

The Impacts of Magnetogram Projection Effects on Solar Flare Forecasting

GRIFFIN T. GOODWIN ¹, VIACHESLAV M. SADYKOV ¹ AND PETRUS C. MARTENS ¹

¹*Physics & Astronomy Department, Georgia State University, Atlanta, GA 30303, USA; ggoodwin5@gsu.edu*

ABSTRACT

This work explores the impacts of magnetogram projection effects on machine learning-based solar flare forecasting models. Utilizing a methodology proposed by Falconer et al. (2016), we correct for projection effects present in Georgia State University’s Space Weather Analytics for Solar Flares (SWAN-SF) benchmark data set. We then train and run a support vector machine classifier on the corrected and uncorrected data, comparing differences in performance. Additionally, we provide insight into several other methodologies that mitigate projection effects, such as stacking ensemble classifiers and active region location-informed models. Our analysis shows that data corrections slightly increase both the true positive (correctly predicted flaring samples) and false positive (non-flaring samples predicted as flaring) prediction rates, averaging a few percent. Similarly, changes in performance metrics are minimal for the stacking ensemble and location-based model. This suggests that a more complicated correction methodology may be needed to see improvements. It may also indicate inherent limitations when using magnetogram data for flare forecasting.

Keywords: Space weather (2037), Solar flares (1496), Support vector machine (1936)

1. INTRODUCTION

Active region (AR) vector magnetograms are frequently used to forecast solar flares as they provide insight into one of the only reservoirs capable of driving the flaring process: free magnetic energy. By deriving a set of physics-based parameters from the data, we can produce a comprehensive set of preflare features that can be easily integrated into machine learning (ML) models. However, as ARs traverse farther from the solar disk center, the quality of magnetic field data degrades substantially. Near the solar limb, AR vector magnetograms and their properties (magnetic flux, current helicity, etc) suffer from systematic observational trends unrelated to the evolution of the AR itself, known as projection effects. These trends appear due to line height formation effects, foreshortening of magnetogram pixels, and increased noise which may make an AR appear stronger or weaker than it truly is (Kitiashvili et al. 2015; Falconer et al. 2016). This complicates forecasting, especially for events near the classification border, potentially introducing unnecessary false positive and negative predictions. Correcting for these effects appears to be imperative in ensuring a well-performing model, yet, as far as we know, no studies have explored multi-parameter corrections thoroughly. In this work, our goal is to employ a methodology similar to that of Falconer et al. (2016), which was originally used for unsigned magnetic flux corrections, to remove the projection effects for the AR features present in Georgia State University’s Space Weather Analytics for Solar Flares benchmark

data set (Angryk et al. 2020a,b), and examine the impacts these corrections may have on forecasting performance. Additionally, we make use of two other correction methodologies, a stacking ensemble classifier, and an active region location-informed model, to reduce the influence of projection effects. Interestingly, as far as we are aware, these techniques have not been explored much in the literature. We would like to note that ideally projection effect corrections should be applied directly to the data source itself (the magnetograms), however, due to the significant amount of effort required for computing the derived magnetogram features, this methodology is beneficial to consider.

The remaining sections of this work are organized as follows: Section 2 highlights the data used in our predictions. Section 3 describes the methodology used to correct the projection effects. Finally, Sections 4 and 4.3 detail our results and conclusions from this work.

2. DATA

Georgia State’s Space Weather Analytics for Solar Flares (SWAN-SF) benchmark data set (Angryk et al. 2020a,b) is an ML-ready compilation of Solar Dynamics Observatory Helioseismic and Magnetic Imager (SDO HMI, Scherrer et al. 2012) AR vector magnetogram parameter time series spanning part of Solar Cycle 24 (May 2010 - August 2018). Each file contains 12 hours’ worth of data at a 12-minute cadence, with twenty-four physics-based parameters calculated at each timestamp (see Table 1). Every file is labeled based on the strongest flaring instance to occur in the following 24 hours, with M- and X-class flares deemed as positive (or flaring) events, while weaker B-, C-class, and completely flare quiet instances are considered negative (or non-flaring) events. SWAN-SF is divided into five temporally segmented partitions, each containing approximately the same number of M- and X-class flares (Ahmadzadeh et al. 2021). This allows for quick and efficient training/testing of ML classifiers, which we take advantage of in this work. Figure 1 depicts the distribution of flaring/non-flaring classes within each partition. If the reader would like to learn more about how SWAN-SF was constructed, please see Angryk et al. (2020b).

We want to make clear that our primary focus in this work is to study the impacts of projection effects on flare forecasting, rather than produce the best-performing classifier. Thus, to simplify and accelerate our training/testing process, we employ a feature-vector-based (contrary to a time series-based) forecaster, which requires us to reduce each multivariate time series file to a single data point. To achieve this, we calculate the descriptive statistics for each of the twenty-four magnetogram parameters in a given 12 hr file. This includes the mean, median, standard deviation, variance, maximum, minimum, skewness, and kurtosis. Then, to capture aspects pertinent to the evolution of the time series, we record the last value in the file, the mean of the gradient derivative, and the standard deviation of the gradient derivative. For files with missing data, we linearly interpolated beforehand. This method saves us significant time, reducing each file from 60 timestamps by 24 parameters (or 1440 dimensions) to 1 timestamp by 264 parameters, i.e., 11 parameters for each of the 24 magnetogram parameters.

3. METHODOLOGY

In this work, we follow a similar methodology introduced by Falconer et al. (2016) to correct for projection effects. By normalizing the magnetic field data for a given AR to the value near the central meridian (where projection effects are minimal), combining the results across many different ARs, and employing a polynomial fit, we create a curve that describes the severity of the projection effects as a function of heliocentric angle (θ , the angle between an AR’s line-of-sight and surface normal). For

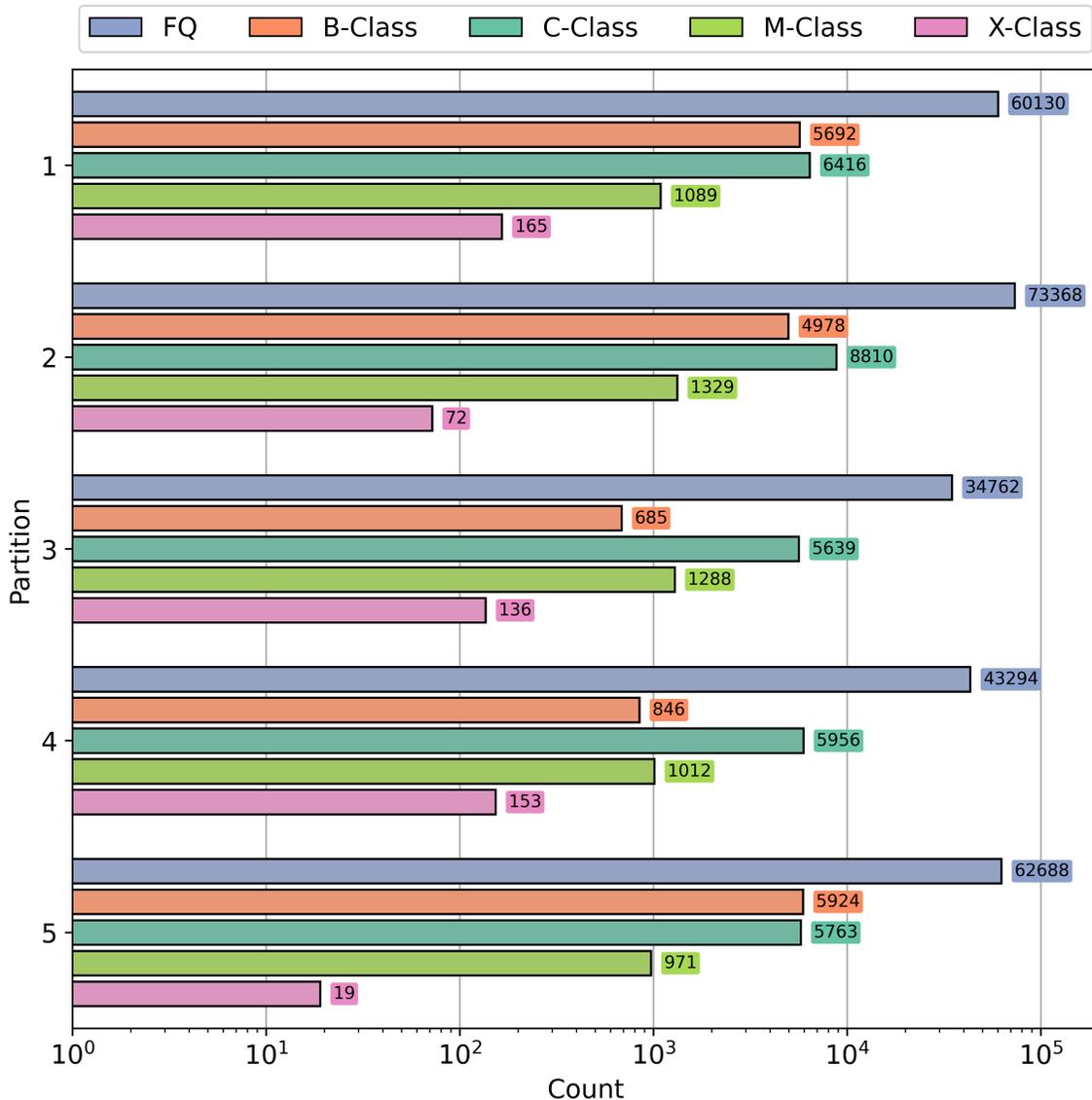


Figure 1. The flare class distribution for each partition within the Space Weather Analytics for Solar Flares benchmark data set. Bars are color-coded by the flaring class: Flare quiet (blue), B- (orange), C- (teal), M- (green), and X-class (pink).

any new AR, we can then remove the projection effects by dividing the data by the polynomial value at the AR’s current heliocentric angle, rescaling the data back to unity (or the estimated unprojected value). In the following sections, we will provide details on how we selected our “benchmark” ARs for modeling the polynomials (see Section 3.1), how we generated the polynomial fits for each correction (see Section 3.2), how we trained our ML models (see Section 3.3), and the performance metrics we used to analyze our results (see Section 3.4).

