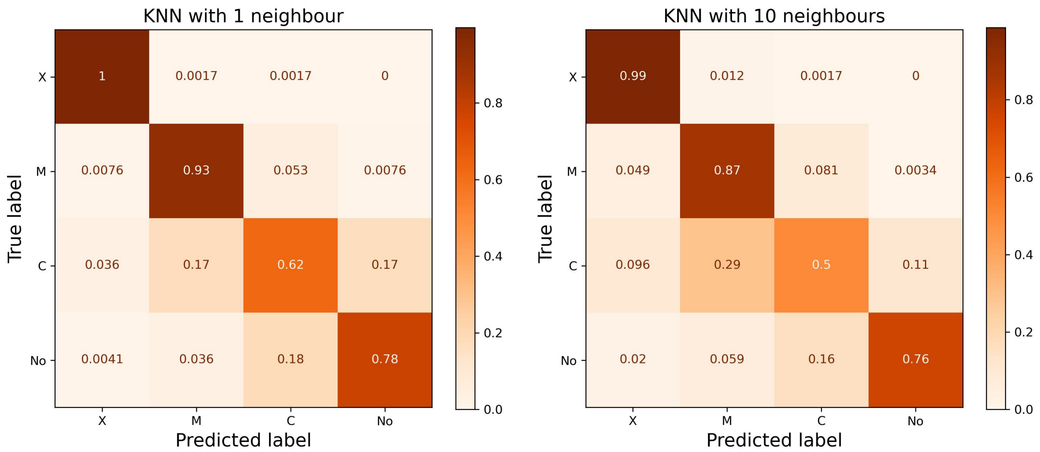

Figure 6: Normalized confusion matrices of the results of KNN with (left) only one nearest neighbor and (right) ten nearest neighbors taken into account.

Focusing on the confusion matrix in the right panel of Fig. 6, the following conclusions can be made: almost all of the X-flares are correctly identified. However, this is probably influenced by the over-sampling of the X-flares by a factor of approximately 160. 87% of the true M-flares are correctly identified. This high percentage is also somewhat influenced by the over-sampling. When M-flares are misclassified, it is $\sim$ 37% of the time as an X-flare and $\sim$ 61% of the time as a C-flare. 76% of the non-flaring active regions are correctly classified as well. This is quite a good result, considering that this class is largely under-sampled. The non-flaring active regions are most of the time mistaken for C-flares. Finally, the C-flares turn out to be hardest to distinguish, with only 50% of the active regions correctly identified as C-flares. They are $\sim$ 58% of the time overestimated as M-flares, $\sim$ 19% of the time as X-flares and $\sim$ 22% of the time underestimated as non-flaring. The flares are mostly mistaken for their neighboring classes, in terms of X-ray flux strength. This indicates that the clusters are partly overlapping.

The TSS has been calculated for each of the flare types separately. A TSS of 0.93 is found for the X-flares, 0.75 for the M-flares, 0.42 for the C-flares and 0.72 for the non-flaring active regions.

7.2 Unsupervised (K-means + GMM)

Unsupervised clustering methods are more useful in practice, since there is not always information present about the flaring nature of an active region. These methods do not take into account the information about the X-ray flux, but only the reduced magnetic field parameters. For both unsupervised methods used in this work (K-means and GMM) the number of clusters needs to be determined using a hyperparameter optimization technique, as described in sections 5.2 and 5.3. For K-means an optimal number of four (4) clusters is found, while GMM has an optimal number of three (3) clusters.

Table 3 shows the Calinski-Harabasz (Caliński & Harabasz, 1974) and Silhouette (Rousseeuw, 1987) coefficients, which evaluate the clusters found through K-means and GNN. The first one should be maximized, while the latter should be as close to 1 as possible. Both coefficients indicate that K-means does a better job at clustering the data. However, a relatively low Silhouette score of 0.25 indicates that the clusters are either not very well separated or the points within a cluster are distributed relatively far apart. The possibility that the clusters are overlapping was already mentioned in the previous section.

Table 3: Evaluation coefficients for K-means and GNN.

	K-means	**GMM**
Calinski-Harabasz	7506	1886
Silhouette	0.25	0.12

With unsupervised machine learning methods no confusion matrix can be constructed, since no labels are used. However, we have already access to the expected flare classification in the data set. These values are not used to train the unsupervised clustering algorithms. We used this information to evaluate the accuracy of the automatic unsupervised classification with respect to the expected flare classes. The resulting visualization is shown in Figure 7, where for each of the two clustering algorithms the percentage of each flare included in each of the clusters is shown. Normalization is performed per flare type.

Analyzing the clusters of K-means learns us that 66% of the non-flaring active regions are included in Cluster 3. Cluster 3 also includes 17% of the C-flares, 12% of the M-flares and 5% of the X-flares. This cluster can be considered as one with mostly non- and weakly-flaring active regions. If an active region is classified in Cluster 3, chances are thus relatively low that it is a strong flare. Clusters 1, 2 and 4 contain less non-flaring active regions, respectively 14%, 7% and 12%. They do contain more of the flaring active regions. Cluster 2 contains $\sim$ 40% of each of the flare types. Cluster 4 contains $\sim$ 40% of the X-flares and only $\sim$ 20% of the C- and M-flares. Cluster 1 also contains flaring active regions, with more C- and M-flares than X-flares. Since all four clusters contain a significant fraction of all four flare types, there is no way to determine with certainty the type of flare, based on this clustering of the active regions. What one could conclude from these results is that an active region that is classified in Cluster 3 is most likely to be non-flaring or weakly flaring. On the other hand, an active region that is classified in Cluster 4 has a higher probability to be an X-flare, since these are most abundantly present. If an active region is classified in Cluster 2, it is very probable to be flaring, but nothing can be concluded about the type of flare. Finally, if an active region is classified in Cluster 1, it is most probable to produce a C- or M-flare.

The resulting clusters found with GMM are visualized in Figure 7 on the right. Cluster 3 contains 52% of the non-flaring active regions and 14 to 18% of the flaring active regions. Meanwhile, Cluster 2 contains 34% of the non-flaring active regions and 8 to 18% of the flaring active regions. Active regions that are classified into Cluster 2 and Cluster 3 have thus a relatively large probability to be non-flaring. This statement can be made stronger when the probabilities to belong to multiple clusters are analysed. If an active region has a high probability to belong to both Cluster 2 and Cluster 3, it is highly probable to be non-flaring. Cluster 1 contains only 14% of the non-flaring active regions and 68 to 78% of each of the flaring active regions. This cluster is thus a good one to identify flaring active regions.

In each of the clusters found with GMM, the percentage of each of the different types of flaring active regions is very similar. Therefore, in contrast to K-means, the clustering with GMM is not able to distinguish the strength of the flares.

To get a more quantitative analysis, Figure 8 is a useful addition to 7. They show the same data, but in Figure 8 the normalization is performed per cluster. Therefore, this visualisation can be used to determine the probability that an active regions is of a certain flare type if it belongs to a certain cluster. We clarify this by giving a few examples. When an active regions is assigned to Cluster 3 by the K-means algorithm, it is with 66% probability non-flaring, with 17% a C-flare and with 12% probability an M-flare. An active regions that is assigned to Cluster 2 by K-means will with 94% probability (31% + 33% + 30%) be flaring, with approximately equal probability to be a C-flare, M-flare or X-flare. If an active region belongs to Cluster 1, found with GMM, there is only a 6% chance that it is not flaring. However, when the active region is assigned

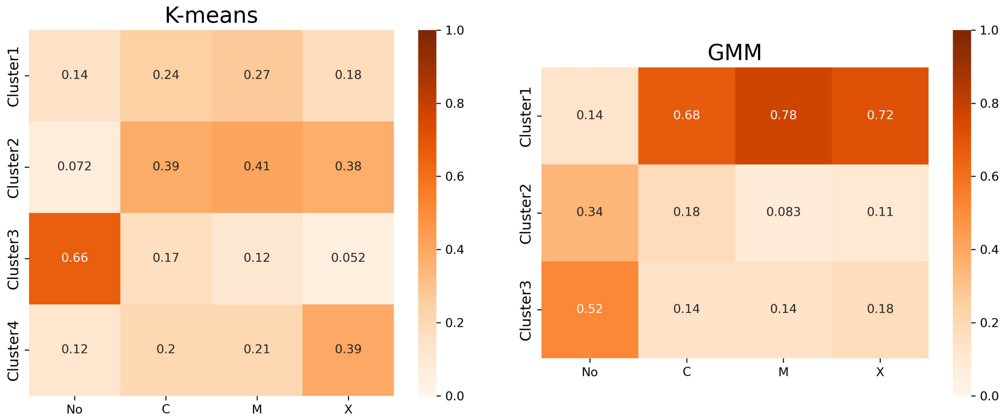

Figure 7: Clustering results of K-means (left) and GMM (right) on the sampled data set. The percentage of each flare included in each of the clusters is shown, where normalization is performed per flare type.

to Cluster 2 or 3 by GMM, there is respectively a chance of 48% and 53% that there are no flares coming out of this active region.

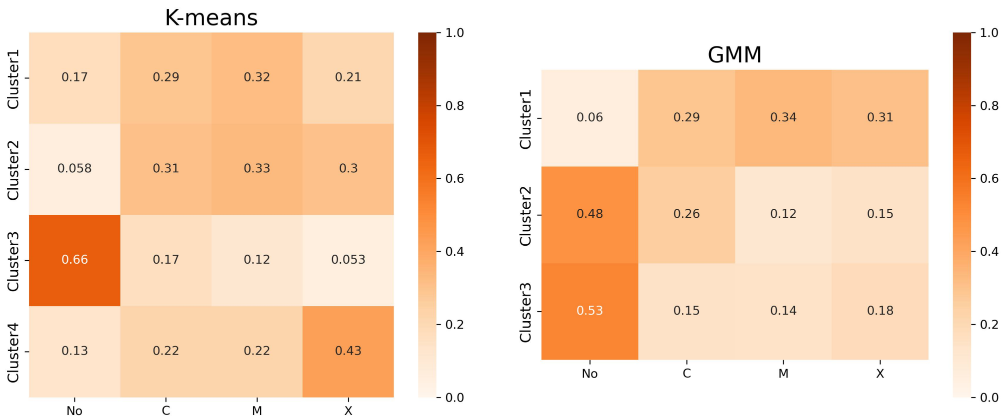

Figure 8: Clustering results of K-means (left) and GMM (right) on the sampled data set. The percentage of each flare included in each of the clusters is shown, where normalization is performed per cluster.

8 Discussion

8.1 Data Processing

In section 4.3.1, we found with Common Factor Analysis that almost all of the information included in the 24 magnetic field parameters could be reduced to only five factors. This is because a lot of the initial parameters were strongly correlated, and do not add any additional information. It is possible then to construct a smaller data set, with only the most useful parameters, containing different distributions for different flare types. This redundancy due to intrinsic correlations between the parameters was also mentioned previously in Bobra and Couvidat (2015) and Barnes et al. (2016).

8.2 Active Region Classification

The supervised clustering method (KNN) has good performance for the M- and X-flares, as well as for the non-flaring active regions. The performance on the C-flares is less accurate, since they are often confused with M-flares and non-flaring active regions. This is probably because their magnetic field parameters are similar to the ones of both the non-flaring data and the M-flares, and their distributions tend to overlap.

With unsupervised clustering (K-means and GMM), non-flaring active regions can be distinguished from flaring active regions. To distinguish between the different flaring active regions is a lot harder. The resulting clusters from K-means show that it is possible to make a distinction between an active region producing strong flares from active regions producing weak flares, but there is still a lot of uncertainty in the distinction among the different flaring energy levels.

The difficulty of differentiating between the flare types is inherent to the data itself, as predicted by analysis of Figure 4. The parameters are very similar for all flaring active regions. Therefore, there is not enough information in the data set for the technique to identify clear differences between C-flares, M-flares and X-flares. Integrating more information into the analysis could provide a clearer distinction. The vector magnetic field data alone is not fully representative of the activity in the whole active region. For example, the maximal difference in magnitude of the magnetic field over the active region could provide valuable information. In future research, the magnetic field parameters should be combined with other features, created through good feature engineering from the original images, for example through edge detection or with variational autoencoders. More data can be included by taking into account EUV observations, at multiple wavelenghts, of the same region.

An extension to the use of the magnetic field parameters is to study their evolution, through time series. The variation of the magnetic field in anticipation of the release of a flare will provide valuable information, being probably more significant for strong flares than for weak flares. The use of time series can also help to distinguish the natural variability of the solar magnetic field from a sudden change in the magnetic field due to flare formation.

The difficulty of differentiating C-, M- and X-flares is also caused by the arbitrary boundaries of the classes, determined by their peak X-ray flux. A C9-flare is very similar to an M1-flare, but they were for this work considered as strictly different classes of flares. The difference between background radiation (non-flaring active regions) and weak C-flares can be very small as well. The strength of flares is a continuous parameter, but was here treated as strictly discrete.

Rather than trying to cluster C-, M- and X-flares separately, trying to distinguish flaring from non-flaring, or weakly flaring from strongly flaring active regions might yield more accurate results. But still the problem remains that an artificial boundary needs to be set in the continuous domain.

Strongly flaring active regions could also be identified as regions with parameter values significantly larger than the mean or median value. Both Sun et al. (2022) and Bobra and Couvidat (2015) tried to identify flaring active regions based on a training set containing only active regions that were either non-flaring or strongly flaring. All active regions that produced C-flares were eliminated. This makes it easier to distinguish flaring from non-flaring active regions. However, for flare prediction, in real-time data the C-flares can not be eliminated and need to be classified correctly as well.

In future research, it could be useful to only consider flaring data. When both non-flaring and flaring data is taken into account, regions with complex and intense magnetic fields are compared against completely quiet regions. This might give the impression that all flaring active regions have similar properties. It is possible that they do appear more distinct when only compared against each other.

9 Conclusion

Throughout this work detailed data cleaning and parameter transformation was conducted to enhance the quality of the Angryk data set and improve the classification results. Supervised clustering, with KNN, is able to distinguish the M- and X-flares, with respectively 99% and 87% correctly identified. However, only half of the C-flares are accurately classified. Unsupervised clustering, with K-means and GMM, identifies clusters with mainly non-flaring active regions and clusters with mainly flaring active regions. However, the clusters contain a mixture of weakly-flaring and strongly-flaring active regions. There is no clear hyperplane in the SHARP parameter space that can separate active regions with different flaring activity. For future projects, additional information should be included, like time series, different parameters - indicating e.g. the topology of active regions - or images of the active regions.