3.1. Active Region Selection

The first step in the correction process is to identify the “benchmark” ARs that will be used to model the average projection effects across the solar disk. Ideally, these ARs should demonstrate

Parameter	Description	Formula
ABSNJZH	Absolute value of the net current helicity in G^2/m	$H_{c_{abs}} \propto \sum B_z \cdot J_z $
EPSX	Sum of X-component of normalized Lorentz force	$\delta F_x \propto \frac{\sum B_x B_z}{\sum B^2}$
EPSY	Sum of Y-component of normalized Lorentz force	$\delta F_y \propto \frac{-\sum B_y B_z}{\sum B^2}$
EPSZ	Sum of Z-component of normalized Lorentz force	$\delta F_z \propto \frac{\sum (B_x^2 + B_y^2 - B_z^2)}{\sum B^2}$
MEANALP	Mean twist parameter	$\alpha_{total} \propto \frac{\sum J_z \cdot B_z}{\sum B_z^2}$
MEANGAM	Mean inclination angle	$\bar{\gamma} = \frac{1}{N} \sum \arctan\left(\frac{B_h}{B_z}\right)$
MEANGBH	Mean value of the 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	Mean value of the 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	Mean value of the 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	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	Mean current helicity	$\overline{H_c} \propto \frac{1}{N} \sum B_z \cdot J_z$
MEANPOT	Mean photospheric excess magnetic energy density	$\bar{\rho} \propto \frac{1}{N} \sum (B^{Obs} - B^{Pot})^2$
MEANSHR	Mean shear angle	$\bar{\Gamma} = \frac{1}{N} \sum \arccos\left(\frac{B^{Obs} \cdot B^{Pot}}{ B^{Obs} B^{Pot} }\right)$
R.VALUE	Total unsigned flux around high gradient polarity inversion lines using the B_{los} component	$\Phi = \sum B_{los} \cdot dA$ (within R mask)
SAVNCP	Sum of the absolute value of the net current per polarity	$J_{z_{sum}} \propto \sum B_z^+ J_z dA + \sum B_z^- J_z dA $
SHRGT45	Area with shear angle greater than 45 degrees	$\frac{\text{Area with Shear } > 45^\circ}{\text{Total Area}}$
TOTBSQ	Total magnitude of Lorentz force	$F \propto \sum B^2$
TOTFX	Sum of X-component of Lorentz force	$F_x \propto \sum B_x B_z dA$
TOTFY	Sum of Y-component of Lorentz force	$F_y \propto \sum B_y B_z dA$
TOTFZ	Sum of Z-component of Lorentz force	$F_z \propto \sum (B_x^2 + B_y^2 - B_z^2) dA$
TOTPOT	Total photospheric magnetic energy density	$\rho_{tot} \propto \sum (\vec{B}^{Obs} - \vec{B}^{Pot})^2 dA$
TOTUSJH	Total unsigned current helicity	$H_{c_{total}} \propto \sum B_z \cdot J_z$
TOTUSJZ	Total unsigned vertical current	$J_{z_{total}} = \sum J_z dA$
USFLUX	Total unsigned flux in Maxwells	$\Phi = \sum B_z dA$

Table 1. The derived magnetic field parameters in the Space Weather Analytics for Solar Flares benchmark data set. These parameters have been curated by [Bobra & Couvidat \(2015\)](#). This table has been adapted from [Angryk et al. \(2020b\)](#).

minimal variations in their strength over their lifetimes. We can ensure this by analyzing an AR's unsigned magnetic flux profiles, only selecting those that show a highly symmetric evolution across the solar meridian. This guarantees that any observed growths/decays in the data are (mostly) due to the projection effects themselves, rather than any intrinsic evolution of the AR that may occur. We invoke the following methodology to quantitatively analyze the symmetry of the flux profile for a

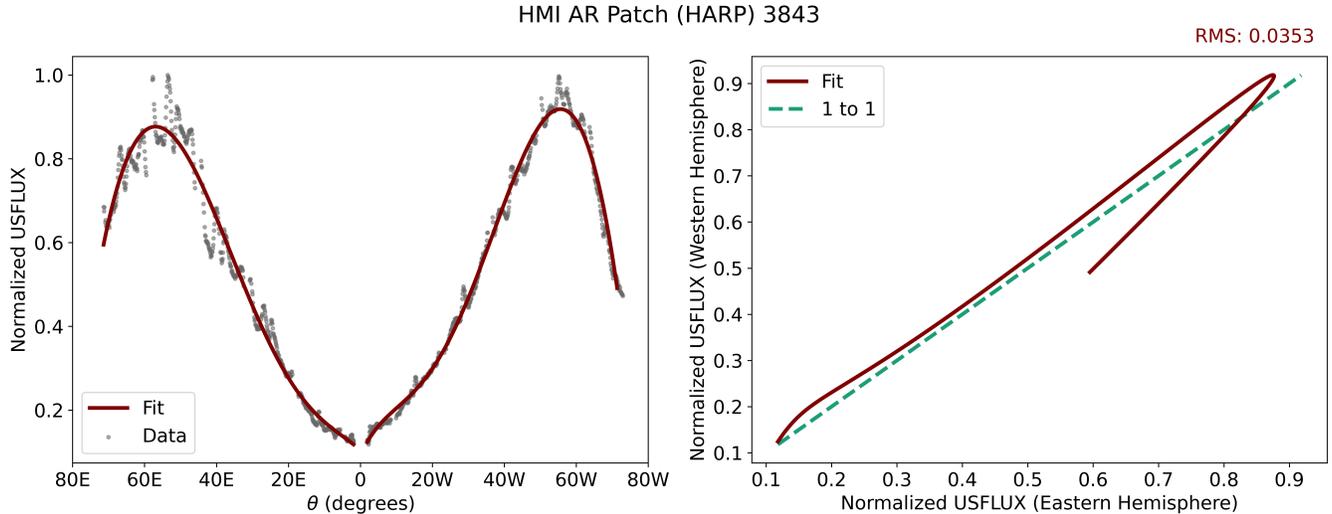


Figure 2. (*Left*) The normalized unsigned magnetic flux for HARP 3843 versus its heliocentric angle (θ) in degrees. The data in grey are normalized to the max unsigned magnetic flux. A spline is fit through the data in the eastern hemisphere (left of 0°) and the western hemisphere (right of 0°) in maroon. Notice the highly symmetric nature of the data across the traversal. This indicates that the flux variations we observe are mostly dominated by projection effects. (*Right*) In maroon, the unsigned magnetic flux spline from the western hemisphere is plotted versus the unsigned magnetic flux spline from the eastern hemisphere (at the same heliocentric angle). A one-to-one line is shown in green. The root mean squared value between the fit data and one-to-one line is shown in the top right.

given AR. First, we down-select ARs from the 4098 present in SWAN-SF based on the criterion that their data spans most of the solar disk. Since we will only be making forecasts for ARs with θ s out to 70° , we select ARs that have data within longitudes and θ s of 73° (a 3° buffer is included to ensure a good final polynomial fit). We also want the ARs to pass relatively close to the disk center, so we include an additional criterion that data must exist within θ s of 30° . From here, we take each of the 398 down-sampled ARs and fit two (one for each hemisphere) 5th-degree splines through the unsigned magnetic flux normalized to the maximum value along the traversal, as a function of the θ . We then take the data from the fit of the western hemisphere and plot it against the data from the eastern hemisphere (at the same θ). We also include a one-to-one line that depicts a perfectly symmetric evolution (see Figure 2). By calculating the root mean squared (RMS) error between the fit data and the one-to-one line, we can quantify how symmetric the flux evolution is across hemispheres. A low RMS should correspond to a good one-to-one fit. Through visual inspection, we settled on an RMS cutoff of <0.3 for our final sample of ARs. This ensures that we have a sufficient number of ARs to work with while removing the egregiously asymmetric ARs. In the end, we were left with 260 ARs.

3.2. Fitting and Applying the Corrective Polynomials

Now that we have selected our “benchmark” ARs, we can begin generating the correction polynomials for each magnetic field parameter in Table 1. Our approach consists of two stages. First, we establish a “prefit” polynomial that will calibrate the final fit. This “prefit” curve is constructed using only ARs that pass within 10° of the solar meridian. For each of these ARs, we normalize the data based on the mean value of the observed magnetic field parameter between the AR’s minimum

θ and the minimum plus 3° . This ensures that near 0° , where we expect minimal projection effects, the correction is set to unity. Then, the rescaled data beyond this θ range models the strength of the projection effects. Combining all of our “prefit” normalized AR data, we can apply a bootstrapping technique to generate a fit with uncertainties. This method involves resampling our “prefit” AR data 500 times, with replacement. More specifically, say we had x number of ARs within our “prefit” data. Within a single bootstrapped sample, x random AR selections are made from our original pool with the possibility of selecting the same region multiple times. We repeat this process 500 times and apply a 7th-degree polynomial fit through the data of each of our bootstrapped samples. This fit is defined as follows:

$$y = 1 + \beta_0\theta + \beta_1\theta^2 + \beta_2\theta^3 + \beta_4\theta^5 + \beta_5\theta^6 + \beta_6\theta^7 \quad (1)$$

where β_n are constants, θ is the observed heliocentric angle, and y is the correction factor. The average bootstrapped fit is then used as the final correction polynomial for the “prefit” data. The reason we create “prefit” polynomials first is that ARs with minimum θ s greater than 10° should not necessarily be normalized to unity. In principle, these ARs will already have some degradation due to projection effects. Thus, we normalize their data to the average “prefit” polynomial value between the minimum θ and the minimum plus 3° . Once this is done, we re-run the bootstrapping method, averaging the curves to achieve our final fit for each magnetic field parameter. Having these polynomials then allows us to generate our corrected SWAN-SF data set. By iterating through each time series file, rescaling the 24 magnetogram parameters by their corrective factors at the appropriate θ , and recalculating the descriptive statistics, we should largely remove any projection effects within the data.

We want to note that because some parameters are signed (TOTFX/Y/Z, ESPX/Y/Z, MEANJZH, MEANJZD, MEANALP), we took the absolute value of these features when generating our polynomial fits. The absolute value was not used during forecasting. Additionally, the R_VALUE feature is dependent on the polarity inversion line (PIL) of the AR. Since not all ARs have a distinct PIL, some time series may contain R_VALUE = 0, which may interfere with our fit. Thus, data points where R_VALUE = 0 were removed during fitting.

Figure 3 highlights the results of the fitting process, while Table 2 provides the constants for each fit. The grey dots show the normalized data for the 260 “benchmark” ARs, the solid maroon line is the average fit of the bootstrapped fits in orange, the dotted maroon line shows the median absolute deviation from the fit, and the black dashed line emphasizes where the correction factor is 1. From the figure, we see that each parameter is influenced differently by projection effects, with correction factors ranging from as low as 0.5 (MEANJZD) up to 25 (TOTFX) at certain θ s.

The shape of the polynomial tends to land within two categories, either a gradual fall-off at larger θ s or a peak occurring between 30° and 70° . Of course, there are a few oddball parameters such as MEANALP, MEANJZH, R_VALUE, and TOTFY that do not fall into either of these categories. Those that are associated with the hump are directly related to the radial field component of the magnetic field as well as the masking area (dA) of the AR (eg. USFLUX), while those with a gradual drop-off tend to be directly related to the horizontal component of the magnetic field (eg. MEANGBH). As ARs move towards the limb, the radial field component (B_z in the local Cartesian coordinate system rotating with the AR) increasingly influences the transversal field measurements (perpendicular to the line of sight), while the horizontal field components (B_x, B_y) begin to dominate the line of sight

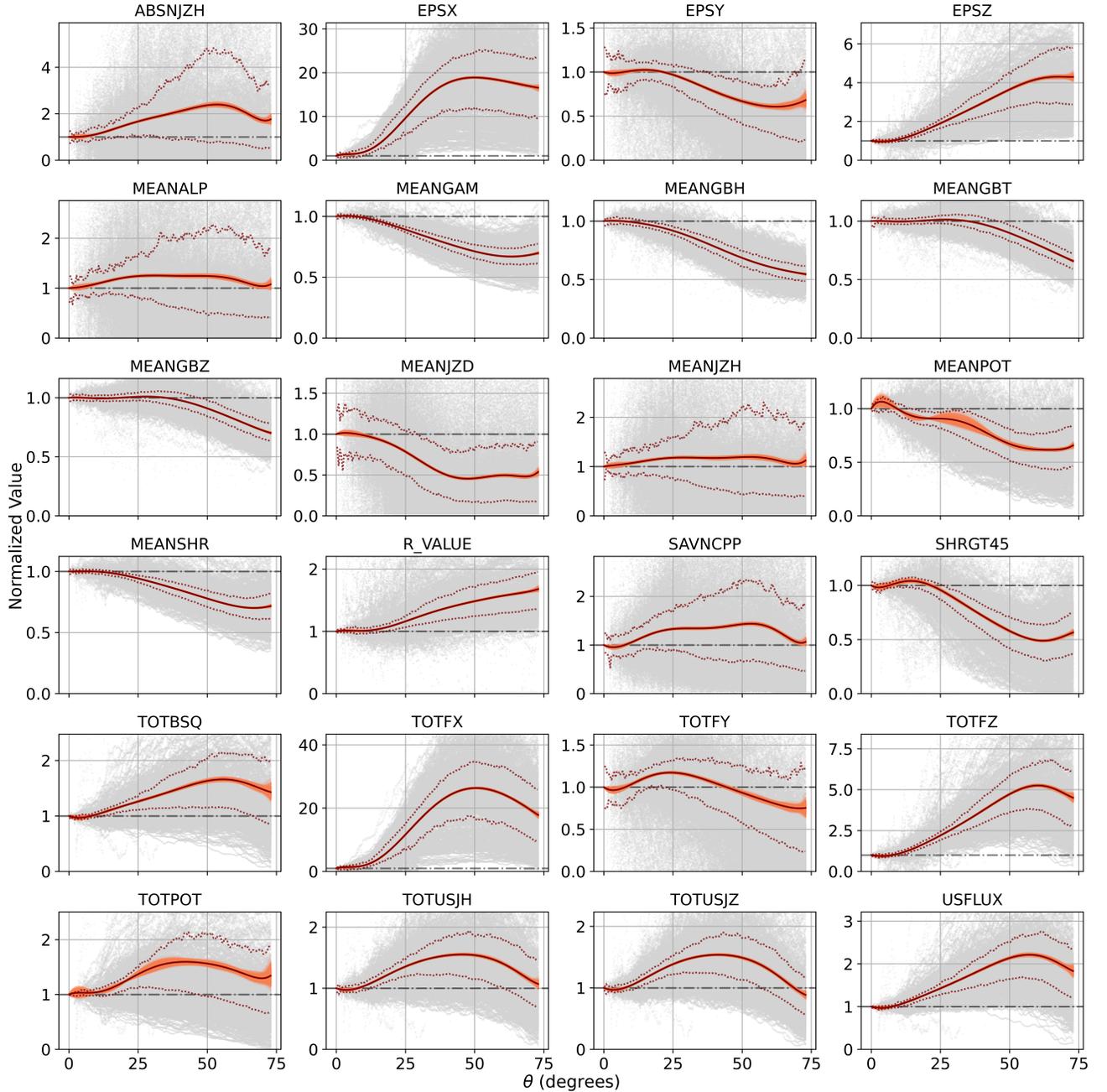


Figure 3. The correction polynomials for each of the 24 derived magnetic field parameters measured in normalized units as a function of heliocentric angle (θ). The solid maroon line represents the average fit from the bootstrapped fits in orange for the data in grey. The dashed maroon lines highlight the median absolute deviation of the data above and below the polynomial fit. The black dashed-dotted line emphasizes the point where the correction factor is 1. Please note that often the bootstrapped fits (orange) are relatively similar to the average fit (maroon), thus the curves can occasionally be hard to identify (e.g. MEANSHR).

measurements. In the HMI magnetograms, the line of sight component of the field has an error of 5-10 G, while the transverse field noise is on the order of 100 G (Liu et al. 2012; Hoeksema et al. 2014;

Falconer et al. 2016). Consequently, this means near the limb, the increased noise and contribution of the transversal component will expand the ARs masking region. This synthetically increases the features correlated to B_z and dA . For the parameters dependent on the horizontal field components, the drop-off likely occurs due to a combination of foreshortening and an increased contribution of the line of sight observation (which has less noise) when making horizontal field measurements. The reduced noise near the limb should decrease the average measurements of the horizontal component, compared to those made near the central meridian, where noise is high.

Finally, when analyzing the goodness of fit, we find that it varies widely depending on the parameter. For example, MEANGAM, MEANBH, MEANGBT, MEANGBZ, and MEANSHR all have relatively clear trends, while parameters such as ABSNJZH, EPSY, MEANALP, MEANJZD, MEANJZH, SAVNCPP, and TOTFY show no obvious fit and are quite scattered.

Parameter	β_0	β_1	β_2	β_3	β_4	β_5	β_6
ABSNJZH	-3.780e-04	-1.992e-04	2.772e-04	-1.732e-05	4.523e-07	-5.451e-09	2.473e-11
EPSX	2.147e-01	-5.051e-02	5.401e-03	-1.911e-04	3.148e-06	-2.519e-08	7.959e-11
EPSY	-9.715e-03	2.231e-03	-1.492e-04	4.212e-06	-6.138e-08	4.579e-10	-1.367e-12
EPSZ	-1.955e-02	2.224e-03	1.086e-04	-8.265e-06	2.109e-07	-2.428e-09	1.045e-11
MEANALP	2.747e-03	5.508e-04	3.417e-05	-3.986e-06	1.217e-07	-1.557e-09	7.261e-12
MEANGAM	2.700e-03	-4.375e-04	4.165e-06	2.204e-07	-7.620e-09	9.435e-11	-4.099e-13
MEANGBH	2.332e-03	-4.772e-04	2.407e-05	-9.089e-07	1.768e-08	-1.620e-10	5.655e-13
MEANGBT	1.657e-03	-5.667e-04	5.370e-05	-2.051e-06	3.661e-08	-3.163e-10	1.076e-12
MEANGBZ	2.270e-03	-7.084e-04	6.245e-05	-2.311e-06	4.112e-08	-3.602e-10	1.260e-12
MEANJZD	1.063e-02	-2.389e-03	1.861e-04	-8.809e-06	2.121e-07	-2.411e-09	1.036e-11
MEANJZH	1.025e-02	-1.240e-03	1.574e-04	-8.281e-06	2.030e-07	-2.343e-09	1.029e-11
MEANPOT	4.287e-02	-8.872e-03	6.295e-04	-2.167e-05	3.868e-07	-3.458e-09	1.229e-11
MEANSHR	-3.996e-04	2.581e-04	-3.342e-05	1.280e-06	-2.422e-08	2.267e-10	-8.214e-13
R_VALUE	9.112e-03	-2.176e-03	1.942e-04	-6.922e-06	1.240e-07	-1.117e-09	4.032e-12
SAVNCPP	-2.819e-02	4.946e-03	-1.587e-04	-9.472e-07	1.276e-07	-2.131e-09	1.108e-11
SHRGT45	-1.564e-02	4.008e-03	-3.019e-04	1.006e-05	-1.762e-07	1.584e-09	-5.730e-12
TOTBSQ	-1.975e-02	3.703e-03	-1.876e-04	4.892e-06	-6.384e-08	3.555e-10	-4.791e-13
TOTFX	2.451e-01	-5.886e-02	6.093e-03	-2.029e-04	3.164e-06	-2.440e-08	7.576e-11
TOTFY	-2.346e-02	4.590e-03	-2.393e-04	5.621e-06	-6.687e-08	3.742e-10	-6.807e-13
TOTFZ	-2.945e-02	3.581e-03	1.099e-04	-1.064e-05	3.034e-07	-3.800e-09	1.740e-11
TOTPOT	2.206e-02	-5.202e-03	5.040e-04	-1.980e-05	3.837e-07	-3.693e-09	1.415e-11
TOTUSJH	-1.6339e-02	2.592e-03	-5.808e-05	-5.554e-07	4.066e-08	-6.017e-10	2.950e-12
TOTUSJZ	-1.901e-02	3.218e-03	-9.366e-05	4.323e-07	2.494e-08	-4.716e-10	2.530e-12
USFLUX	-1.963e-02	3.462e-03	-1.261e-04	2.062e-06	-1.715e-09	-3.156e-10	2.322e-12