Open Research

This research uses the open source data set SWAN-SF of Angryk et al. (2020b). For more information we would like to refer the reader to the respective paper (Angryk et al., 2020a). The data is available for download through: `https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/EBCFKM`.

The code used to perform all data transformations and generate the clustering results is completely written in Python 3.10, and is accessible on Gitlab: `https://gitlab.com/hanneb/clustering_ar_sf_hbaeke.git` (Baeke, 2022).

Acknowledgments

This work was supported by the Research Foundation – Flanders (FWO), Frank De Winne PhD-aspirant grant (1SF8522N). This work has also received funding from the KULeuven Bijzonder Onderzoeksfonds (BOF) under the C1 project KULeuven Bijzonder Onderzoeksfonds (BOF), from the European Union's Horizon 2020 research and innovation program under grant agreement No. 955606 (DEEP-SEA). The resources and services used in this work were provided by the VSC (Flemish Supercomputer Center), funded by the Research Foundation – Flanders (FWO) and the Flemish Government. The data used in this project was constructed by Angryk et al. (2020b), which originates from the SHARP data series (Bobra et al., 2011), covering the period from 2010-05-01 until 2018-08-31, and the flare reports from GOES containing SSW and XRT flares. We would like to thank the assessors, Jasmina Magdalenic and Andrew Tkachenko, of the master thesis report for their feedback and suggestions. Their feedback led to an improved discussion section in this paper. Finally, the authors declare that there are no conflicts of interest.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., ... Zheng, X. (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems.* Retrieved from `https://www.tensorflow.org/` (Software available from tensorflow.org)

Abduallah, Y., Wang, J. T. L., Nie, Y., Liu, C., & Wang, H. (2020). DeepSun: Machine-Learning-as-a-Service for Solar Flare Prediction. *arXiv e-prints*, arXiv:2009.04238.

Angryk, R., Martens, P., Aydin, B., Kempton, D., Mahajan, S., Basodi, S., ... Georgoulis, M. (2020a). Multivariate Time Series Dataset for Space Weather Data Analytics. *Scientific Data*, *7*, 227. doi: 10.1038/s41597-020-0548-x

Angryk, R., Martens, P., Aydin, B., Kempton, D., Mahajan, S., Basodi, S., . . . Georgoulis, M. (2020b). *SWAN-SF [Dataset].* Retrieved from https://doi.org/10.7910/DVN/EBCFKM (Harvard Dataverse) doi: 10.7910/DVN/EBCFKM

Baeke, H. (2022). *Identifying Solar Flares by Clustering Active Regions [Code].* Retrieved from https://gitlab.com/hanneb/clustering_ar_sf_hbaeke.git (GitLab)

Barnes, G., Leka, K. D., Schrijver, C. J., Colak, T., Qahwaji, R., Ashamari, O. W., . . . Wagner, E. L. (2016). A Comparison of Flare Forecasting Methods. I. Results from the All-Clear Workshop. *The Astrophysical Journal*, *829*(2), 89. Retrieved from https://dx.doi.org/10.3847/0004-637X/829/2/89 doi: 10.3847/0004-637X/829/2/89

Biggs, J. (2019). *FactorAnalyzer release 0.3.2.* Retrieved from https://github.com/EducationalTestingService/factor_analyzer

Bobra, M., Hoeksema, J. T., Sun, X., & Turmon, M. (2011). SHARPs - A New Space Weather Data Product from SDO/HMI. In *Agu fall meeting abstracts* (Vol. 2011, p. SH51B-2006).

Bobra, M. G., & Couvidat, S. (2015). Solar Flare Prediction Using SDO/HMI Vector Magnetic Field Data with a Machine Learning Algorithm. *The Astrophysical Journal*, *798*(2), 135. Retrieved from https://doi.org/10.1088/0004-637x/798/2/135 doi: 10.1088/0004-637x/798/2/135

Bobra, M. G., Wright, P. J., Sun, X., & Turmon, M. J. (2021). SMARPs and SHARPs: Two Solar Cycles of Active Region Data. *The Astrophysical Journal Supplement Series*, *256*(2), 26. Retrieved from https://doi.org/10.3847%2F1538-4365%2Fac1f1d doi: 10.3847/1538-4365/ac1f1d

Brownlee, J. (2020). *Tour of Evaluation Metrics for Imbalanced Classification.* Retrieved from https://machinelearningmastery.com/tour-of-evaluation-metrics-for-imbalanced-classification/ (machinelearningmastery.com)

Caliński, T., & Harabasz, J. (1974). A Dendrite Method for Cluster Analysis. *Communications in Statistics*, *3*(1), 1-27. Retrieved from https://www.tandfonline.com/doi/abs/10.1080/03610927408827101 doi: 10.1080/03610927408827101

Campello, R. J. G. B., Moulavi, D., & Sander, J. (2013). Density-Based Clustering Based on Hierarchical Density Estimates. In *Advances in knowledge discovery and data mining* (pp. 160–172). Berlin, Heidelberg: Springer Berlin Heidelberg.

Cattell, R. B. (1966). The Scree Test For The Number Of Factors. *Multivariate Be-*

havioral Research, *1*(2), 245-276. Retrieved from https://doi.org/10.1207/s15327906mbr0102_10 (PMID: 26828106) doi: 10.1207/s15327906mbr0102_10

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, *16*, 321–357. Retrieved from http://dx.doi.org/10.1613/jair.953 doi: 10.1613/jair.953

Chen, Y., Manchester, W. B., Hero, A. O., Toth, G., DuFumier, B., Zhou, T., … Gombosi, T. I. (2019). Identifying Solar Flare Precursors Using Time Series of SDO/HMI Images and SHARP Parameters. *Space Weather*, *17*(10), 1404–1426. Retrieved from http://dx.doi.org/10.1029/2019SW002214 doi: 10.1029/2019sw002214

Chollet, F., et al. (2015). *Keras.* https://keras.io.

Cinto, T., Gradvohl, A. L. S., Coelho, G. P., & da Silva, A. E. A. (2020). A Framework for Designing and Evaluating Solar Flare Forecasting Systems. *Monthly Notices of the Royal Astronomical Society*, *495*(3), 3332–3349. Retrieved from http://dx.doi.org/10.1093/mnras/staa1257 doi: 10.1093/mnras/staa1257

Colak, T., & Qahwaji, R. (2008). Automated McIntosh-Based Classification of Sunspot Groups Using MDI Images. *Solar Physics*, *248*(2), 277-296. doi: 10.1007/s11207-007-9094-3

Cunningham, P., & Delany, S. (2007). k-Nearest Neighbour Classifiers. *Technical Report UCD-CSI-2007-4*.

Dang, T., Li, X., Luo, B., Li, R., Zhang, B., Pham, K., … Wang, Y. (2022). Unveiling the Space Weather During the Starlink Satellites Destruction Event on 4 February 2022. *Space Weather*, *20*(8), e2022SW003152. doi: 10.1029/2022SW003152

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, *39*(1), 1–22.

Fawcett, T. (2006). Introduction to ROC analysis. *Pattern Recognition Letters*, *27*, 861-874. doi: 10.1016/j.patrec.2005.10.010

Hale, G. E., Ellerman, F., Nicholson, S. B., & Joy, A. H. (1919). The Magnetic Polarity of Sun-Spots. *The Astrophysical Journal*, *49*, 153. doi: 10.1086/142452

Hanssen, A., & Kuipers, W. (1965). *On the Relationship Between the Frequency of Rain and Various Meteorological Parameters: (with Reference to the Problem Ob Objective Forecasting)*. Staatsdrukkerij- en Uitgeverijbedrijf. Retrieved from `https://books.google.be/books?id=nTZ8OgAACAAJ`

Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the Dimensionality of Data with Neural Networks. *Science*, *313*(5786), 504–507. Retrieved from `https://science.sciencemag.org/content/313/5786/504` doi: 10.1126/science.1127647

Horn, J. (1965). A rationale and Test for the Number of Factors in Factor Analysis. *Psychometrika*, *30*, 179-185.

Housseal, S. N., Berger, T. E., & Deshmukh, V. (2019). Using Unsupervised Machine Learning to Explore New Classification of Sunspot Active Regions. In *Agu fall meeting abstracts.*

Ilonidis, S., Bobra, M. G., & Couvidat, S. (2015). Solar Flare Forecasting Using Time Series of SDO/HMI Vector Magnetic Field Data and Machine Learning Methods. In *AAS/AGU triennial earth-sun summit* (Vol. 1, p. 311.03).

Jiang, N., Rong, W., Peng, B., Nie, Y., & Xiong, Z. (2015). An Empirical Analysis of Different Sparse Penalties for Autoencoder in Unsupervised Feature Learning. In *Ijcnn* (p. 1-8). IEEE. Retrieved from `http://dblp.uni-trier.de/db/conf/ijcnn/ijcnn2015.html#JiangRPNX15`

Jiao, Z., Sun, H., Wang, X., IV, W., Gombosi, T., Hero, A., & Chen, Y. (2020). Solar Flare Intensity Prediction With Machine Learning Models. *Space Weather*, *18*. doi: 10.1029/2020SW002440

Jonas, E., Bobra, M., Shankar, V., Todd Hoeksema, J., & Recht, B. (2018). Flare Prediction Using Photospheric and Coronal Image Data. *Solar Physics*, *293*(3), 48. doi: 10.1007/s11207-018-1258-9

Kawabata, Y., Iida, Y., Doi, T., Akiyama, S., Yashiro, S., & Shimizu, T. (2018, dec). Statistical Relation between Solar Flares and Coronal Mass Ejections with Respect to Sigmoidal Structures in Active Regions. *The Astrophysical Journal*, *869*(2), 99. Retrieved from `https://dx.doi.org/10.3847/1538-4357/aaebfc` doi: 10.3847/1538-4357/aaebfc

Kingma, D. P., & Ba, J. (2015). Adam: A Method for Stochastic Optimization. *CoRR*, *abs/1412.6980*.

Klambauer, G., Unterthiner, T., Mayr, A., & Hochreiter, S. (2017). Self-Normalizing Neural Networks. In *Proceedings of the 31st international conference on neural information processing systems* (p. 972–981). Red Hook, NY, USA: Curran Associates Inc.

Knipp, D., Ramsay, A., Beard, E., Boright, A., Cade, T., Hewins, I., ... Smart, D. (2016). The May 1967 Great Storm and Radio Disruption Event: Extreme Space Weather and Extraordinary Responses: May 1967 Solar and Geomagnetic Storm. *Space Weather*, *14*. doi: 10.1002/2016SW001423

Lavorini, V. (2018). *Gaussian Mixture Model clustering: how to select the number of components.* Retrieved from `https://towardsdatascience.com/gaussian-mixture-model-clusterization-how-to-select-the-number-of-components-clusters-553bef45f6e4` (Towards Data Science)

Lemaître, G., Nogueira, F., & Aridas, C. K. (2017). Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning. *Journal of Machine Learning Research*, *18*(17), 1-5. Retrieved from `http://jmlr.org/papers/v18/16-365.html`

Liu, C., Deng, N., Wang, J., & Wang, H. (2017). Predicting Solar Flares Using SDO/HMI Vector Magnetic Data Products and the Random Forest Algorithm. *The Astrophysical Journal*, *843*, 104. doi: 10.3847/1538-4357/aa789b

Liu, J., Wang, W., Qian, L., Lotko, W., Burns, A. G., Pham, K., ... Wilder, F. (2021). Solar Flare Effects in the Earth's Magnetosphere. *Nature Physics*, *17*(7), 807-812. doi: 10.1038/s41567-021-01203-5

Lloyd, S. (1982). Least Squares Quantization in PCM. *IEEE Transactions on Information Theory*, *28*(2), 129-137. doi: 10.1109/TIT.1982.1056489

MacQueen, J. B. (1967). Some Methods for Classification and Analysis of Multivariate Observations. In *Berkeley symposium on mathematical statistics and probability* (p. 281-297).

Makhzani, A., & Frey, B. J. (2014). k-Sparse Autoencoders. *CoRR*, *abs/1312.5663*.

Maloney, S. A., & Gallagher, P. T. (2018). Sunspot Group Classification using Neural Networks. In *Catalyzing solar connections* (p. 92).

McIntosh, P. S. (1990). The Classification of Sunspot Groups. *Solar Physics*, *125*(2), 251-267. doi: 10.1007/BF00158405

Mikaelian, T. (2009). *Spacecraft Charging and Hazards to Electronics in Space.* arXiv. Retrieved from `https://arxiv.org/abs/0906.3884` doi: 10.48550/ARXIV.0906.3884

Narkhede, S. (2018). *Understanding AUC-ROC Curve.* Retrieved from `https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5` (TowardsDataScience)

Ng, A., et al. (2011). Sparse Autoencoder. *CS294A Lecture notes*, *72*(2011), 1–19.

Nguyen, T. T., Willis, C. P., Paddon, D. J., Nguyen, S. H., & Nguyen, H. S. (2006). Learning Sunspot Classification. *Fundamenta Informaticae*, *72*(1–3), 295–309.