Table 2. The constants for each 7th-degree polynomial fit for each magnetic field parameter. The fits are defined mathematically in the following way: $y = 1 + \beta_0\theta + \beta_1\theta^2 + \beta_2\theta^3 + \beta_4\theta^5 + \beta_5\theta^6 + \beta_6\theta^7$, where β_n are constants, θ is the observed heliocentric angle, and y is the correction factor

3.3. Machine Learning Models, Feature Selection, and Hyperparameter Tuning

Machine learning (ML) has emerged as a powerful tool in heliophysics for forecasting solar transient events. In this study, we apply ML techniques to predict the occurrence of M- or X-class flares from an AR within a subsequent 24 hr time window. Known as a binary classification task, we provide positive (flare) and negative (no flare) predictions for individual point-in-time data. In ML, there are three components to every forecasting model: a training (plus validation) data set, a testing data set, and the algorithm itself, which generally includes hyperparameters, and occasionally weights or biases (e.g., see [Badillo et al. 2020](#) for more details). In this work, we use every possible combination of the temporally segmented partitions of SWAN-SF to train/test our algorithm (1/2, 1/3, 1/4, 1/5, 2/1, etc.). This was done for both corrected and uncorrected data. To address the problem of differing scales within the derived magnetogram parameters, we renormalize each training dataset feature distribution to a mean of 0 and a variance of 1. This transformation is calculated by using the following formula: $z = \frac{x-u}{s}$, where z is the transformed value, x is the input value, u is the mean of the training samples, and s is the standard deviation of the training samples. We then apply this exact transformation to the testing dataset, with u and s being the (same) values obtained during the scaling of the training set. By implementing this training/testing approach, we obtain a wealth of results critical to judging the effectiveness of corrections.

We selected the support vector machine (SVM) algorithm for forecasting purposes. SVM is a robust, efficient, and widely used classifier, making it an excellent choice for flare prediction. It has also been shown to perform comparably to decision tree and multilayer perceptron forecasters when utilizing point-in-time magnetogram data, providing superior performance, while not compromising too much in terms of complexity and training time ([Goodwin et al. 2024](#)). This algorithm aims to develop a multidimensional hyperplane that maximizes the separation between labels within a feature space. In the context of this paper, our labels are the flaring/non-flaring events, and our feature space consists of some subset of the 264 descriptive magnetogram parameters. More often than not, labels are not linearly separable, requiring a more sophisticated hyperplane to be developed. As a remedy, “kernel tricks” are frequently applied to remap the feature space into a higher dimension. In this work, we utilize the popular radial basis function (RBF) kernel, which is dependent on two hyperparameters: C and γ . Both influence the complexity of the final decision boundary, with C behaving as the penalty parameter for misclassification and γ as the “width” of the kernel. The larger C and γ are, the more complicated the resulting decision boundary will be (see [Bobra & Couvidat 2015](#); [Goodwin et al. 2024](#) for more details). The RBF kernel is defined by the following formula: $K(x, x') = \exp(-\gamma||x - x'||^2)$, where x and x' are feature vectors ([Ding et al. 2021](#)). The kernel, in essence, calculates the Euclidian distance between two points, mapping the data into a higher dimensional parameter space to better capture the similarity between points.

To ensure that we achieve clear-cut results on the impacts of projection effects on forecasting, we structure our feature selection methodology to amplify potential improvements from corrections. We calculate the Fisher score (F-score) of the 264 descriptive magnetogram parameters for all corrected and uncorrected training partitions. The F-score is mathematically defined in the following way ([Chen & Lin 2006](#); [Gu et al. 2011](#)):

$$F(i) = \frac{(\bar{x}_i^+ - \bar{x}_i)^2 + (\bar{x}_i^- - \bar{x}_i)^2}{\frac{1}{n^+ - 1} \sum_{c=1}^{n^+} (\bar{x}_{c,i}^+ - \bar{x}_i)^2 + \frac{1}{n^- - 1} \sum_{c=1}^{n^-} (\bar{x}_{c,i}^- - \bar{x}_i)^2} \quad (2)$$

where, for a given feature i , \bar{x}_i^+ is the average value across flaring events, \bar{x}_i^- is the average value across non-flaring events, \bar{x}_i is the average value across all events, n^+ is the number of flaring events, and n^- is the number of non-flaring events. Features with large F-scores have flaring and non-flaring distributions that are highly separated with small standard deviations, meaning they are useful to include when making predictions. Once we calculate the F-scores within a given partition, we rank the parameters from highest to lowest depending on their score. However, we only include parameters that show an improvement to their F-score after corrections were applied. For example, when we run this methodology on partition 1, we find that `SAVNCPP_max` (the maximum value of the sum of the absolute value of the net current per polarity) achieves an F-score of 16,920.76 and 17,220.60 for the uncorrected and corrected versions respectively. This means we will include this parameter in the rankings. However, `ABSNJZH_max` (the max value of the absolute value of the net current helicity) achieves an F-score of 18,327.77 and 17,591.47 for the uncorrected and corrected versions respectively. Thus, we do not include this parameter in the rankings. While this may not reflect the most optimal feature selection methodology, this logic ensures that potential improvements from corrections do not go unnoticed. In Section 4.2, we explore an additional selection methodology that includes parameters with no improvement.

After getting our list of improved parameters, we select the 25 with the highest F-score to use in both our uncorrected and corrected data sets. Previously it was found that for point-in-time magnetogram forecasting, the number of features (when selected based on F-score) has minimal impact on the resulting skill scores (Goodwin et al. 2024). Thus, for this study, we settled on 25 features as this balances both performance and training time. This process was repeated for every new training partition used. Please see Table 3 for the complete list of features selected.

Table 3. The 25 parameters selected for each training partition. Parameters were only chosen if they improved their F-scores after corrections were applied. The rank highlights the parameter’s original order when all parameters were ranked by the highest F-score they achieved across both the corrected and uncorrected data. The relative improvement to the F-score after corrections were applied is shown in parentheses.