Priest, E. R., & Forbes, T. G. (2002). The Magnetic Nature of Solar Flares. *Astronomy and Astrophysics Reviews*, *10*(4), 313-377. doi: 10.1007/s001590100013

Pulkkinen, A., Lindahl, S., Viljanen, A., & Pirjola, R. (2005). Geomagnetic Storm of 29-31 October 2003: Geomagnetically Induced Currents and their Relation to Problems in the Swedish High-Voltage Power Transmission System. *Space Weather*, *3*. doi: 10.1029/2004SW000123

Ran, H., Liu, Y. D., Guo, Y., & Wang, R. (2022). *Relationship between Successive Flares in the Same Active Region and Space-Weather HMI Active Region Patch (SHARP) Parameters.* arXiv. Retrieved from `https://arxiv.org/abs/2207.07254` doi: 10.48550/ARXIV.2207.07254

Redmon, R., Seaton, D., Steenburgh, R., He, J., & Rodriguez, J. (2018). September 2017's Geoeffective Space Weather and Impacts to Caribbean Radio Communications During Hurricane Response. *Space Weather*, *16*. doi: 10.1029/2018SW001897

Rousseeuw, P. J. (1987). Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis. *Journal of Computational and Applied Mathematics*, *20*, 53-65. Retrieved from `https://www.sciencedirect.com/science/article/pii/0377042787901257` doi: https://doi.org/10.1016/0377-0427(87)90125-7

Satopaa, V., Albrecht, J., Irwin, D., & Raghavan, B. (2011). Finding a "Kneedle" in a Haystack: Detecting Knee Points in System Behavior. In *2011 31st international conference on distributed computing systems workshops* (p. 166-171). doi: 10.1109/ICDCSW.2011.20

Schwarz, G. (1978). Estimating the Dimension of a Model. *The Annals of Statistics*, *6*(2), 461 – 464. Retrieved from `https://doi.org/10.1214/aos/1176344136` doi: 10.1214/aos/1176344136

Sheeley, N.R. (2020). Solar Magnetic Field. *AccessScience*.

Sinha, S., Gupta, O., Singh, V., Lekshmi, B., Nandy, D., Mitra, D., ... Pal, S. (2022). A Comparative Analysis of Machine-learning Models for Solar Flare Forecasting: Identifying High-performing Active Region Flare Indicators. *The Astrophysical Journal*, *935*(1), 45. Retrieved from `https://dx.doi.org/10.3847/1538-4357/ac7955` doi: 10.3847/1538-4357/ac7955

Smith, L. N. (2017). *Cyclical Learning Rates for Training Neural Networks.* arXiv. Retrieved from `https://arxiv.org/abs/1506.01186` doi: 10.48550/ARXIV.1506.01186

Smith, M. C., Jones, A. R., & Sandoval, L. (2018). Automating the McIntosh Classification System using Machine Learning. In *Agu fall meeting abstracts* (Vol. 2018, p. SM31D-3526).

Sorkin, A. (1982). Economic Aspects of Natural Hazards. In *Economic aspects of natural hazards.* Lexington Books; distributed Gower.

Spearman, C. (1904). General Intelligence, Objectively Determined and Measured. *The American Journal of Psychology*, *15*(2), 201–292. Retrieved from http://www.jstor.org/stable/1412107

Sun, Z., Bobra, M. G., Wang, X., Wang, Y., Sun, H., Gombosi, T., ... Hero, A. (2022). Predicting Solar Flares Using CNN and LSTM on Two Solar Cycles of Active Region Data. *The Astrophysical Journal*, *931*(2), 163. Retrieved from https://doi.org/10.3847\%2F1538-4357\%2Fac64a6 doi: 10.3847/1538-4357/ac64a6

Wang, X., Chen, Y., Toth, G., IV, W., Gombosi, T., Hero, A., ... Liu, Y. (2020). Predicting Solar Flares with Machine Learning: Investigating Solar Cycle Dependence. *The Astrophysical Journal*, *895*, 3. doi: 10.3847/1538-4357/ab89ac

Yu, L., & Liu, H. (2004). Efficient Feature Selection via Analysis of Relevance and Redundancy. *Journal of Machine Learning Research*, *5*, 1205–1224.

Zhang, D., & Tsai, J. J. P. (2005). *Machine Learning Applications in Software Engineering.* World Scientific. Retrieved from https://www.worldscientific.com/doi/abs/10.1142/5700 doi: 10.1142/5700

Zhang, H., Li, Q., Yang, Y., Jing, J., Wang, J. T. L., Wang, H., & Shang, Z. (2022). Solar Flare Index Prediction Using SDO/HMI Vector Magnetic Data Products with Statistical and Machine-learning Methods. *The Astrophysical Journal Supplement Series*, *263*(2), 28. Retrieved from https://doi.org/10.3847%2F1538-4365%2Fac9b17 doi: 10.3847/1538-4365/ac9b17

Classification of Solar Flares using Data Analysis and Clustering of Active Regions

H. Baeke[1], J. Amaya[1], G. Lapenta[1]

[1]Center for mathematical Plasma Astrophysics, Department of Mathematics, KU Leuven, Celestijnenlaan 200B, 3001 Leuven, Belgium

Key Points:

- SHARP parameters of solar active regions contain redundant information that can be reduced to five parameters using Common Factor Analysis.
- Unsupervised classification allows to differentiate inactive regions, from C/M flaring active regions, and extremely active X-flare regions.
- We detect no clear boundaries in the reduced parameters between different levels of moderate flaring activity.

Corresponding author: Hanne Baeke, `hanne.baeke@kuleuven.be`

Abstract

We devised a new data analysis technique to identify the threat level of solar active regions by processing a combined data set of magnetic field properties and flaring activity. The data set is composed of two elements: a reduced factorization of SHARP properties of the active regions, and information about the flaring activity at the time of measurement of the SHARP parameters. Machine learning is used to reduce the data and to subsequently classify the active regions. For this classification we used both supervised and unsupervised clustering. The following processing steps are applied to reduce and enhance the SHARP data: outlier detection, redundancy elimination with common factor analysis, addition of sparsity with autoencoders, and construction of a balanced data set with under- and over-sampling. Supervised clustering (based on K-nearest neighbors) produces very good results on the strong X- and M-flares, with TSS scores of respectively 0.93 and 0.75. Unsupervised clustering (based on K-means and Gaussian Mixture Models) shows that non-flaring and flaring active regions can be distinguished, but there is not enough information in the data set for the technique to identify clear differences between the different flaring levels. This work shows that the SHARP database lacks information to accurately make flaring predictions: there is no clear hyperplane in the SHARP parameter space, even after a detailed cleaning procedure, that can separate active regions with different flaring activity. We propose instead, for future projects, to complement the magnetic field parameters with additional information, like images of the active regions.

Plain Language Summary

One of the main sources of space weather activity are solar active regions. In these zones the magnetic activity of the Sun is increased and can produce the two most energetic events in the solar system: flares and coronal mass ejections. We investigate the magnetic field properties of active regions, and the amount of energy they release. Our end goal is to produce an automatic model that can forecast the energy level released by a flare from solar active regions, using only their current magnetic field properties.

For this study, we used machine learning techniques that recognize patterns in data, without being explicitly told what to look for. These techniques can sometimes find patterns that escape the human intuition. The technique classifies different active regions, based on their magnetic properties, identifying those that can release large amounts of energy in the near future.

Our technique is able to discover differences between flaring and non-flaring active regions. But the data contains not enough information to predict how strong the energy releases will be. Therefore, improvement is still needed since we want to identify the strongest, most dangerous energy releases. Future research should incorporate other data types to get better results.

1 Introduction

Solar flares pose a serious threat to the near-Earth environment. They can produce streams of highly energetic particles, which can affect the Earth's magnetosphere within a few hours or minutes (Cinto et al., 2020). These particles pose radiation hazards to astronauts and spacecrafts (Mikaelian, 2009). Flares are also associated with radio communication disruptions (Knipp et al., 2016; Redmon et al., 2018), and the associated high energy particles can ionize our atmosphere at low altitudes (Liu et al., 2021). The largest flares are often accompanied by coronal mass ejections (CMEs). Kawabata et al. (2018) show that CMEs are associated with approximately all events whose X-ray flux is larger than $10^{-3.9} W m^{-2}$, which correspond to the X-flares. These CMEs can trigger geomagnetic storms, which can disable satellites (Dang et al., 2022) and even knock out electrical power grids (Pulkkinen et al., 2005). Should such a large storm happen nowadays, it would have catastrophic results, causing considerable economic damage. For example, the 1977 New York City blackout cost is estimated at $624 million dollars (Sorkin, 1982). A similar event today would have an even higher cost. Forecasting solar energetic activity is a critical topic in space weather research.

The differentiation of solar active regions very often involves the use of sunspot classifications - Mount Wilson (Hale et al., 1919) and McIntosh (McIntosh, 1990) - which are still performed manually. These classes are based on human observations in the visible light spectrum. This leads to inference of the subjectivity of the experts. Moreover, the visible light spectrum provides very limited information regarding the critical properties of solar active regions. Today it is possible to automatize the classification of solar active regions, reducing the influence of human bias. This will allow to produce fast solar flare forecasting systems.

This work focuses on the development of an unsupervised classification of solar active regions, using machine learning, and on their relation to their (non-)flaring activity. The classification is based on the SHARP parameters, extracted from SDO HMI observations of the magnetic field of active regions. A detailed processing of the SHARP data is performed to achieve the best possible results from unsupervised classification techniques. Therefore, these processing steps are also discussed with care throughout this paper.

There have been multiple previous attempts to build an automated classification of active regions. However, most of these studies tried to automate the existing McIntosh or Mount-Wilson classifications, e.g. (Colak & Qahwaji, 2008; Maloney & Gallagher, 2018; Nguyen et al., 2006; Smith et al., 2018). These studies applied machine learning on solar images, often combined with automatic sunspot detection. The machine learning methods used in the literature include neural networks, k-nearest neighbors, Support Vector Machines (SVMs), Random Forest and layered learning. In most cases, the percentage of correct classifications depends strongly on the specific class and on the amount of data available. The results of Colak and Qahwaji (2008) for example show results with a percentage of correct classifications between $\sim 40\%$ and $\sim 85\%$.

Housseal et al. (2019) performed unsupervised classification of sunspots, however, the authors did not use the magnetic field parameters: they used instead HMI magnetogram images to look for patterns in the sunspots connected to the active regions.

Recently, multiple papers have used the SHARP magnetic field parameters to construct solar flare prediction algorithms based on machine learning, e.g. (Abduallah et al., 2020; Bobra & Couvidat, 2015; Chen et al., 2019; Ilonidis et al., 2015; Jiao et al., 2020; Jonas et al., 2018; Liu et al., 2017; Ran et al., 2022; Sinha et al., 2022; Sun et al., 2022; Wang et al., 2020; Zhang et al., 2022). The methods used include Random Forest, MLPs, extreme learning machines, LSTMs, CNNs, SVMs, etc. Ilonidis et al. (2015) used time series of the SDO magnetic field data and constructed SVMs to forecast solar flares, which yielded a True Skill Score of 91%. Bobra and Couvidat (2015) also used SVMs on SHARP data, to distinguish between flare producing active regions and non-flare producing active regions. The authors did not include C-flares, which simplified the distinction between flaring and non-flaring active regions. Sun et al. (2022) focused on the prediction of M- and X-flares versus flare-quiet instances. They discarded all C-flares and lower from their data set. Jiao et al. (2020) took a different approach and applied machine learning on the SHARP parameters to identify the flare intensity, a continuous variable, instead of the discrete solar flare types.

A number of studies have investigated the importance of each of the SHARP parameters for solar flare prediction (Ran et al., 2022; Sinha et al., 2022; Zhang et al., 2022). They found that the most influential SHARP parameters are `TOTUSJH`, `TOTUSJZ`, `MEANPOT`, `TOTPOT`, `USFLUX` and `R_VALUE`. See Table 1 for the physical meaning of these parameters.

A new data set has been created by Bobra et al. (2021), called SMARPs. These are similar to SHARPs, but constructed from the solar images taken by MDI of SOHO. It attempts to extend backwards the SHARP database to the more active Solar Cycle 23. However, the SMARPs do not include as much information as the SHARPs and the data quality is lower (Sun et al., 2022).

Some studies combined the SHARP magnetic field parameters with features that are automatically generated from the solar images with machine learning methods, e.g (Chen et al., 2019; Jonas et al., 2018). Chen et al. (2019) compared the results of LSTM models trained on the SHARP data and on autoencoder-derived features and found that they were very similar. Therefore, the autoencoder-derived features could be a viable alternative for the SHARP parameters.

The goal of the present work is to classify the flaring activity of solar active regions, based only on the SHARP parameters extracted from the SDO HMI instrument. We apply rigorous and comprehensive pre-processing techniques to extract as much useful information as possible from the SHARP database. The results will inform us if there is enough information in the data to perform flare forecasts. While many of the classification methods used in the literature are based on supervised learning, we use unsupervised clustering to allow the computer to extract patterns unknown to the human experts. We show how the unsupervised classes that we obtain correlate with the flaring activity of active regions. In this work we also try to distinguish the different levels of flaring activity, whereas most studies are limited to the prediction of binary classes, only finding differences between flaring and non-flaring data.

The paper is structured as follows. Active regions and solar flares are briefly introduced in section 2. Section 3 discusses the data used, followed by section 4, which explains the data processing methods and results. Sections 5 and 6 introduce the clustering methods and types of evaluation. The clustering results are shown in section 7, followed by the discussion in section 8. Finally, section 9 summarizes the main conclusions of the research results.

2 Active Regions and Solar Flares

Solar active regions are large areas on the Sun where the magnetic activity temporarily and locally increases. The magnetic field there is complex and intense. Magnetic fields in active regions can be a thousand times stronger than the average solar magnetic field of a few Gauss (Sheeley, N.R., 2020). The number of active regions observed in the solar disk varies over the course of the solar cycle and are most common during its peak.

A solar flare is a sudden, intense brightening of a small area on the Sun, lasting minutes to a few hours. Flares occur in the solar corona when magnetic field lines of opposite polarity are forced together, by the convective motion of their foot-points in the convection zone, or by travelling coronal pressure waves. This causes magnetic reconnection, a sudden transformation of magnetic energy into kinetic and thermal energy. Streams of highly energetic particles travel along magnetic field lines, generating high intensity electromagnetic radiation on their path and during their interaction with matter. Solar flares typically erupt from solar active regions, because their complex and intense magnetic field is the perfect locus of magnetic reconnection (Priest & Forbes, 2002).

Flares are classified according to the strength of their soft X-ray emission, as recorded by the GOES satellites located in geostationary orbit. The following is a list of the flare classes in order of exponentially increasing magnitude: A, B, C, M and X. Strong solar flares occur very infrequently, compared to weak solar flares. Therefore, solar flare data is by definition largely imbalanced. This always has to be taken into account during the processing of the data and the interpretation of the results.