Partition 1			Partition 2			Partition 3			Partition 4			Partition 5		
Rank	Parameter	Rank	Parameter	Rank	Parameter	Rank	Parameter	Rank	Parameter	Rank	Parameter	Rank	Parameter	
3	SAVNCPP_max (1.8%)	16	ABSNJZH_gderivative_stddev (0.3%)	31	TOTUSJH_stddev (7.6%)	1	TOTUSJH_median (0.002%)	17	TOTUSJZ_last_value (1.8%)					
4	SAVNCPP_last_value (1.1%)	32	USFLUX_last_value (3.6%)	33	USFLUX_last_value (0.4%)	2	TOTUSJH_min (0.1%)	27	TOTUSJZ_max (0.5%)					
7	SAVNCPP_mean (1.0%)	33	USFLUX_median (2.5%)	35	USFLUX_mean (0.4%)	3	TOTUSJH_mean (0.02%)	33	SAVNCPP_min (0.5%)					
8	SAVNCPP_median (0.8%)	34	USFLUX_mean (2.6%)	36	USFLUX_median (0.2%)	10	TOTUSJZ_min (0.6%)	38	TOTUSJH_stddev (5.3%)					
9	ABSNJZH_stddev (0.2%)	35	USFLUX_max (3.8%)	37	USFLUX_max (0.8%)	11	TOTUSJZ_median (0.4%)	43	TOTIFY_stddev (1.2%)					
10	SAVNCPP_min (0.7%)	36	USFLUX_min (1.4%)	44	TOTFY_stddev (3.9%)	13	TOTUSJZ_mean (0.4%)	54	TOTUSJZ_stddev (11.5%)					
13	SAVNCPP_stddev (2.8%)	37	TOTUSJH_stddev (15.1%)	47	MEANPOT_min (10.5%)	17	USFLUX_last_value (1.3%)	57	MEANPOT_min (12.6%)					
14	TOTUSJH_median (1.1%)	38	TOTPOT_median (0.02%)	49	R_VALUE_min (5.1%)	21	USFLUX_mean (1.1%)	58	TOTPOT_mean (6.9%)					
15	TOTUSJH_max (1.1%)	40	TOTUSJH_var (24.9%)	50	USFLUX_stddev (16.9%)	22	USFLUX_median (0.9%)	59	R_VALUE_min (1.8%)					
16	TOTUSJH_mean (1.0%)	41	TOTBSQ_stddev (13.9%)	51	TOTUSJZ_stddev (18.8%)	24	USFLUX_max (3.6%)	60	TOTUSJZ_gderivative_stddev (1.1%)					
19	TOTUSJH_min (0.5%)	43	TOTFZ_min (8.6%)	53	TOTUSJH_var (3.3%)	31	TOTUSJH_max (1.7%)	61	USFLUX_stddev (22.0%)					
25	ABSNJZH_gderivative_stddev (0.9%)	44	TOTFZ_last_value (9.7%)	57	TOTFY_var (16.5%)	36	TOTBSQ_stddev (23.4%)	62	SHRGT45_min (28.6%)					
27	SAVNCPP_gderivative_stddev (0.4%)	45	TOTPOT_mean (3.7%)	58	SHRGT45_min (22.1%)	38	TOTFZ_min (4.9%)	63	SHRGT45_last_value (32.0%)					
30	TOTUSJZ_last_value (1.9%)	47	TOTFZ_mean (9.1%)	59	R_VALUE_mean (3.0%)	40	ABSNJZH_gderivative_stddev (1.4%)	64	SHRGT45_median (29.7%)					
31	TOTPOT_last_value (3.5%)	48	TOTFZ_median (9.2%)	62	R_VALUE_last_value (2.0%)	41	TOTFZ_last_value (2.9%)	65	SHRGT45_mean (29.6%)					
32	TOTUSJZ_min (3.5%)	50	USFLUX_var (89.5%)	63	R_VALUE_median (2.3%)	42	TOTFZ_mean (2.3%)	66	MEANGAM_min (45.9%)					
33	TOTUSJZ_median (3.4%)	51	USFLUX_stddev (29%)	64	SHRGT45_mean (22.7%)	43	TOTFZ_median (1.9%)	67	MEANSHR_min (18.1%)					
34	TOTUSJZ_mean (3.6%)	52	TOTFZ_max (8.6%)	65	SHRGT45_median (22.6%)	49	USFLUX_stddev (36.9%)	68	MEANSHR_last_value (22.7%)					
35	TOTUSJH_stddev (9.2%)	55	TOTBSQ_gderivative_stddev (1.2%)	66	SHRGT45_last_value (23.8%)	50	TOTFY_last_value (1.2%)	69	MEANSHR_mean (20.1%)					
36	TOTUSJZ_max (2.6%)	57	TOTBSQ_var (55.7%)	67	MEANSHR_min (14.5%)	51	TOTFY_median (0.9%)	70	MEANSHR_median (20.2%)					
43	TOTBSQ_stddev (4.8%)	60	TOTPOT_last_value (6.8%)	68	MEANGAM_min (34.5%)	52	TOTFY_mean (0.7%)	72	USFLUX_var (114.5%)					
45	TOTUSJH_var (9.7%)	61	TOTUSJZ_stddev (10.9%)	69	MEANSHR_mean (15.8%)	53	TOTFY_max (0.4%)	74	MEANGAM_last_value (59.3%)					
46	TOTFY_gderivative_stddev (3.0%)	63	R_VALUE_min (8.2%)	70	MEANSHR_median (15.9%)	54	MEANPOT_min (2.0%)	75	MEANGAM_median (56.7%)					
51	TOTBSQ_gderivative_stddev (1.3%)	65	R_VALUE_mean (6.3%)	71	MEANSHR_last_value (17.1%)	56	TOTFZ_gderivative_stddev (83.5%)	76	MEANGAM_mean (56.7%)					
53	TOTPOT_max (11.1%)	66	R_VALUE_last_value (5.6%)	73	MEANGAM_median (41.0%)	57	R_VALUE_min (6.7%)	79	SHRGT45_max (33.8%)					

Partition	Uncorrected	Corrected
1	$C: 10, \gamma: 0.0001$	$C: 0.01, \gamma: 0.01$
2	$C: 0.01, \gamma: \text{"scale"}$	$C: 0.1, \gamma: 0.0001$
3	$C: 100, \gamma: 0.0001$	$C: 10, \gamma: 0.0001$
4	$C: 0.001, \gamma: 0.01$	$C: 0.0001, \gamma: \text{"scale"}$
5	$C: 0.001, \gamma: 0.01$	$C: 0.1, \gamma: 0.0001$

Table 4. The final combination of C and γ for each training partition for the uncorrected and corrected data sets. The “scale” parameter for γ utilizes the inverse of the number of features times the variance of the feature vector.

Finally, to ensure that each ML model provides reasonable predictive performance, an exhaustive hyperparameter grid search was applied to all training partitions (both corrected and uncorrected) to optimize each model’s C and γ hyperparameter. For a given training partition, a model was generated for every possible combination of the following values for C (0.0001, 0.001, 0.01, 0.1, 1, 10, 100, 1000) and γ (“scale”, 0.0001, 0.001, 0.01, 0.1, 1, 10, 100, 1000). Where $\gamma = \text{“scale”}$ utilizes the inverse of the number of features times the variance of the feature vector. For each of these models, the `class_weight` hyperparameter was set to “balanced” (meaning no under/oversampling techniques were applied), and the `kernel` parameter was set to “rbf”¹. These models were then trained/tested using a stratified group three-fold cross-validation. This ensures that within the training and testing folds, no data overlaps between ARs, and a similar number of flaring and non-flaring events are present. The model that produced the highest true skill statistic score (see Section 3.4) was then selected for application to the full training dataset. The final hyperparameters for each training partition are listed in Table 4.

3.4. Performance Metrics

To analyze the performance of our models, we focus on four different metrics: true positive rate (TPR), false positive rate (FPR), true skill statistic (TSS), and Heidke skill score (HSS_2). They are each defined mathematically in the following way:

$$TPR = \frac{TP}{TP + FN}, \quad (3)$$

$$FPR = \frac{FP}{FP + TN}, \quad (4)$$

$$TSS = \frac{TP}{TP + FN} - \frac{FP}{FP + TN}, \quad (5)$$

$$HSS_2 = \frac{2 \times [(TP \times TN) - (FN \times FP)]}{(TP + FN) \times (FN + TN) + (TP + FP) \times (FP + TN)}, \quad (6)$$

¹ See <https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html> for more details.

where TP are true positives (the number of correctly predicted flaring events), TN are true negatives (the number of correctly predicted non-flaring events), FP are false positives (the number of non-flaring events predicted as flaring), and FN are false negatives (the number of flaring events predicted as non-flaring). The TPR can be thought of as the likelihood of correctly predicting that a flaring event will occur, while the FPR is the likelihood of predicting that a flaring event will occur when it does not. Taking the difference of these metrics gives us the TSS score. This metric ranges from -1 to +1, and provides insight into the improvement of a classifier over a random forecast. A score of 0 would be given to a classifier that provides purely positive, negative, or simply random forecasts. A score of +1 would be given to a classifier that is always right, and a score of -1 would be given to a classifier that is always wrong. This metric is frequently used in flare forecasting as it is impervious to class imbalance (Bobra & Couvidat 2015). Lastly, similar to TSS, HSS_2 also provides insight into a model’s performance improvement over a random forecast with scores ranging from -1 to +1. However, unlike TSS, HSS_2 is highly susceptible to class imbalance, with scores decreasing when the imbalance increases (see Figure 2 in Bobra & Couvidat 2015 and Figure 4 in Ahmadzadeh et al. 2021). Due to the nature of our data set, changes in the number of false positives and true negatives have a much greater impact on this score. Thus, it’s important to include both TSS and HSS_2 when analyzing performance to see the results holistically. If the reader would like to learn more about these metrics in the context of flare forecasting, please see Bloomfield et al. (2012), Bobra & Couvidat (2015), Ahmadzadeh et al. (2021), and Goodwin et al. (2024).

4. RESULTS AND DISCUSSION

In the following sections, we discuss how corrections change forecasts at different heliocentric angles (Section 4.1), how performance metrics change after corrections are applied (see Section 4.2), and what are some other potential methods that can be used to address projection effects (see Section 4.3).

4.1. *At What Heliocentric Angles Do Corrections “Work”?*

Gathering the results across all training/testing partition pairs allows us to evaluate the “good changes” and “bad changes” made to our forecasts following corrections. In this context, a “good change” equates to an originally incorrect forecast that was later corrected after removing projection effects. In contrast, a “bad change” is an originally correct forecast that is now incorrect after removing projection effects. In Figure 4, we highlight these changes as a function of the mean θ . Events whose predictions stayed the same between forecasts were not included. It is clear from the figure that the “bad changes” significantly outnumber the “good changes” near the central meridian. However, as we move toward the limb, they quickly balance out. Using the intersection of the kernel density estimation (see Weglarczyk 2018; Waskom 2021 for more details) for each distribution, we find that this occurs at $\approx 44.5^\circ$. This suggests two things: first, predicting with corrective data is more effective near the limb. Second, if we want to maximize performance, an ensemble-like forecaster must be applied. We suggest using the uncorrected data/classifier for predictions of ARs with θ s less than 44.5° and the corrected data/classifier for anything above this angle. This is explored further in Section 4.2.

In addition to looking at the data as a whole, we can dive into the flaring classes themselves to see those most affected by corrections. In Figure 5, we plot the same results from Figure 4, however, the data is now separated into the “good” (left column) and “bad” (right column) changes and color-