3 Data Set

The open source data set of Angryk et al. (2020b) is used for this research. The authors developed a data set (henceforth called the Angryk data set), extracted from the Space Weather HMI Active Region Patch series (SHARP) (Bobra et al., 2011), integrated with information from solar flare catalogs. These SHARP patches and their magnetic field parameters are derived from solar photospheric vector magnetograms obtained by the Helioseismic and Magnetic Imager (HMI) from the Solar Dynamics Observatory (SDO). The HMI instrument provides information on the magnetic field in the solar photosphere. These observations are bundled in patches for each active region. Magnetic field parameters are extracted from these patches and integrated over the whole area. They give an indication of the magnetic activity of the complete patch.

The Angryk data set contains sixteen SHARP parameters and eight additional parameters proposed by Angryk et al. (2020a). These 24 parameters are listed in Table 1. The data set also contains parameters `BFLARE`, `CFLARE`, `MFLARE` and `XLFARE`. These express the number of flares of each flare class occurring at the time of measurement of the SHARP and therefore indicate the concurrent solar flare activity of that active region. For simplicity, in this work, each data point has been assigned to only one of four classes:

No-flare, C-flare, M-flare or X-flare. These correspond to the strongest occurring flare originating from the active region at that time. The No-flare class signifies the flare-quiet instances, but also the weakest, A- and B-class, flares. This because the A- and B-flares are hard to distinguish against the background brightness of the Sun (Chen et al., 2019). The assignment of flare types to the data points leads to the following ratio: 2 602 509 No-flares, 6717 C-flares, 680 M-flares and 47 X-flares. The data was collected between May 2010 and December 2018. This corresponds with solar cycle 24 (December 2008 - December 2019) and includes the solar maximum in April 2014. This solar cycle was an unusual quiet one, and the data set contains only few strong flares. The Angryk data set is meant to serve as a benchmark data set for testing flare prediction algorithms (Angryk et al., 2020a).

4 Data Processing

Some pre-processing of the data set was already carried out by Angryk et al. (2020a). Further processing includes outlier removal, data transformation and dimensionality reduction. These steps are explained in more detail in the following sections.

There is a large class imbalance present in the data set, with 2 602 509 No-flares, 6717 C-flares, 680 M-flares and only 47 X-flares. This class imbalance needs to be taken into account when processing the data. To reduce the impact of class imbalance, in this work the No-flare class is randomly under-sampled to 50 000 No-flares. This is done by randomly selecting 50 000 data points from the 2 602 509 No-flares, without selecting the same data point twice.

The selected number of No-flares is determined after multiple tests of the autoencoding procedure, described in section 4.3.2, the most data-intensive processing step in this work. In short, in an autoencoder a compression and decompression of the data set is performed, and the active region properties before and after the procedure should be exactly the same. We applied the procedure with different sample sizes. For each case the error is computed. When the sample size is too small, the error is large. Increasing the size of the sample reduces the error. A plot of the sample size versus the error presents an optimal inflection point, which in this work corresponds to the selected sample size: 50 000 data points are sufficient to obtain an accuracy comparable to the full 2 602 509 data points.

In section 4.4 we show how we handle additional class imbalances using over- and under-sampling techniques.

Table 1: Magnetic field parameters from Angryk et al. (2020b). Parameters with * are derived by Angryk et al. (2020a), the others are contained in SHARP. Units from Liu et al. (2017) and SDO.

Parameters	**Description**	**Formula**
`ABSNJZH` [$10\text{G}^2/\text{m}$]	Absolute net current helicity	$H_{c_{abs}} \propto \lvert \sum B_z \cdot J_z \rvert$
`EPSX`* [-10^{-1}]	Sum normalized Lorentz force (X)	$\delta F_x \propto \frac{\sum B_x B_z}{\sum B^2}$
`EPSY`* [-10^{-1}]	Sum normalized Lorentz force (Y)	$\delta F_y \propto \frac{-\sum B_y B_z}{\sum B^2}$
`EPSZ`* [-10^{-1}]	Sum normalized Lorentz force (Z)	$\delta F_z \propto \frac{\sum (B_x^2 + B_y^2 - B_z^2)}{\sum B^2}$
`MEANALP` [1/Mm]	Mean twist parameter	$\alpha_{total} \propto \frac{\sum J_z \cdot B_z}{\sum B_z^2}$
`MEANGAM` [°]	Mean inclination angle	$\overline{\gamma} = \frac{1}{N} \sum \arctan\left(\frac{B_h}{B_z}\right)$
`MEANGBH` [G/Mm]	Mean horizontal field gradient	$\overline{\nabla B_h} = \frac{1}{N} \sum \sqrt{\left(\frac{\partial B_h}{\partial x} + \frac{\partial B_h}{\partial y}\right)}$
`MEANGBT` [G/Mm]	Mean total field gradient	$\overline{\nabla B_{tot}} = \frac{1}{N} \sum \sqrt{\left(\frac{\partial B}{\partial x} + \frac{\partial B}{\partial y}\right)}$
`MEANGBZ` [G/Mm]	Mean vertical field gradient	$\overline{\nabla B_z} = \frac{1}{N} \sum \sqrt{\left(\frac{\partial B_z}{\partial x} + \frac{\partial B_z}{\partial y}\right)}$
`MEANJZD` [mA/m^2]	Mean vertical current density	$\overline{J_z} \propto \frac{1}{N} \sum \left(\frac{\partial B_y}{\partial x} - \frac{\partial B_x}{\partial y}\right)$
`MEANJZH` [G^2/m]	Mean current helicity	$\overline{H_c} \propto \frac{1}{N} \sum B_z \cdot J_z$
`MEANPOT` [10^3ergs/cm^3]	Mean photospheric excess magnetic energy density	$\overline{\rho} \propto \frac{1}{N} \sum (\mathbf{B}^{Obs} - \mathbf{B}^{Pot})^2$
`MEANSHR` [°]	Mean shear angle	$\overline{\Gamma} = \frac{1}{N} \sum \arccos\left(\frac{\mathbf{B}^{Obs} \cdot \mathbf{B}^{Pot}}{\lvert B^{Obs}\rvert \lvert B^{Pot}\rvert}\right)$
`R_VALUE`* [Mx]	Total unsigned flux around high gradient polarity inversion lines	$\phi = \sum \lvert B_{los}\rvert \cdot dA$ (within R mask)
`SAVNCPP` [10^{12}A]	Summed absolute value of net current per polarity	$J_{\Sigma z} \propto \left\lvert \sum^{B_z^+} J_z dA \right\rvert + \left\lvert \sum^{B_z^-} J_z dA \right\rvert$
`SHRGT45` [%]	Area with shear angle > 45°	$\frac{\text{Area with Shear} > 45°}{\text{Total Area}}$
`TOTBSQ`* [10^{10}G^2]	Total magnitude of Lorentz force	$F \propto \sum B^2$
`TOTFX`* [-10^{23}dyne]	Sum X-component of Lorentz force	$F_x \propto \sum B_x B_z dA$
`TOTFY`* [-10^{23}dyne]	Sum Y-component of Lorentz force	$F_y \propto \sum B_y B_z dA$
`TOTFZ`* [-10^{23}dyne]	Sum Z-component of Lorentz force	$F_z \propto \sum \left(B_x^2 + B_y^2 - B_z^2\right) dA$
`TOTPOT` [10^{23}ergs/cm^3]	Total photospheric magnetic energy density	$\rho_{tot} \propto \sum \left(\overrightarrow{\mathbf{B}^{Obs}} - \overrightarrow{\mathbf{B}^{Pot}}\right)^2 dA$
`TOTUSJH` [$10^2\text{G}^2/\text{m}$]	Total unsigned current helicity	$H_{c_{total}} \propto \sum B_z \cdot J_z$
`TOTUSJZ` [10^{12}A]	Total unsigned vertical current	$J_{z_{total}} = \sum \lvert J_z \rvert dA$
`USFLUX` [10^{21}Mx]	Total unsigned flux	$\phi = \sum \lvert B_z \rvert dA$

4.1 Outlier Removal

Multiple entries in the data set contain one or more empty properties (NaN values). We eliminate from the original data set every entry where at least one of the properties was empty. We also perform a detection and elimination of outliers. These were identified using the hierarchical clustering algorithm HDBSCAN. This method is able to automatically choose the optimal clustering of a cloud of points in an N-dimensional space. The points that are detached from the core cloud of points are identified as outliers. A more detailed explanation of HDBSCAN can be found in Campello et al. (2013).

With this technique 586 outliers were found. About 20% of the outliers come from HMI magnetogram images taken during rotation or re-positioning of the SDO spacecraft, causing distortions in the data.

In addition, 36 outliers were identified and removed by hand. Thirty-three of these additional outliers were due to the same parameter, `MEANPOT`. The other three were due to the parameter `TOTFZ`. The fact that they were missed by HDBSCAN is probably due to a combination of the standardization and some extreme outliers. The standardization transforms the data to zero mean and to unit variance. If there are a few extreme outliers, this will shift the majority of the data to very small values. Because this is not the case for the other parameters, there is a difference of $\sim 2-3$ orders of magnitude, which hinders HDBSCAN to detect all outliers.

4.2 Data Transformation

To be able to differentiate groups of points in the parameter space, it is necessary to identify high concentrations of points that can be separated by a hyper-plane. An initial visual inspection of the distribution function of each one of the parameters can show if there are peaks and valleys in the distribution that clearly separate active regions with different properties. Some of the parameters have a very small spread of values among all the active regions. Unsupervised clustering techniques have difficulties identifying multiple clusters in unimodal distributed parameters, since this would only lead to one cluster. We applied transformations to some of the parameters to perform a rebinning of the data distributions. This is one of the procedures known in machine learning as 'feature engineering'. The transformations used are listed in Table 2.

Figure 1 shows the difference a good transformation can make, and how this can improve clustering. After a logarithmic transformation two peaks are visible, while before there is only one very large one.

Table 2: Data transformations used to expand some very narrow distributions.

Parameter (Table 1)	**Transformation**
TOTUSJH	$\ln(x + \lvert\min(x)\rvert + 0.01)$
TOTBSQ	$\ln(x + \lvert\min(x)\rvert + 0.01)$
TOTPOT	$\ln(x + \lvert\min(x)\rvert + 0.01)$
TOTUSJZ	$\ln(x + \lvert\min(x)\rvert + 0.01)$
ABSNJZH	$\ln(x + \lvert\min(x)\rvert + 0.01)$
SAVNCPP	$\ln(x + \lvert\min(x)\rvert + 0.01)$
USFLUX	$\ln(x + \lvert\min(x)\rvert + 0.01)$
MEANPOT	$\ln(x + \lvert\min(x)\rvert + 0.0001)$
TOTFZ	$\ln(-x + \lvert\max(x)\rvert + 0.01)$
TOTFY	$\ln(\lvert x\rvert)$
TOTFX	$\ln(\lvert x\rvert)$

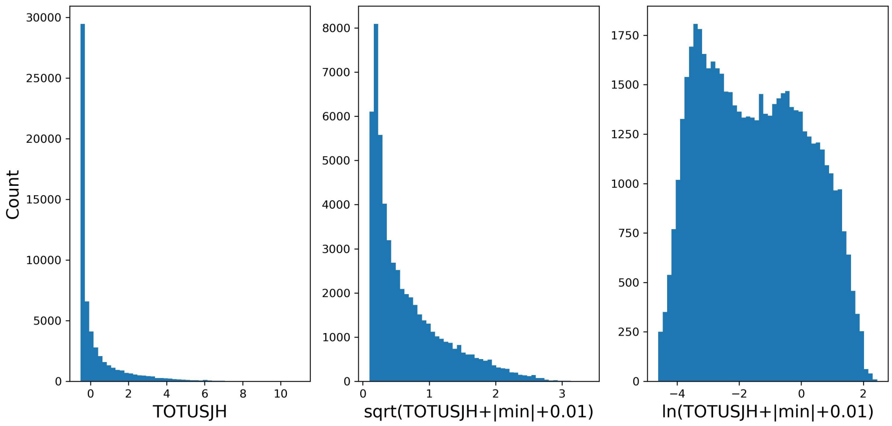

Figure 1: Example of two transformations of the parameter `TOTUSJH` (left). While the root squared transformation produces a better coverage of the distribution (centre), the transformation of the bins with the natural logarithm (right) yields a distribution more useful for clustering.

4.3 Dimensionality Reduction

High-dimensional data is computationally expensive to process. If possible, it is important to reduce the number of dimensions. In addition, clustering methods and other techniques based on the calculation of distances in an Eulerian space are subject to the 'curse of dimensionality': in high dimensions every point tends to be equidistant to each other point. Moreover, we want to reduce high correlations by removing redundant features. Figure 2 (left) illustrates the presence of correlations between the magnetic field parameters. This is not surprising, since they often depend on the same magnetic coefficients, e.g. $\mathbf{B}_z$ and $\mathbf{J}_z$ (see Table 1). These redundant features do not add any relevant information and may hinder the learning algorithm, possibly causing overfitting (Yu & Liu, 2004). To mitigate this problem, we applied Common Factor Analysis (Spearman, 1904) (CFA) to our data set.

4.3.1 Common Factor Analysis

Common Factor Analysis (CFA) is a technique which searches for latent, unobserved variables, called factors, from a set of observed variables. The package `FactorAnalyzer` of (Biggs, 2019) is used. The number of factors is determined with the help of *Horn's Parallel Analysis* (Horn, 1965). Figure 2 (right) shows the resulting factor loadings, a measure of how much a factor explains the associated magnetic field parameters. The first factor has high explanatory power for multiple magnetic field parameters, which confirms that many of these parameters are inter-correlated. Calculation of the covariance of the selected five factors confirms that they show zero covariance with each other.

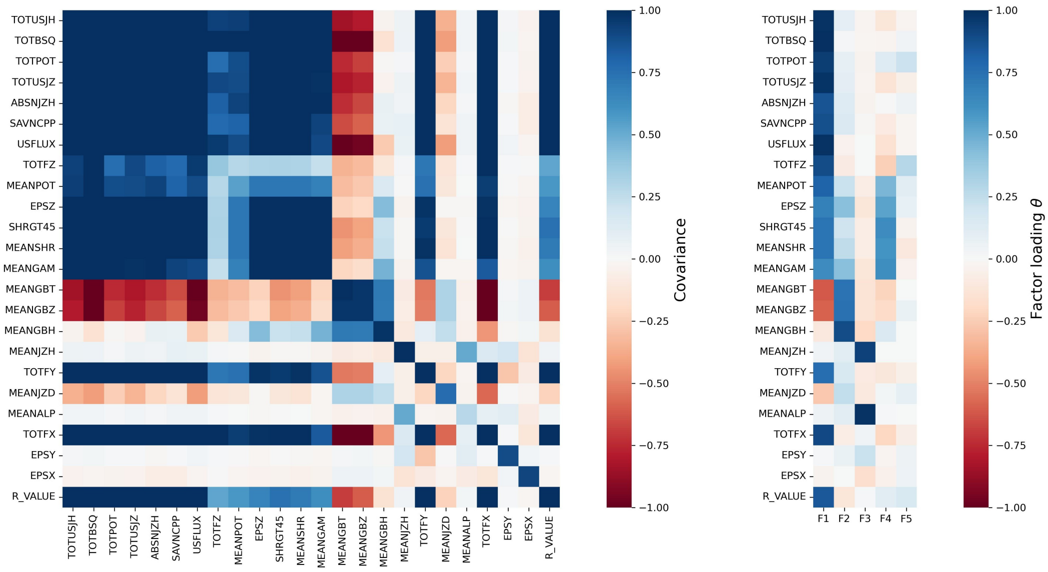

Figure 2: Left: Covariance matrix of the data set before applying CFA on it. A lot of the parameters are strongly correlated with each other. Right: Heatmap of factor loadings of CFA.

4.3.2 Sparse Autoencoders

Makhzani and Frey (2014) shows improvement in classification tasks when sparse data representations are used. To improve sparsity in our data set, we applied an additional data processing step. Sparse autoencoders are able to transform the data into a higher dimensional space, where it is possible to create hyperplanes that allow to separate different clusters of points.

Sparse autoencoders are a special kind of unsupervised neural networks. For an explanation on neural networks, we refer the reader to the notes of Ng et al. (2011). The underlying mathematics of autoencoders are the same as for neural networks. The special property of autoencoders is that the target values ($\hat{X}$) are set equal to the input values (X) (Hinton & Salakhutdinov, 2006): $f : X \rightarrow \hat{X}$, where $X \approx \hat{X}$. The model learns an approximation of the identity function. This may seem like a trivial task, but by placing constraints on the network interesting structures can be discovered.

In a basic (vanilla) autoencoder, also called encoder-decoder, $AE = \{f, f'\}$, the applied constraint consists to limit the number of nodes in an intermediary hidden layer to less than the number of input features of the model: the autoencoder functions are defined as $f : X \in \mathbb{R}^n \rightarrow Z \in \mathbb{R}^m$, followed by $f' : Z \in \mathbb{R}^m \rightarrow \hat{X} \in \mathbb{R}^n$, where $n > m$. A second autoencoder category corresponds to sparse autoencoders (Jiang et al., 2015), where the constraint is applied by forcing sparsity in the intermediary hidden layer. In this case the dimension of the hidden layer does not have to be smaller than the input layer. This sparsity constraint ensures that only a few hidden nodes are allowed to be active at the same time, i.e. most of the hidden nodes will have a value of zero. Sparse autoencoders provide an information bottleneck without having to reduce the number of nodes. This also means that low dimensional data sets can be projected into a higher dimension where sparsity is encouraged, allowing for a better differentiation between different classes.

4.3.2.1 Implementation Details The sparse autoencoder is implemented using Python, together with libraries `Tensorflow` (Abadi et al., 2015) and `Keras` (Chollet et al., 2015). Any kind of neural network learns by minimizing a cost, or loss function, obtained by comparing the output of the model with the expected output. The loss function, Eq. 1, consists of two terms: (1) a reconstruction error and (2) a sparsity penalty. As reconstruction error the mean squared error is used. The sparsity penalty is a regularization acting on the outputs of individual neural network nodes in the hidden layer. It penalizes the activation of the hidden nodes, $a_i^{(h)} \in Z$, using the L1-norm. In the sparsity term of Eq. 1, λ is the pre-factor that determines the influence of the sparse regularization.

$$L = \frac{1}{n}\sum_i (X_i - \hat{X}_i)^2 + \lambda \sum_i \left| a_i^{(h)} \right| \tag{1}$$

The autoencoder is optimized following the traditional error minimization techniques used in classical neural networks. The optimization algorithm that we selected is the `Adam` (Kingma & Ba, 2015) technique. This is an extension to stochastic gradient descent that maintains separate learning rates for each parameter.

To determine the accuracy of the output the R-squared metric, Eq. 2 is used:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(X_i - \hat{X}_i)^2}{\sum_{i=1}^{N}(X_i - \overline{X_i})^2} \text{ with } \overline{X_i} = \frac{1}{N}\sum_{j=1}^{N} X_j \tag{2}$$

To reduce the influence of the class imbalance, different weights have been assigned to the data samples corresponding to different flare classes. A weight of respectively 1, 4, 16 and 64 has been assigned to classes No-flare, C-flare, M-flare and X-flare.

In the Adam optimization algorithm one of the hyperparameters is the learning rate. This hyperparameter influences the speed at which the model converges towards the minimum loss. The optimal learning rate is determined using the method introduced by Smith (2017). This method trains a network starting with a low learning rate, which is exponentially increased throughout the epochs (training cycles). The optimal learning rate corresponds to the fastest decrease in loss throughout the training. An additional method to determine the optimal learning rate is to run the algorithm for multiple values of the learning rate for a limited number of epochs, and to select one with the lowest validation loss. In our work, the combination of these two optimization methods yields an optimal learning rate of 0.0005.

Our data set is split into three sub-groups: 60% training, 20% validation and 20% testing data. The split is performed using stratification, which means that in each data portion the percentage of each flare type is preserved.

4.3.2.2 Architecture Optimization To find the optimal autoencoder architecture, three parameters need to be optimized: (1) the magnitude λ of the sparsity constraint, (2) the number of hidden nodes and (3) the activation function.

If the sparsity pre-factor is too high, all hidden nodes will tend to produce values of zero; if this parameter is too small, no sparsity will be introduced. The optimal value of λ is obtained by finding a balance between the level of sparsity and the activity on the hidden nodes. The pre-factor needs to be set to ensure that only part of the nodes (less than the number of input nodes) are active at the same time, without leaving inactive nodes. This balance is found for $\lambda = 0.1$.

The most adequate architecture is selected by comparing the loss function between the training and the validation set. The optimal architecture contains one hidden layer with seven hidden nodes and uses SELU (Klambauer et al., 2017) activation function.

4.3.2.3 Resulting Distributions The resulting optimal sparse autoencoder is used to increase the dimensionaliy, generating sparsity in the data set. The R-squared metric returns a value of 0.9942, indicating that the model is able to nearly perfectly mimic the original distributions. A two-dimensional projection of the distribution of each pair of parameters in the final data set is shown in Figure 3. This higher dimensional encoding of the data will be used for clustering in later sections.

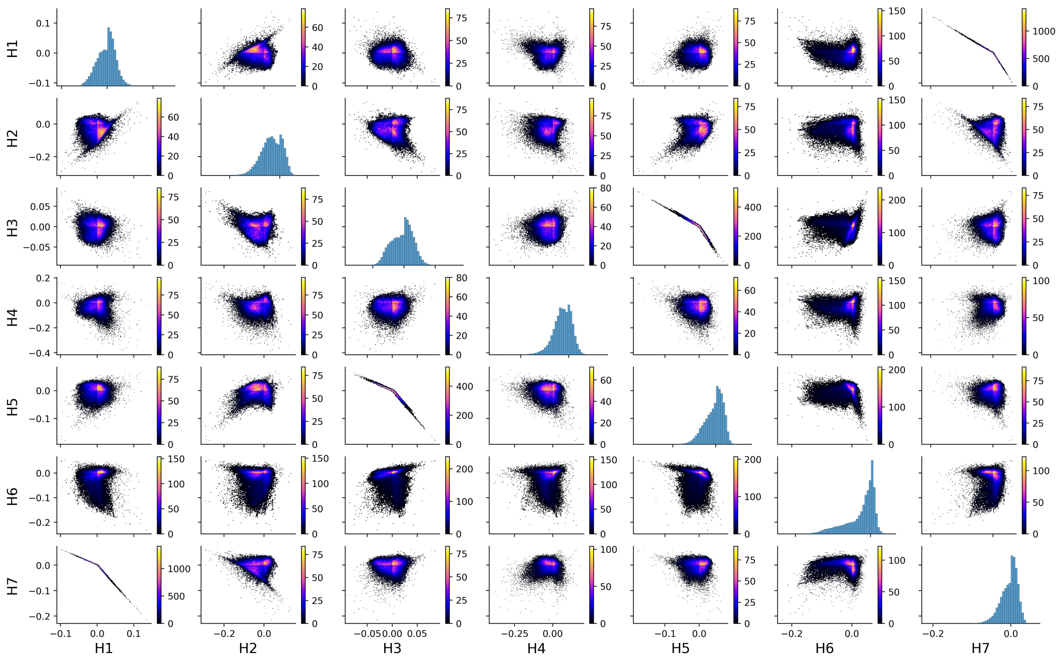

Figure 3: Distributions of the encoded data produced by the hidden layer of the sparse autoencoder. The autoencoder includes one hidden layer, with seven neurons, and SELU activation functions. The pre-factor λ for the activity regularization is set to 0.1.

4.4 Data Sampling

Solar flare data is by definition largely imbalanced, since strong solar flares are scarce, affecting the classification results. Machine learning methods tend to favor the dominant class, which in our case corresponds to the non-flaring active regions. The four different flare activity classes are either over-sampled or under-sampled to construct a balanced data set with a similar amount of data points per flare class. A random under-sampling of the No-flares was already presented in section 4, but the imbalance among flare classes is still large.

4.4.1 Random Sampling

Random sampling can be applied to either under-sample or over-sample data. The methods `RandomUnderSampler` and `RandomOverSampler` of the package `imbalanced-learn` (Lemaître et al., 2017) are used. Random under-sampling picks samples from the majority classes without replacement, while over-sampling picks samples from the minority classes with replacement. However, random over-sampling of the minority class can lead to duplication, which might lead to overfitting. Therefore an alternative over-sampling method is used.

4.4.2 SMOTE Sampling

The alternative Synthetic Minority Over-sampling TEchnique (SMOTE) (Chawla et al., 2002) technique is also included in the `imbalanced-learn` package. SMOTE does not duplicate any samples, but generates new data points by randomly selecting a minority class instance (a), and then finding its k nearest neighbors. Subsequently, one of those k neighbors (b) is chosen at random and a synthetic example is created at a random point on the line segment between the instance (a) and its selected neighbor (b).

4.4.3 Resulting Data Set

It has been shown by Chawla et al. (2002) that the combination of SMOTE and under-sampling performs better than plain under-sampling. In our work the majority classes, No-flare and C-flare, are randomly under-sampled, while the minority classes, M-flare and X-flare, are over-sampled with SMOTE. Every class is sampled to 6000 samples, making the data set balanced.

5 Clustering

We tested multiple clustering algorithms on the data set to classify the solar active regions based on their processed magnetic field parameters and found common aspects among the corresponding active regions.

Clustering is a machine learning method which groups data in subgroups that share similar properties (in our case, similar reduced magnetic field parameters). A good clustering method minimizes the intra-cluster distances, while maximizing inter-cluster distances (Zhang & Tsai, 2005). The implementation and the way clusters are defined differ from method to method. Every method that is considered here is implemented with the `scikit-learn` package.

5.1 k-Nearest Neighbors (supervised)

k-Nearest Neighbors (KNN), explained in e.g. Cunningham and Delany (2007), is a supervised and instance-based clustering algorithm. It assumes similar objects exist in close proximity to the evaluated data point. The class of a data point is determined based on the most frequent class among its k nearest neighbors.

The optimal number of neighbors k is the one that minimizes the error, the percentage of wrong predictions, while maintaining the ability to make accurate predictions on new data. The method minimizes the loss on the validation data, without overfitting on the training data. In general, lower k makes the predictions less stable. Increasing the number of neighbors makes the predictions more stable due to averaging and therefore more likely to produce reliable results. We selected the optimal k by performing the KNN algorithm for a range of k-values, fitting a fourth order polynomial to the corresponding error values and selecting the k corresponding to the minimum error.

5.2 K-means (unsupervised)

K-means (Lloyd, 1982; MacQueen, 1967) is an unsupervised, centroid-based clustering method and assumes that the clusters are spherical and equally sized. The method works best when the clusters are equally dense and not too contaminated by noise or outliers. The clustering is achieved by iteratively assigning each data point to its nearest centroid and creating new centroids by computing the mean of each cluster.

The optimal number of clusters is determined by a *scree* plot (Cattell, 1966), where the 'knee' point is associated to the optimum value, and corresponds to the inflection point of the curve. The position of this 'knee' is determined through the *Kneedle algorithm* (Satopaa et al., 2011). The scree plot is configured by computing the error for different runs for a range of different number of clusters. A line is plotted between the first and last point of the curve and the distances between each point and the line are computed. The point with maximal distance between the two lines marks the maximum of curvature, i.e. the elbow.

5.3 Gaussian Mixture Models (unsupervised)

Gaussian Mixture Models (GMM) assume that all data points are generated from a mixture of Gaussian distributions and identifies for each data point the probabilities of belonging to each of the Gaussian distributions. This method allows the detection of more elongated clusters. The Gaussian distributions are approximated by the Expectation-Maximization method (Dempster et al., 1977). The GMM is a probabilistic method.

To determine the number of clusters for GMM, several methods can be used. We chose to use the gradient of the Bayesian Information Criterion (BIC). BIC (Schwarz, 1978) gives an estimation on how accurately the model represents the existing data, with lower BIC value indicating a better estimation. BIC is defined in Eq. 3, with k the number of unknown model parameters (mean and variance for each cluster), n the number of samples and $\hat{L}$ the maximum likelihood.

$$BIC = k \ln n - 2 \ln \hat{L} \tag{3}$$

A high number of clusters corresponds to low BIC scores, but the error curve shows an inflection point. This point can be found by checking the gradient of BIC. The optimal number of clusters is the point where the gradient no longer changes, i.e. when the second derivative is zero (Lavorini, 2018).