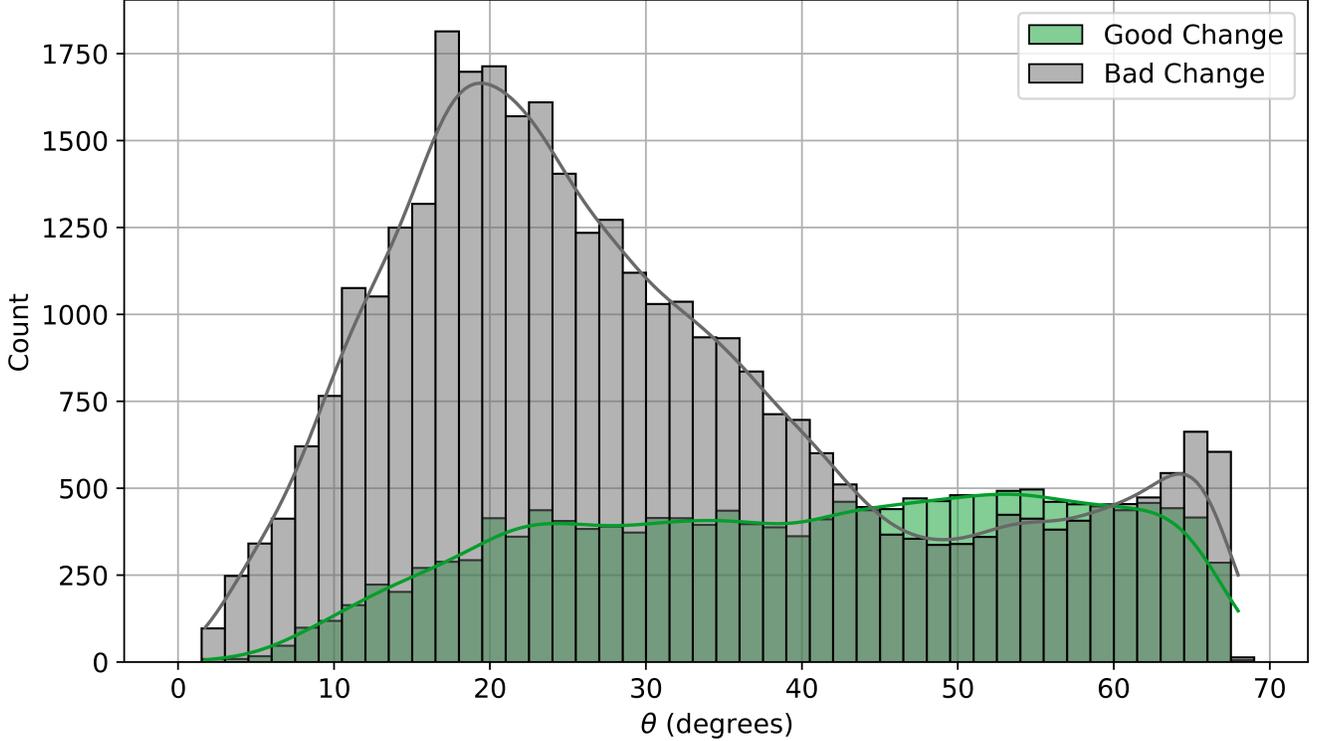


Figure 4. A histogram illustrating how removing projection effects impacts the predictions made at different average heliocentric angles (θ) across all training/testing partition combinations. Bars are color-coded to indicate if an incorrect forecast was corrected (Good Change) or if a correct forecast became incorrect (Bad Change) after projection effects were removed. A solid line representing the kernel density estimation of each distribution is plotted as well.

coded based on the non-flaring (top row) and flaring (bottom row) events that occur at a specific mean θ . From this plot, we can see that the M- and X-class flares tend to have more “good” than “bad” changes after corrections are applied, with the opposite observed for the non-flaring data. Thus, we can expect to see increases in both the true positive and false positive rates when analyzing the performance metrics in Section 4.2. However, we expect this increase to be quite minimal. This is because the changes in our predictions occur for only a small subset of our data. Analyzing the results across every possible partition combination we find that, on average, “good changes” occur to the flaring data around $5.1\% \pm 6.8\%$ of the time, with “bad changes” around $1.3\% \pm 2.1\%$. For the non-flaring data, “good changes” occur around $1.1\% \pm 1.0\%$ of the time, with “bad changes” around $2.7\% \pm 1.4\%$. Unfortunately, there is no clear explanation for why these changes occur the way they do. However, we speculate that an improvement in M- and X-class forecasting may be due to the activity of the selected “benchmark” ARs. We find that around 37% of these ARs produce at least one M- or X-class flare during their lifetime. Additionally, these ARs make up 51% of the total number of flaring ARs within SWAN-SF. Since we have such a large representation of flaring ARs when calibrating our corrections, this could be a reasonable explanation for the phenomenon observed here.

4.2. Analysis Of Performance Metrics For Each Training-Testing Partition

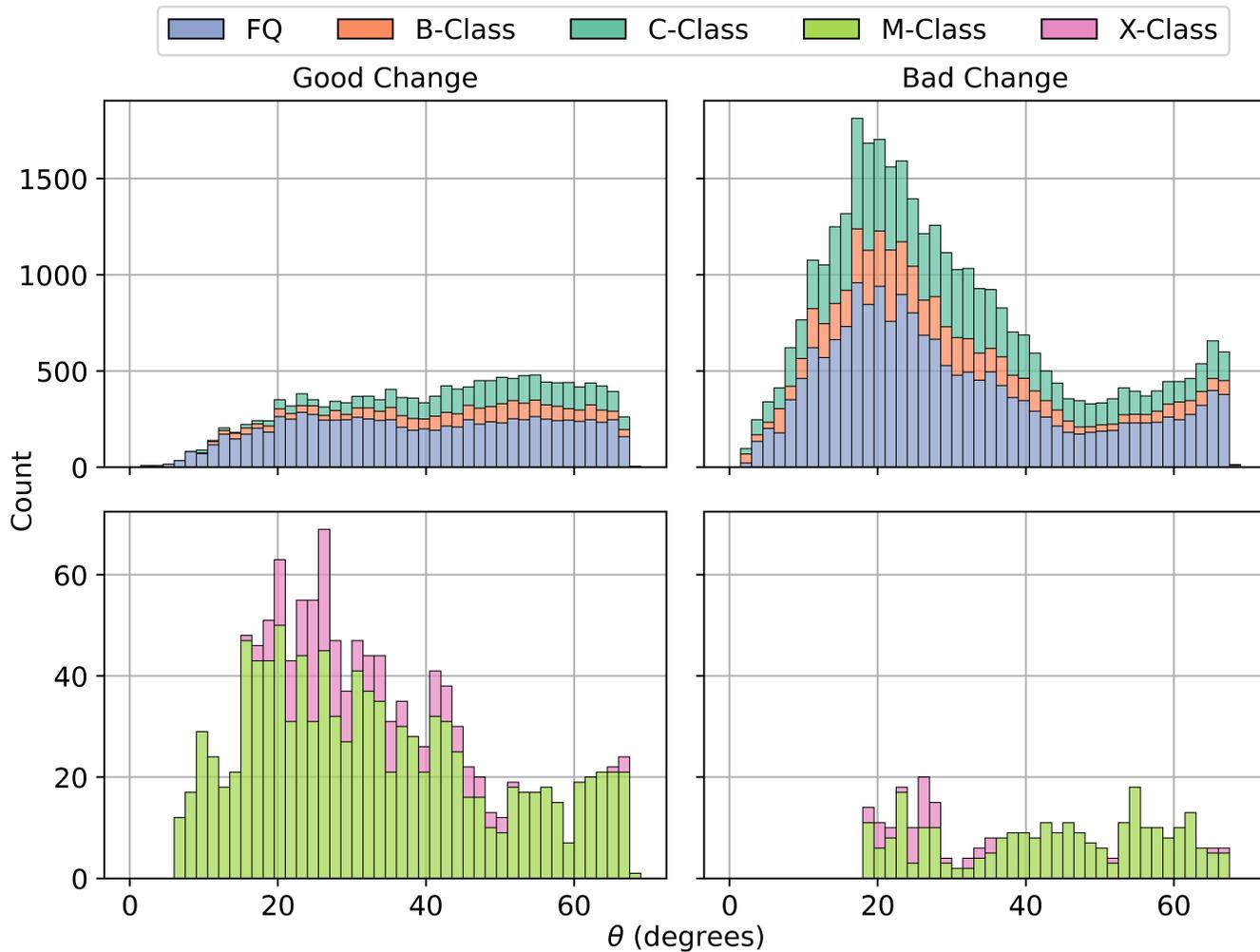


Figure 5. A stacked histogram highlighting the “good” (left column) and “bad” (right column) changes to forecast after removing projection effects as a function of heliocentric angle (θ). These data come from the collection training/testing partition experiments. Results are color-coded based on flaring class. Flare quiet (FQ), B-, and C-class events are shown in the top panels. M- and X-class events are shown in the bottom panels.

Now, let us analyze the changes to performance metrics, specifically TPR, FPR, TSS, and HSS₂. Figure 6 highlights our results. The scatter plots on the left show the scores for each partition combination, while the box plots on the right summarize the findings across all combinations. Results are color-coded to indicate whether the classifier was trained/tested with uncorrected data (grey), corrected data (maroon), or a combination of the two (salmon pink - “44.5° Cut”). As was previously mentioned, corrections appear to be the most useful beyond mean θ s of 44.5°. Thus, the “44.5° Cut” results use the uncorrected forecasts below mean θ s of 44.5° and the corrected forecasts above this angle. From the figure, we find that on average, removing projection effects increases both the TPR and FPR by a few percent. In turn, this produces an enhancement in the TSS scores but a reduction in the HSS₂ scores. Additionally, it is interesting to note that removing projection effects reduces the overall data spread, with the interquartile range decreasing for both the TPR and FPR.