6 Evaluation Methods

To determine the quality of a clustering method a good evaluation method is essential. An Area Under the Curve Receiver Operating Characteristics (AUC-ROC) plot (Fawcett, 2006) is a good evaluation technique for supervised classification methods, when the data is severely imbalanced (Brownlee, 2020).

ROC curves are in general used in binary classifications, but can be extended to multi-class data by using one-vs-rest for each class, which provides one ROC curve per class. The macro-average can be computed by taking the average of all ROC curves, treating all classes equally.

The ROC curve is a visual measure of the predictive quality of the model, that visualizes the trade-off between sensitivity and specificity. The plot of a ROC curve displays the True Positive Rate (TPR), see equation 4, on the y-axis and the False Positive Rate (FPR), see equation 5, on the x-axis. These rates are computed for different thresholds. The threshold is the lowest probability necessary to be assigned to the positive cluster.

$$TPR = \frac{TP}{TP + FN} \tag{4}$$

$$FPR = \frac{FP}{TN + FP} \tag{5}$$

An AUC score can be computed from the ROC, by computing the area under the curve. AUC is a measure of the ability of a classifier to distinguish between classes, where e.g. 0.7 means that in 70% of the cases the model is able to distinguish between the positive and the negative class (Narkhede, 2018).

In addition, the True Skill Statistic, also called the Hanssen score (Hanssen & Kuipers, 1965), will be computed for the supervised clustering, see equation 6. The value of TSS lies between -1 and 1, with a higher value indicating a better forecast. This is one of the most used evaluation metrics to assess solar flare forecasts.

$$TSS = \frac{TP}{TP + FN} - \frac{FP}{FP + TN} = \frac{TP}{P} - \frac{FP}{N} \tag{6}$$

It is a lot harder to assess whether unsupervised clustering methods perform well, because no labels are present. A viable alternative are validation methods that check whether there is a high separation between clusters and a high cohesion within the clusters. Examples of such metrics are the Calinsky-Harabasz (CH) coefficient (Caliński & Harabasz, 1974) and the Silhouette coefficient (SC) (Rousseeuw, 1987). The Calinski-Harabasz coefficient is defined as the ratio between the within-cluster dispersion and the between-cluster dispersion. This coefficient should be maximized. The Silhouette coefficient is computed, for each sample, using: (a) the mean inter-cluster distance, and (b) the mean nearest-cluster distance. The formula is given in equation 7. The final Silhouette score is found by computing the mean over all samples. The best value is 1, the worst is -1 and values near 0 indicate that the clusters overlap. If the value is negative it is generally an indication that samples are assigned to the wrong cluster, as it is found that a different cluster is more similar.

$$SC = \frac{b - a}{max(a, b)} \tag{7}$$

7 Results

Figure 4 shows the mean value and standard deviation of each of the seven reduced parameters, for each flare class. In general, the parameters are very similar for all flaring active regions (C, M and X-flares). X-flare classes present only slight differences with respect to the other flaring classes. Parameters H2, H5 and H6 have a larger absolute mean value for these stronger flare classes. The mean value of the data without flares

(No) is clearly different. It can be expected that flaring active regions will be distinguishable from non-flaring active regions, while distinguishing between the different flare classes may be more challenging with the available data.

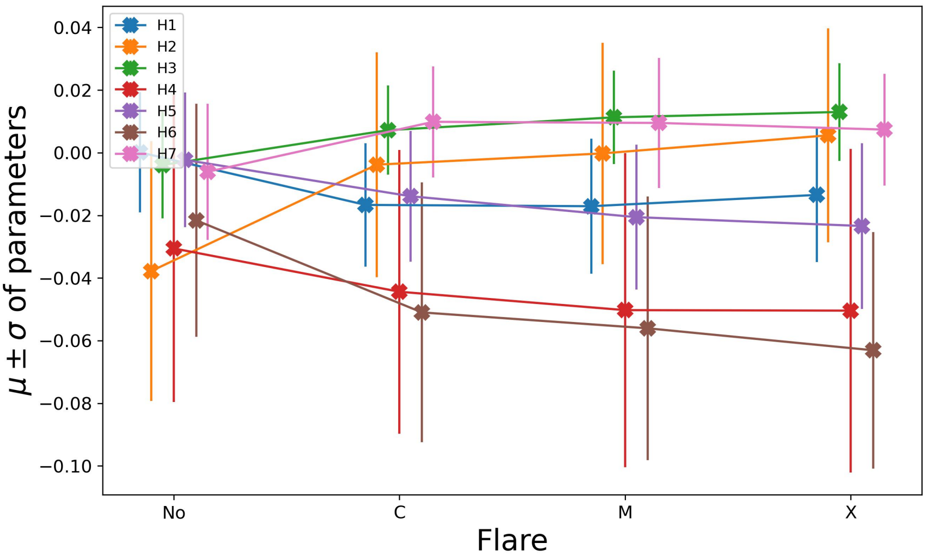

Figure 4: Mean and standard deviation of the features resulting from the sparse autoencoder, per flare label. The flaring data looks very similar, while the non-flaring data has distinct parameter values.

7.1 Supervised (KNN)

In our work the hyperparameter selection for KNN was based on the data set before the sampling procedure used in section 4.4, to avoid using under-/over-sampled data points. Performing the hyperparameter selection on the sampled data yields an optimal number of neighbors of one, which leads to unstable results. By applying the hyperparameter selection on the data set before sampling, we find an optimal number of neighbors of ten. To validate this selection method, the KNN clustering is conducted multiple times, testing the use of one, three, six and ten nearest neighbors. The resulting ROC curves are shown in Figure 5. These figures show that when more neighbours are taken into account for the clustering, the results improve, producing a higher value for the area-under-the-curve. This is the case for the macro-average along the whole data set, as well as for the individual flare types. This shows that taking only one neighbor into account would not have been optimal. The differences between the results with three, six and ten neighbors are not too large.

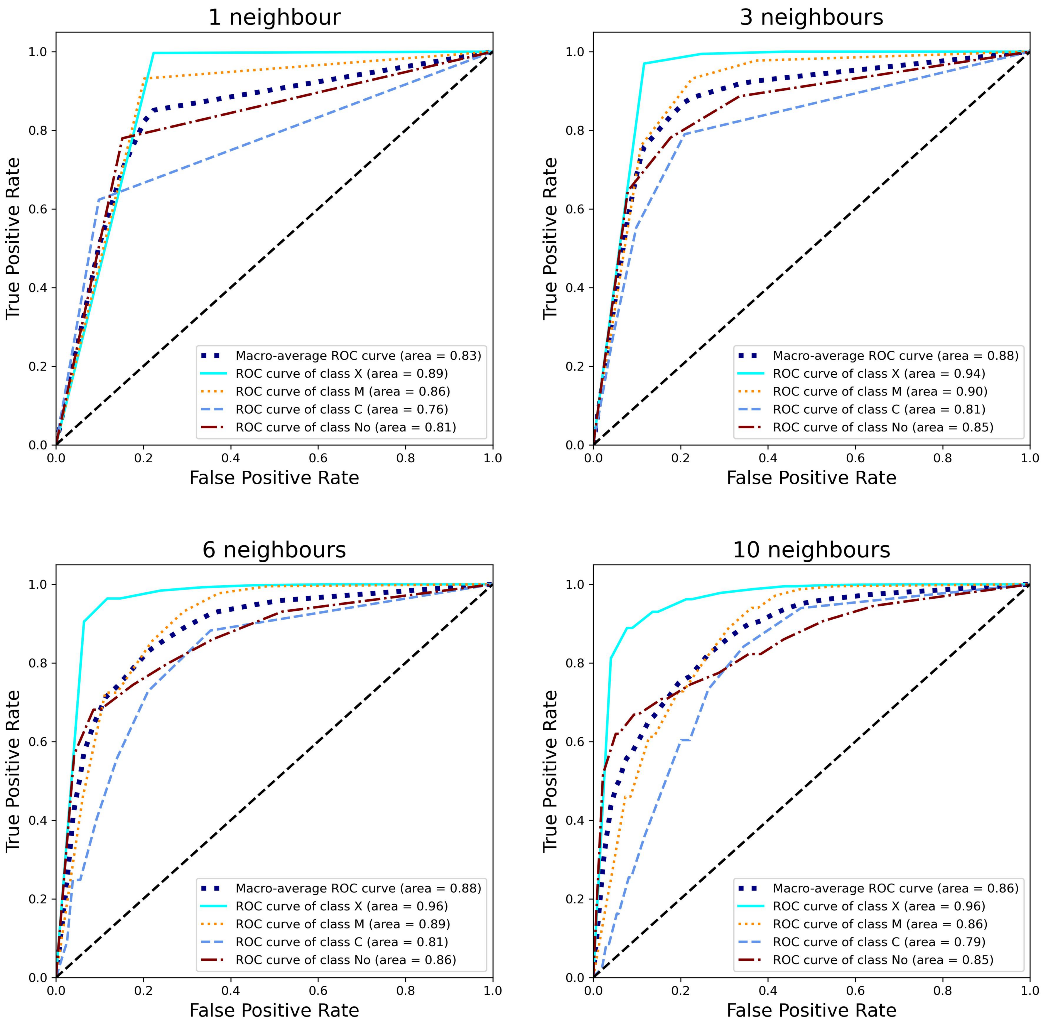

Figure 5: AUC ROC plot of the results of KNN, performed on the sampled data set, for varying number of neighbors.

Figure 6 shows the normalized confusion matrices for the clustering of KNN. On the x-axis the figure shows the predictions and on the y-axis the true classes. On the left panel we present the results using one nearest neighbor, and on the right panel the result when ten neighbors are considered. The largest difference is observed in the number of C-flares that are classified correctly. When more neighbors are taken into account, the C-flares are more often misclassified as larger M- and X-flares. On the other hand, when more neighbors are taken into account, C-flares are less often misclassified as non-flaring. The fact that the C-flares are more often misclassified as stronger flares is not necessarily a bad thing. For flare prediction, we are most interested in recognising the strongest flares. Therefore, it could be considered better to have a prediction method that is more likely to overestimate the strength of a flare, than to underestimate the strength of a flare. However, false warnings will lessen the trust of the industry in flare predictions, so ideally we want to minimize both the false positives and the false negatives.

The percentage of true positives for each flare type is higher when only one neighbor is taken into account versus when ten neighbors are taken into account. While the results with one neighbor might look better on this figure, they are unstable and more influenced by the artificial data introduced by the sampling.

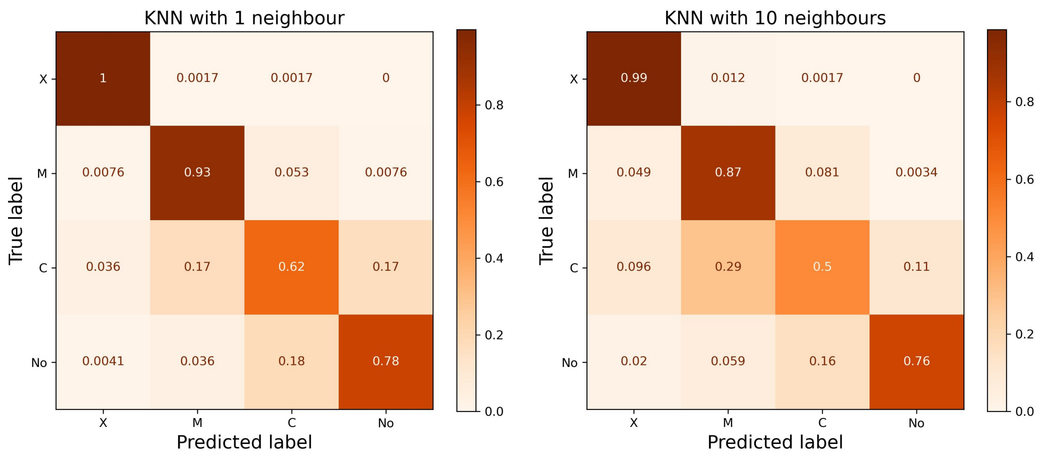

Figure 6: Normalized confusion matrices of the results of KNN with (left) only one nearest neighbor and (right) ten nearest neighbors taken into account.

Focusing on the confusion matrix in the right panel of Fig. 6, the following conclusions can be made: almost all of the X-flares are correctly identified. However, this is probably influenced by the over-sampling of the X-flares by a factor of approximately 160. 87% of the true M-flares are correctly identified. This high percentage is also somewhat influenced by the over-sampling. When M-flares are misclassified, it is $\sim$ 37% of the time as an X-flare and $\sim$ 61% of the time as a C-flare. 76% of the non-flaring active regions are correctly classified as well. This is quite a good result, considering that this class is largely under-sampled. The non-flaring active regions are most of the time mistaken for C-flares. Finally, the C-flares turn out to be hardest to distinguish, with only 50% of the active regions correctly identified as C-flares. They are $\sim$ 58% of the time overestimated as M-flares, $\sim$ 19% of the time as X-flares and $\sim$ 22% of the time underestimated as non-flaring. The flares are mostly mistaken for their neighboring classes, in terms of X-ray flux strength. This indicates that the clusters are partly overlapping.

The TSS has been calculated for each of the flare types separately. A TSS of 0.93 is found for the X-flares, 0.75 for the M-flares, 0.42 for the C-flares and 0.72 for the non-flaring active regions.

7.2 Unsupervised (K-means + GMM)

Unsupervised clustering methods are more useful in practice, since there is not always information present about the flaring nature of an active region. These methods do not take into account the information about the X-ray flux, but only the reduced magnetic field parameters. For both unsupervised methods used in this work (K-means and GMM) the number of clusters needs to be determined using a hyperparameter optimization technique, as described in sections 5.2 and 5.3. For K-means an optimal number of four (4) clusters is found, while GMM has an optimal number of three (3) clusters.

Table 3 shows the Calinski-Harabasz (Caliński & Harabasz, 1974) and Silhouette (Rousseeuw, 1987) coefficients, which evaluate the clusters found through K-means and GNN. The first one should be maximized, while the latter should be as close to 1 as possible. Both coefficients indicate that K-means does a better job at clustering the data. However, a relatively low Silhouette score of 0.25 indicates that the clusters are either not very well separated or the points within a cluster are distributed relatively far apart. The possibility that the clusters are overlapping was already mentioned in the previous section.

Table 3: Evaluation coefficients for K-means and GNN.

	K-means	**GMM**
Calinski-Harabasz	7506	1886
Silhouette	0.25	0.12

With unsupervised machine learning methods no confusion matrix can be constructed, since no labels are used. However, we have already access to the expected flare classification in the data set. These values are not used to train the unsupervised clustering algorithms. We used this information to evaluate the accuracy of the automatic unsupervised classification with respect to the expected flare classes. The resulting visualization is shown in Figure 7, where for each of the two clustering algorithms the percentage of each flare included in each of the clusters is shown. Normalization is performed per flare type.

Analyzing the clusters of K-means learns us that 66% of the non-flaring active regions are included in Cluster 3. Cluster 3 also includes 17% of the C-flares, 12% of the M-flares and 5% of the X-flares. This cluster can be considered as one with mostly non- and weakly-flaring active regions. If an active region is classified in Cluster 3, chances are thus relatively low that it is a strong flare. Clusters 1, 2 and 4 contain less non-flaring active regions, respectively 14%, 7% and 12%. They do contain more of the flaring active regions. Cluster 2 contains $\sim$ 40% of each of the flare types. Cluster 4 contains $\sim$ 40% of the X-flares and only $\sim$ 20% of the C- and M-flares. Cluster 1 also contains flaring active regions, with more C- and M-flares than X-flares. Since all four clusters contain a significant fraction of all four flare types, there is no way to determine with certainty the type of flare, based on this clustering of the active regions. What one could conclude from these results is that an active region that is classified in Cluster 3 is most likely to be non-flaring or weakly flaring. On the other hand, an active region that is classified in Cluster 4 has a higher probability to be an X-flare, since these are most abundantly present. If an active region is classified in Cluster 2, it is very probable to be flaring, but nothing can be concluded about the type of flare. Finally, if an active region is classified in Cluster 1, it is most probable to produce a C- or M-flare.

The resulting clusters found with GMM are visualized in Figure 7 on the right. Cluster 3 contains 52% of the non-flaring active regions and 14 to 18% of the flaring active regions. Meanwhile, Cluster 2 contains 34% of the non-flaring active regions and 8 to 18% of the flaring active regions. Active regions that are classified into Cluster 2 and Cluster 3 have thus a relatively large probability to be non-flaring. This statement can be made stronger when the probabilities to belong to multiple clusters are analysed. If an active region has a high probability to belong to both Cluster 2 and Cluster 3, it is highly probable to be non-flaring. Cluster 1 contains only 14% of the non-flaring active regions and 68 to 78% of each of the flaring active regions. This cluster is thus a good one to identify flaring active regions.

In each of the clusters found with GMM, the percentage of each of the different types of flaring active regions is very similar. Therefore, in contrast to K-means, the clustering with GMM is not able to distinguish the strength of the flares.

To get a more quantitative analysis, Figure 8 is a useful addition to 7. They show the same data, but in Figure 8 the normalization is performed per cluster. Therefore, this visualisation can be used to determine the probability that an active regions is of a certain flare type if it belongs to a certain cluster. We clarify this by giving a few examples. When an active regions is assigned to Cluster 3 by the K-means algorithm, it is with 66% probability non-flaring, with 17% a C-flare and with 12% probability an M-flare. An active regions that is assigned to Cluster 2 by K-means will with 94% probability (31% + 33% + 30%) be flaring, with approximately equal probability to be a C-flare, M-flare or X-flare. If an active region belongs to Cluster 1, found with GMM, there is only a 6% chance that it is not flaring. However, when the active region is assigned

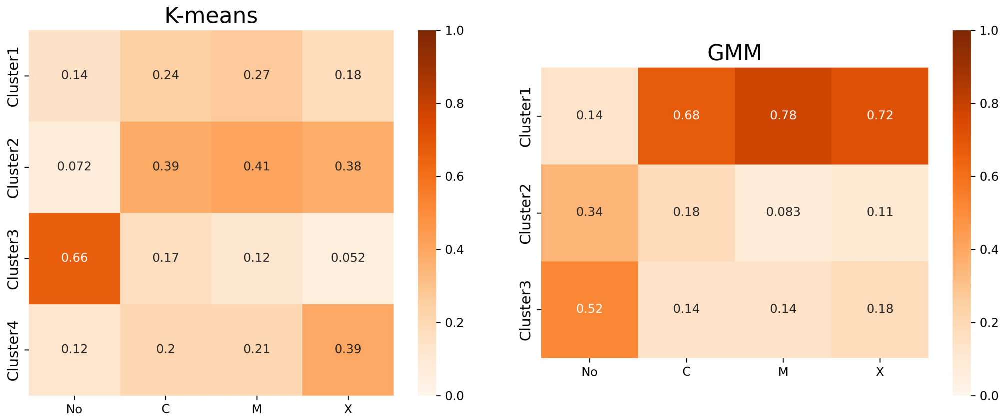

Figure 7: Clustering results of K-means (left) and GMM (right) on the sampled data set. The percentage of each flare included in each of the clusters is shown, where normalization is performed per flare type.

to Cluster 2 or 3 by GMM, there is respectively a chance of 48% and 53% that there are no flares coming out of this active region.

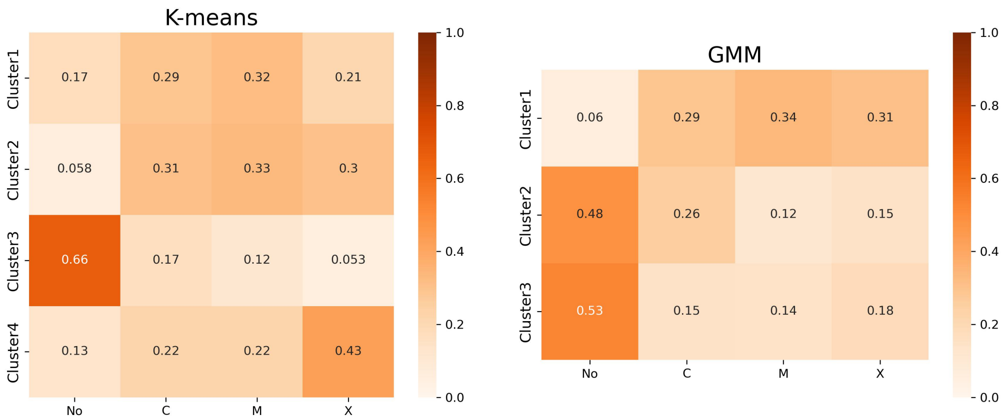

Figure 8: Clustering results of K-means (left) and GMM (right) on the sampled data set. The percentage of each flare included in each of the clusters is shown, where normalization is performed per cluster.

8 Discussion

8.1 Data Processing

In section 4.3.1, we found with Common Factor Analysis that almost all of the information included in the 24 magnetic field parameters could be reduced to only five factors. This is because a lot of the initial parameters were strongly correlated, and do not add any additional information. It is possible then to construct a smaller data set, with only the most useful parameters, containing different distributions for different flare types. This redundancy due to intrinsic correlations between the parameters was also mentioned previously in Bobra and Couvidat (2015) and Barnes et al. (2016).

8.2 Active Region Classification

The supervised clustering method (KNN) has good performance for the M- and X-flares, as well as for the non-flaring active regions. The performance on the C-flares is less accurate, since they are often confused with M-flares and non-flaring active regions. This is probably because their magnetic field parameters are similar to the ones of both the non-flaring data and the M-flares, and their distributions tend to overlap.

With unsupervised clustering (K-means and GMM), non-flaring active regions can be distinguished from flaring active regions. To distinguish between the different flaring active regions is a lot harder. The resulting clusters from K-means show that it is possible to make a distinction between an active region producing strong flares from active regions producing weak flares, but there is still a lot of uncertainty in the distinction among the different flaring energy levels.

The difficulty of differentiating between the flare types is inherent to the data itself, as predicted by analysis of Figure 4. The parameters are very similar for all flaring active regions. Therefore, there is not enough information in the data set for the technique to identify clear differences between C-flares, M-flares and X-flares. Integrating more information into the analysis could provide a clearer distinction. The vector magnetic field data alone is not fully representative of the activity in the whole active region. For example, the maximal difference in magnitude of the magnetic field over the active region could provide valuable information. In future research, the magnetic field parameters should be combined with other features, created through good feature engineering from the original images, for example through edge detection or with variational autoencoders. More data can be included by taking into account EUV observations, at multiple wavelenghts, of the same region.

An extension to the use of the magnetic field parameters is to study their evolution, through time series. The variation of the magnetic field in anticipation of the release of a flare will provide valuable information, being probably more significant for strong flares than for weak flares. The use of time series can also help to distinguish the natural variability of the solar magnetic field from a sudden change in the magnetic field due to flare formation.

The difficulty of differentiating C-, M- and X-flares is also caused by the arbitrary boundaries of the classes, determined by their peak X-ray flux. A C9-flare is very similar to an M1-flare, but they were for this work considered as strictly different classes of flares. The difference between background radiation (non-flaring active regions) and weak C-flares can be very small as well. The strength of flares is a continuous parameter, but was here treated as strictly discrete.

Rather than trying to cluster C-, M- and X-flares separately, trying to distinguish flaring from non-flaring, or weakly flaring from strongly flaring active regions might yield more accurate results. But still the problem remains that an artificial boundary needs to be set in the continuous domain.

Strongly flaring active regions could also be identified as regions with parameter values significantly larger than the mean or median value. Both Sun et al. (2022) and Bobra and Couvidat (2015) tried to identify flaring active regions based on a training set containing only active regions that were either non-flaring or strongly flaring. All active regions that produced C-flares were eliminated. This makes it easier to distinguish flaring from non-flaring active regions. However, for flare prediction, in real-time data the C-flares can not be eliminated and need to be classified correctly as well.

In future research, it could be useful to only consider flaring data. When both non-flaring and flaring data is taken into account, regions with complex and intense magnetic fields are compared against completely quiet regions. This might give the impression that all flaring active regions have similar properties. It is possible that they do appear more distinct when only compared against each other.

9 Conclusion

Throughout this work detailed data cleaning and parameter transformation was conducted to enhance the quality of the Angryk data set and improve the classification results. Supervised clustering, with KNN, is able to distinguish the M- and X-flares, with respectively 99% and 87% correctly identified. However, only half of the C-flares are accurately classified. Unsupervised clustering, with K-means and GMM, identifies clusters with mainly non-flaring active regions and clusters with mainly flaring active regions. However, the clusters contain a mixture of weakly-flaring and strongly-flaring active regions. There is no clear hyperplane in the SHARP parameter space that can separate active regions with different flaring activity. For future projects, additional information should be included, like time series, different parameters - indicating e.g. the topology of active regions - or images of the active regions.

Open Research

This research uses the open source data set SWAN-SF of Angryk et al. (2020b). For more information we would like to refer the reader to the respective paper (Angryk et al., 2020a). The data is available for download through: `https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/EBCFKM`.

The code used to perform all data transformations and generate the clustering results is completely written in Python 3.10, and is accessible on Gitlab: `https://gitlab.com/hanneb/clustering_ar_sf_hbaeke.git` (Baeke, 2022).

Acknowledgments

This work was supported by the Research Foundation – Flanders (FWO), Frank De Winne PhD-aspirant grant (1SF8522N). This work has also received funding from the KULeuven Bijzonder Onderzoeksfonds (BOF) under the C1 project KULeuven Bijzonder Onderzoeksfonds (BOF), from the European Union's Horizon 2020 research and innovation program under grant agreement No. 955606 (DEEP-SEA). The resources and services used in this work were provided by the VSC (Flemish Supercomputer Center), funded by the Research Foundation – Flanders (FWO) and the Flemish Government. The data used in this project was constructed by Angryk et al. (2020b), which originates from the SHARP data series (Bobra et al., 2011), covering the period from 2010-05-01 until 2018-08-31, and the flare reports from GOES containing SSW and XRT flares. We would like to thank the assessors, Jasmina Magdalenic and Andrew Tkachenko, of the master thesis report for their feedback and suggestions. Their feedback led to an improved discussion section in this paper. Finally, the authors declare that there are no conflicts of interest.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., ... Zheng, X. (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems.* Retrieved from `https://www.tensorflow.org/` (Software available from tensorflow.org)

Abduallah, Y., Wang, J. T. L., Nie, Y., Liu, C., & Wang, H. (2020). DeepSun: Machine-Learning-as-a-Service for Solar Flare Prediction. *arXiv e-prints*, arXiv:2009.04238.

Angryk, R., Martens, P., Aydin, B., Kempton, D., Mahajan, S., Basodi, S., ... Georgoulis, M. (2020a). Multivariate Time Series Dataset for Space Weather Data Analytics. *Scientific Data*, *7*, 227. doi: 10.1038/s41597-020-0548-x

Angryk, R., Martens, P., Aydin, B., Kempton, D., Mahajan, S., Basodi, S., ... Georgoulis, M. (2020b). *SWAN-SF [Dataset].* Retrieved from `https://doi.org/10.7910/DVN/EBCFKM` (Harvard Dataverse) doi: 10.7910/DVN/EBCFKM

Baeke, H. (2022). *Identifying Solar Flares by Clustering Active Regions [Code].* Retrieved from `https://gitlab.com/hanneb/clustering_ar_sf_hbaeke.git` (GitLab)

Barnes, G., Leka, K. D., Schrijver, C. J., Colak, T., Qahwaji, R., Ashamari, O. W., ... Wagner, E. L. (2016). A Comparison of Flare Forecasting Methods. I. Results from the All-Clear Workshop. *The Astrophysical Journal*, *829*(2), 89. Retrieved from `https://dx.doi.org/10.3847/0004-637X/829/2/89` doi: 10.3847/0004-637X/829/2/89

Biggs, J. (2019). *FactorAnalyzer release 0.3.2.* Retrieved from `https://github.com/EducationalTestingService/factor_analyzer`

Bobra, M., Hoeksema, J. T., Sun, X., & Turmon, M. (2011). SHARPs - A New Space Weather Data Product from SDO/HMI. In *Agu fall meeting abstracts* (Vol. 2011, p. SH51B-2006).