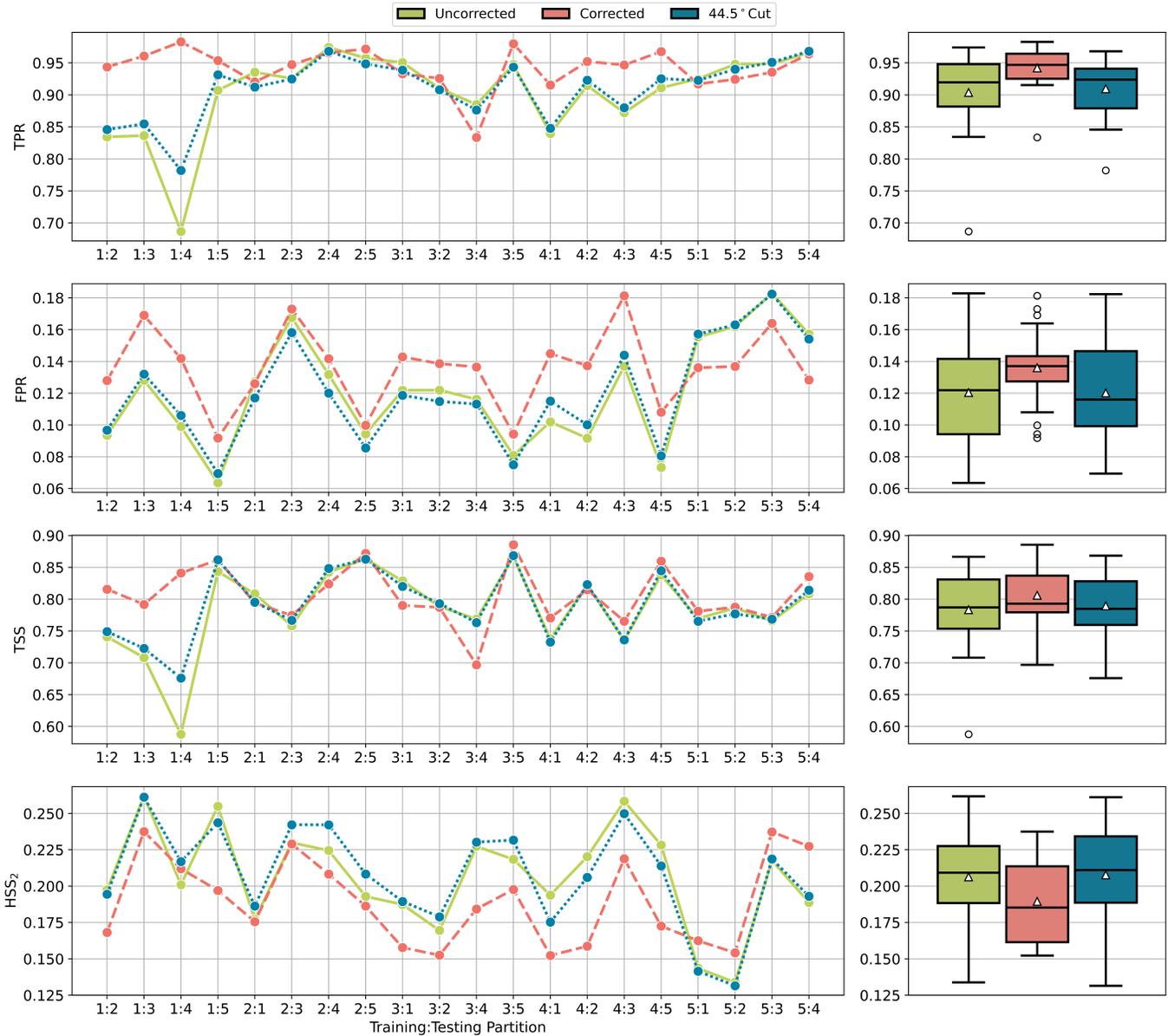


Figure 6. The true positive rate (TPR), false positive rate (FPR), true skill statistic (TSS), and Heidke skill scores (HSS_2) for every training and testing partition pair from SWAN-SF (denoted training:testing). Results are color-coded to indicate whether the classifier was trained and tested with uncorrected data (green), corrected (coral), or a combination of the two (blue - “44.5° Cut”). The “44.5° Cut” results use the uncorrected forecasts below heliocentric angles of 44.5° and the corrected forecasts above this angle. Within the box plots, the triangle represents the data’s mean, while the circles highlight outliers. The outliers are defined as any value lying 1.5 times the interquartile range above the third quartile or below the first quartile.

This suggests we generally get more consistent results after corrections, which is beneficial from a forecaster’s perspective. Looking at the “44.5° Cut”, which should maximize our performance improvements, we see an ever-so-slight increase to both the average TSS and HSS_2 scores.

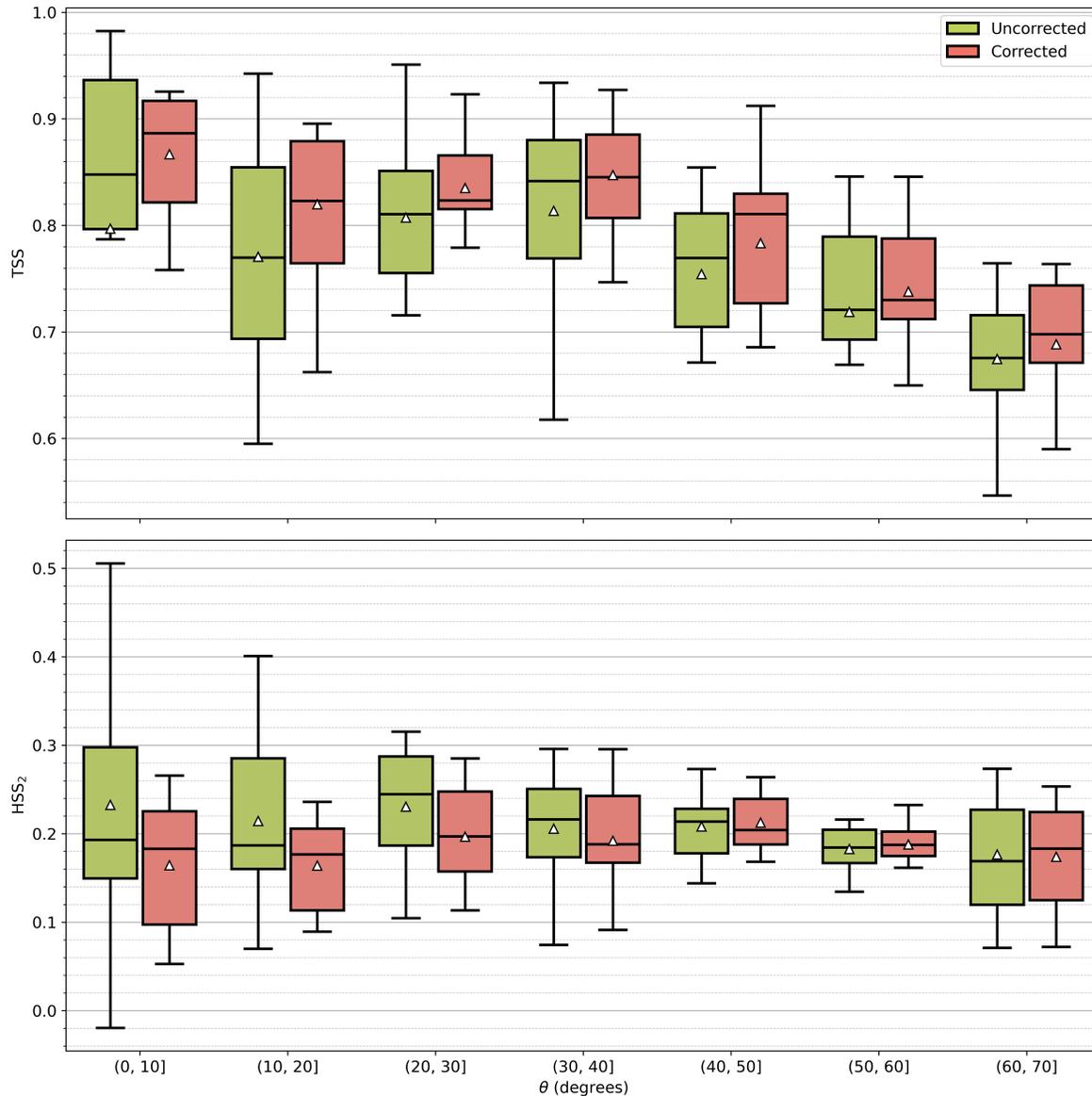


Figure 7. The TSS and HSS_2 scores for each training/testing partition pair for the uncorrected (green) and corrected (coral) data as a function of heliocentric angle (θ). Results are grouped into 10-degree bins. The triangles indicate the mean of the data. Outliers have been removed and are not shown in the plot.

We want to emphasize that although we observe changes in our performance metrics when correcting our data, the hyperparameters we select to train our model can significantly impact these variations. With only a slight tweak to C and γ , our findings may be completely altered, decreasing the true positive and false positive rates. This leads to a classifier with somewhat better HSS_2 scores but worse TSS scores after corrections. *Nonetheless, the key takeaway should be that correcting for projection effects has only a small impact on the achieved skill scores.* This fact has some potentially profound consequences.

First, it may suggest that the correction methodology utilized in this work is not sophisticated enough. If we take a look at how the TSS and HSS_2 scores change as a function of θ (Figure 7),

we see that near the central meridian, at $(0, 10]$, TSS scores hover at ≈ 0.85 , but slowly drop to ≈ 0.67 when moving toward the limb. Once corrections are applied, this drop-off still exists, albeit slightly reduced, suggesting that there is still room for improvements from our corrections. For HSS_2 scores, very little change is observed when approaching the limb, suggesting there is not much room for improvement here. A potential remedy to this problem could be to utilize more fundamental corrections targeting the magnetograms themselves. For example, [Leka et al. \(2022\)](#) implemented several advanced spectropolarimetric inversion techniques to remove biases within the data. It would be interesting to see if applying similar methods to the magnetograms collected for SWAN-SF would provide greater improvements to forecasting.

Second, and perhaps the least desirable interpretation of this outcome, is that the magnetogram data itself is limited in its predictive capabilities for flaring events. Given that applying scaling factors to the data set had minimal effect on the results indicates that classifiers tend to rely on very generalized associations when making predictions. This suggests that additional data sources, such as EUV images ([Nishizuka et al. 2018](#); [Alipour et al. 2019](#); [Krista & Chih 2021](#); [Leka et al. 2023](#)) and spectra ([Panos & Kleint 2020](#); [Woods et al. 2021](#); [Huwlyer & Melchior 2022](#); [Panos et al. 2023](#)) used in recent forecasting works, need to be implemented in conjunction with magnetograms to progress the field further.