Bobra, M. G., & Couvidat, S. (2015). Solar Flare Prediction Using SDO/HMI Vector Magnetic Field Data with a Machine Learning Algorithm. *The Astrophysical Journal*, *798*(2), 135. Retrieved from `https://doi.org/10.1088/0004-637x/798/2/135` doi: 10.1088/0004-637x/798/2/135

Bobra, M. G., Wright, P. J., Sun, X., & Turmon, M. J. (2021). SMARPs and SHARPs: Two Solar Cycles of Active Region Data. *The Astrophysical Journal Supplement Series*, *256*(2), 26. Retrieved from `https://doi.org/10.3847%2F1538-4365%2Fac1f1d` doi: 10.3847/1538-4365/ac1f1d

Brownlee, J. (2020). *Tour of Evaluation Metrics for Imbalanced Classification.* Retrieved from `https://machinelearningmastery.com/tour-of-evaluation-metrics-for-imbalanced-classification/` (machinelearningmastery.com)

Caliński, T., & Harabasz, J. (1974). A Dendrite Method for Cluster Analysis. *Communications in Statistics*, *3*(1), 1-27. Retrieved from `https://www.tandfonline.com/doi/abs/10.1080/03610927408827101` doi: 10.1080/03610927408827101

Campello, R. J. G. B., Moulavi, D., & Sander, J. (2013). Density-Based Clustering Based on Hierarchical Density Estimates. In *Advances in knowledge discovery and data mining* (pp. 160–172). Berlin, Heidelberg: Springer Berlin Heidelberg.

Cattell, R. B. (1966). The Scree Test For The Number Of Factors. *Multivariate Be-*

havioral Research, *1*(2), 245-276. Retrieved from `https://doi.org/10.1207/s15327906mbr0102_10` (PMID: 26828106) doi: 10.1207/s15327906mbr0102_10

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, *16*, 321–357. Retrieved from `http://dx.doi.org/10.1613/jair.953` doi: 10.1613/jair.953

Chen, Y., Manchester, W. B., Hero, A. O., Toth, G., DuFumier, B., Zhou, T., ... Gombosi, T. I. (2019). Identifying Solar Flare Precursors Using Time Series of SDO/HMI Images and SHARP Parameters. *Space Weather*, *17*(10), 1404–1426. Retrieved from `http://dx.doi.org/10.1029/2019SW002214` doi: 10.1029/2019sw002214

Chollet, F., et al. (2015). *Keras.* `https://keras.io`.

Cinto, T., Gradvohl, A. L. S., Coelho, G. P., & da Silva, A. E. A. (2020). A Framework for Designing and Evaluating Solar Flare Forecasting Systems. *Monthly Notices of the Royal Astronomical Society*, *495*(3), 3332–3349. Retrieved from `http://dx.doi.org/10.1093/mnras/staa1257` doi: 10.1093/mnras/staa1257

Colak, T., & Qahwaji, R. (2008). Automated McIntosh-Based Classification of Sunspot Groups Using MDI Images. *Solar Physics*, *248*(2), 277-296. doi: 10.1007/s11207-007-9094-3

Cunningham, P., & Delany, S. (2007). k-Nearest Neighbour Classifiers. *Technical Report UCD-CSI-2007-4*.

Dang, T., Li, X., Luo, B., Li, R., Zhang, B., Pham, K., ... Wang, Y. (2022). Unveiling the Space Weather During the Starlink Satellites Destruction Event on 4 February 2022. *Space Weather*, *20*(8), e2022SW003152. doi: 10.1029/2022SW003152

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, *39*(1), 1–22.

Fawcett, T. (2006). Introduction to ROC analysis. *Pattern Recognition Letters*, *27*, 861-874. doi: 10.1016/j.patrec.2005.10.010

Hale, G. E., Ellerman, F., Nicholson, S. B., & Joy, A. H. (1919). The Magnetic Polarity of Sun-Spots. *The Astrophysical Journal*, *49*, 153. doi: 10.1086/142452

Hanssen, A., & Kuipers, W. (1965). *On the Relationship Between the Frequency of Rain and Various Meteorological Parameters: (with Reference to the Problem Ob Objective Forecasting)*. Staatsdrukkerij- en Uitgeverijbedrijf. Retrieved from https://books.google.be/books?id=nTZ8OgAACAAJ

Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the Dimensionality of Data with Neural Networks. *Science*, *313*(5786), 504–507. Retrieved from https://science.sciencemag.org/content/313/5786/504 doi: 10.1126/science.1127647

Horn, J. (1965). A rationale and Test for the Number of Factors in Factor Analysis. *Psychometrika*, *30*, 179-185.

Housseal, S. N., Berger, T. E., & Deshmukh, V. (2019). Using Unsupervised Machine Learning to Explore New Classification of Sunspot Active Regions. In *Agu fall meeting abstracts.*

Ilonidis, S., Bobra, M. G., & Couvidat, S. (2015). Solar Flare Forecasting Using Time Series of SDO/HMI Vector Magnetic Field Data and Machine Learning Methods. In *AAS/AGU triennial earth-sun summit* (Vol. 1, p. 311.03).

Jiang, N., Rong, W., Peng, B., Nie, Y., & Xiong, Z. (2015). An Empirical Analysis of Different Sparse Penalties for Autoencoder in Unsupervised Feature Learning. In *Ijcnn* (p. 1-8). IEEE. Retrieved from http://dblp.uni-trier.de/db/conf/ijcnn/ijcnn2015.html#JiangRPNX15

Jiao, Z., Sun, H., Wang, X., IV, W., Gombosi, T., Hero, A., & Chen, Y. (2020). Solar Flare Intensity Prediction With Machine Learning Models. *Space Weather*, *18*. doi: 10.1029/2020SW002440

Jonas, E., Bobra, M., Shankar, V., Todd Hoeksema, J., & Recht, B. (2018). Flare Prediction Using Photospheric and Coronal Image Data. *Solar Physics*, *293*(3), 48. doi: 10.1007/s11207-018-1258-9

Kawabata, Y., Iida, Y., Doi, T., Akiyama, S., Yashiro, S., & Shimizu, T. (2018, dec). Statistical Relation between Solar Flares and Coronal Mass Ejections with Respect to Sigmoidal Structures in Active Regions. *The Astrophysical Journal*, *869*(2), 99. Retrieved from https://dx.doi.org/10.3847/1538-4357/aaebfc doi: 10.3847/1538-4357/aaebfc

Kingma, D. P., & Ba, J. (2015). Adam: A Method for Stochastic Optimization. *CoRR*, *abs/1412.6980*.

Klambauer, G., Unterthiner, T., Mayr, A., & Hochreiter, S. (2017). Self-Normalizing Neural Networks. In *Proceedings of the 31st international conference on neural information processing systems* (p. 972–981). Red Hook, NY, USA: Curran Associates Inc.

Knipp, D., Ramsay, A., Beard, E., Boright, A., Cade, T., Hewins, I., … Smart, D. (2016). The May 1967 Great Storm and Radio Disruption Event: Extreme Space Weather and Extraordinary Responses: May 1967 Solar and Geomagnetic Storm. *Space Weather*, *14*. doi: 10.1002/2016SW001423

Lavorini, V. (2018). *Gaussian Mixture Model clustering: how to select the number of components.* Retrieved from `https://towardsdatascience.com/gaussian-mixture-model-clusterization-how-to-select-the-number-of-components-clusters-553bef45f6e4` (Towards Data Science)

Lemaître, G., Nogueira, F., & Aridas, C. K. (2017). Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning. *Journal of Machine Learning Research*, *18*(17), 1-5. Retrieved from `http://jmlr.org/papers/v18/16-365.html`

Liu, C., Deng, N., Wang, J., & Wang, H. (2017). Predicting Solar Flares Using SDO/HMI Vector Magnetic Data Products and the Random Forest Algorithm. *The Astrophysical Journal*, *843*, 104. doi: 10.3847/1538-4357/aa789b

Liu, J., Wang, W., Qian, L., Lotko, W., Burns, A. G., Pham, K., ... Wilder, F. (2021). Solar Flare Effects in the Earth's Magnetosphere. *Nature Physics*, *17*(7), 807-812. doi: 10.1038/s41567-021-01203-5

Lloyd, S. (1982). Least Squares Quantization in PCM. *IEEE Transactions on Information Theory*, *28*(2), 129-137. doi: 10.1109/TIT.1982.1056489

MacQueen, J. B. (1967). Some Methods for Classification and Analysis of Multivariate Observations. In *Berkeley symposium on mathematical statistics and probability* (p. 281-297).

Makhzani, A., & Frey, B. J. (2014). k-Sparse Autoencoders. *CoRR*, *abs/1312.5663*.

Maloney, S. A., & Gallagher, P. T. (2018). Sunspot Group Classification using Neural Networks. In *Catalyzing solar connections* (p. 92).

McIntosh, P. S. (1990). The Classification of Sunspot Groups. *Solar Physics*, *125*(2), 251-267. doi: 10.1007/BF00158405

Mikaelian, T. (2009). *Spacecraft Charging and Hazards to Electronics in Space.* arXiv. Retrieved from `https://arxiv.org/abs/0906.3884` doi: 10.48550/ARXIV.0906.3884

Narkhede, S. (2018). *Understanding AUC-ROC Curve.* Retrieved from `https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5` (TowardsDataScience)

Ng, A., et al. (2011). Sparse Autoencoder. *CS294A Lecture notes*, *72*(2011), 1–19.

Nguyen, T. T., Willis, C. P., Paddon, D. J., Nguyen, S. H., & Nguyen, H. S. (2006). Learning Sunspot Classification. *Fundamenta Informaticae*, *72*(1–3), 295–309.

Priest, E. R., & Forbes, T. G. (2002). The Magnetic Nature of Solar Flares. *Astronomy and Astrophysics Reviews*, *10*(4), 313-377. doi: 10.1007/s001590100013

Pulkkinen, A., Lindahl, S., Viljanen, A., & Pirjola, R. (2005). Geomagnetic Storm of 29-31 October 2003: Geomagnetically Induced Currents and their Relation to Problems in the Swedish High-Voltage Power Transmission System. *Space Weather*, *3*. doi: 10.1029/2004SW000123

Ran, H., Liu, Y. D., Guo, Y., & Wang, R. (2022). *Relationship between Successive Flares in the Same Active Region and Space-Weather HMI Active Region Patch (SHARP) Parameters.* arXiv. Retrieved from `https://arxiv.org/abs/2207.07254` doi: 10.48550/ARXIV.2207.07254

Redmon, R., Seaton, D., Steenburgh, R., He, J., & Rodriguez, J. (2018). September 2017's Geoeffective Space Weather and Impacts to Caribbean Radio Communications During Hurricane Response. *Space Weather*, *16*. doi: 10.1029/2018SW001897

Rousseeuw, P. J. (1987). Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis. *Journal of Computational and Applied Mathematics*, *20*, 53-65. Retrieved from `https://www.sciencedirect.com/science/article/pii/0377042787901257` doi: https://doi.org/10.1016/0377-0427(87)90125-7

Satopaa, V., Albrecht, J., Irwin, D., & Raghavan, B. (2011). Finding a "Kneedle" in a Haystack: Detecting Knee Points in System Behavior. In *2011 31st international conference on distributed computing systems workshops* (p. 166-171). doi: 10.1109/ICDCSW.2011.20

Schwarz, G. (1978). Estimating the Dimension of a Model. *The Annals of Statistics*, *6*(2), 461 – 464. Retrieved from `https://doi.org/10.1214/aos/1176344136` doi: 10.1214/aos/1176344136

Sheeley, N.R. (2020). Solar Magnetic Field. *AccessScience*.

Sinha, S., Gupta, O., Singh, V., Lekshmi, B., Nandy, D., Mitra, D., ... Pal, S. (2022). A Comparative Analysis of Machine-learning Models for Solar Flare Forecasting: Identifying High-performing Active Region Flare Indicators. *The Astrophysical Journal*, *935*(1), 45. Retrieved from `https://dx.doi.org/10.3847/1538-4357/ac7955` doi: 10.3847/1538-4357/ac7955

Smith, L. N. (2017). *Cyclical Learning Rates for Training Neural Networks.* arXiv. Retrieved from `https://arxiv.org/abs/1506.01186` doi: 10.48550/ARXIV.1506.01186

Smith, M. C., Jones, A. R., & Sandoval, L. (2018). Automating the McIntosh Classification System using Machine Learning. In *Agu fall meeting abstracts* (Vol. 2018, p. SM31D-3526).

Sorkin, A. (1982). Economic Aspects of Natural Hazards. In *Economic aspects of natural hazards.* Lexington Books; distributed Gower.

Spearman, C. (1904). General Intelligence, Objectively Determined and Measured. *The American Journal of Psychology*, *15*(2), 201–292. Retrieved from http://www.jstor.org/stable/1412107

Sun, Z., Bobra, M. G., Wang, X., Wang, Y., Sun, H., Gombosi, T., … Hero, A. (2022). Predicting Solar Flares Using CNN and LSTM on Two Solar Cycles of Active Region Data. *The Astrophysical Journal*, *931*(2), 163. Retrieved from https://doi.org/10.3847\%2F1538-4357\%2Fac64a6 doi: 10.3847/1538-4357/ac64a6

Wang, X., Chen, Y., Toth, G., IV, W., Gombosi, T., Hero, A., … Liu, Y. (2020). Predicting Solar Flares with Machine Learning: Investigating Solar Cycle Dependence. *The Astrophysical Journal*, *895*, 3. doi: 10.3847/1538-4357/ab89ac

Yu, L., & Liu, H. (2004). Efficient Feature Selection via Analysis of Relevance and Redundancy. *Journal of Machine Learning Research*, *5*, 1205–1224.

Zhang, D., & Tsai, J. J. P. (2005). *Machine Learning Applications in Software Engineering*. World Scientific. Retrieved from https://www.worldscientific.com/doi/abs/10.1142/5700 doi: 10.1142/5700

Zhang, H., Li, Q., Yang, Y., Jing, J., Wang, J. T. L., Wang, H., & Shang, Z. (2022). Solar Flare Index Prediction Using SDO/HMI Vector Magnetic Data Products with Statistical and Machine-learning Methods. *The Astrophysical Journal Supplement Series*, *263*(2), 28. Retrieved from https://doi.org/10.3847%2F1538-4365%2Fac9b17 doi: 10.3847/1538-4365/ac9b17