As previously mentioned, our feature selection methodology is not necessarily optimized for flare forecasting. Instead, we chose parameters with the greatest potential to maximize performance improvements when applying corrections. Nonetheless, we also wanted to investigate how performance changes when selecting the most effective features for forecasting. Here, we carry out the same feature ordering procedure mentioned in Section 3.3, however, we now select all of the top 25 ranking features, not just the ones that show improvements to their F-score. Like before, the uncorrected dataset uses only the uncorrected features, however, the corrected classifier may now use some combination of corrected and uncorrected parameters. This is most relevant for partitions 1 and 4, where 12 and 10 of the chosen features, respectively, have been corrected. For partitions 2 and 5, only 1 corrected feature is used, and for partition 3, none of the corrected features are used. Our results are shown in Figure 8. Here, we see a decent discrepancy between the new uncorrected and corrected TSS scores when using partition 1 for training, however, this is not observed in partition 4 or any other partitions. This again emphasizes our point that projection effect corrections have little impact on performance, or at the very least, results seem to be dependent on the partition utilized. Comparing the use of the top 25 features versus the original feature selection methodology, we see that the performance differences are fairly minimal, with the new methodology only slightly outperforming the old. This once again reinforces the idea that there may be some serious limitations when utilizing point-in-time magnetogram data for forecasting.

4.3. *How Else Can We Consider Projection Effects?*

Aside from directly manipulating the data, we address projection effects in two additional ways. First, we employ an ensemble-based forecaster dependent on the mean θ of ARs. This involves training three SVM classifiers on three different θ ranges: $[0^\circ - 23.33^\circ]$, $(23.33^\circ - 46.66^\circ]$, and $(46.66^\circ - 70.00^\circ]$. Within each θ range, we are minimizing the long-term θ -dependent projection effect trends within a particular training data set, which, in a way, removes the impacts of these effects without altering the data itself. However, we are also removing the number of flaring samples used to train an individual model, which is a huge downside. When making forecasts, the appropriate model is then

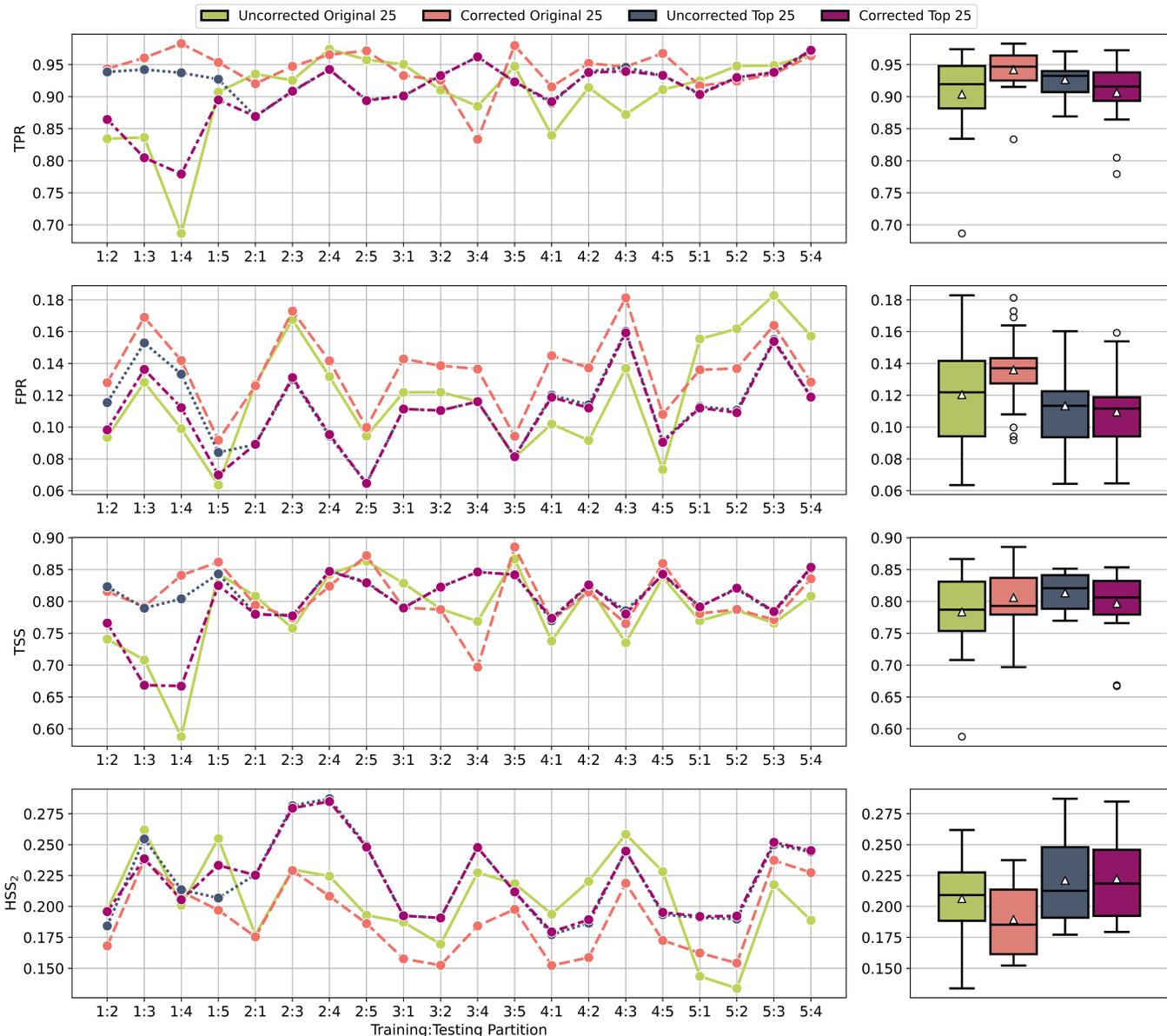


Figure 8. The true positive rate (TPR), false positive rate (FPR), true skill statistic (TSS), and Heidke skill scores (HSS_2) for every training/testing partition combination from SWAN-SF. Results are color-coded to indicate whether the classifier was trained/tested with uncorrected data with the original 25 features (green), the corrected data with the original 25 features (coral), the uncorrected data with the top 25 features (blue), or the corrected data with the top 25 features (purple). See the caption in Figure 6 for the meaning of the triangles and circles in the box plots.

selected based on the location of the observed AR. For each classifier, we utilize the same feature selection and hyperparameter tuning methodology mentioned in Section 3.

We also test an approach that retrain the uncorrected classifiers with the mean θ as a feature. This ensures our model considers the location of the AR when making a forecast. These classifiers were trained using the same selected features (with the addition of the mean θ) and hyperparameters originally chosen for the uncorrected data sets.

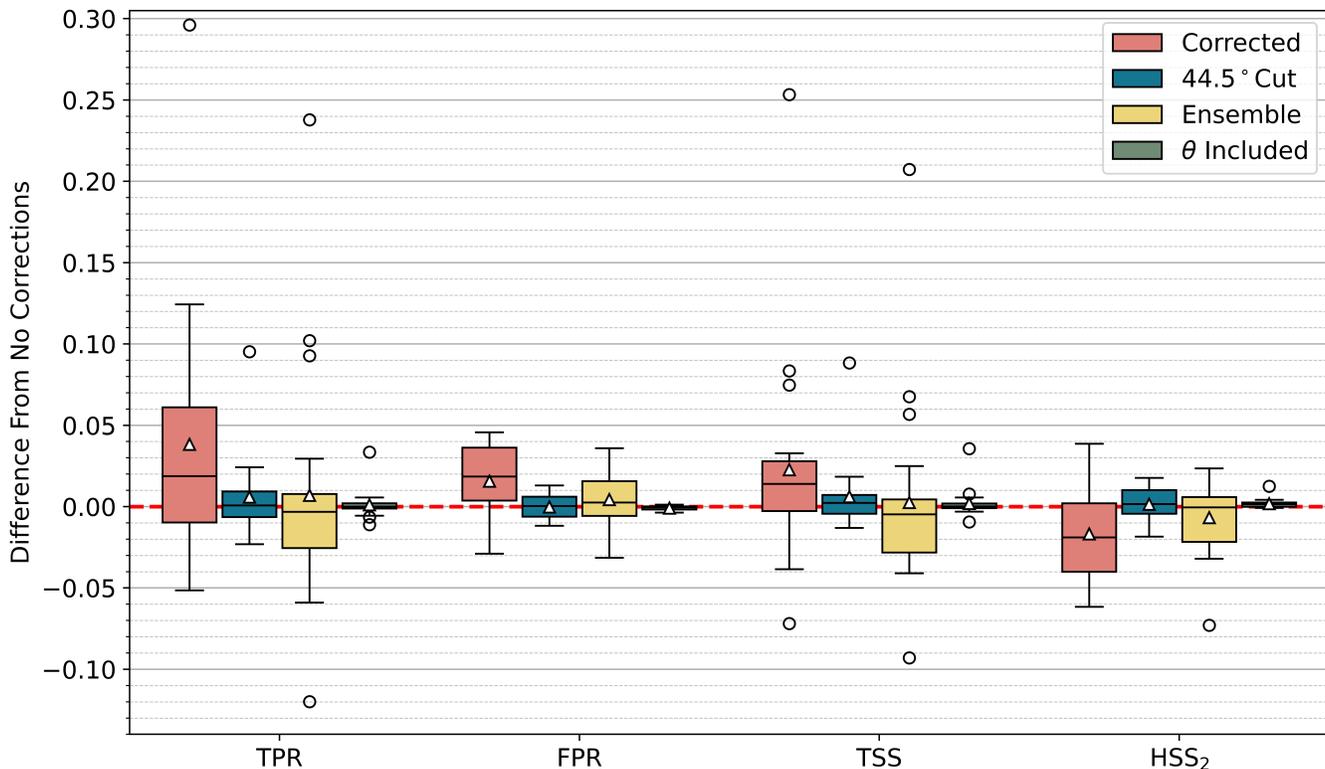


Figure 9. The true positive rate (TPR), false positive rate (FPR), true skill statistic (TSS), and Heidke skill score (HSS₂) differences from the uncorrected approach for every training/testing partition combination from SWAN-SF. Positive values indicate an increase in the score from the baseline of no corrections. Negative values indicate a decrease from the baseline. Results are color-coded to indicate whether the classifier was trained/tested with the corrected data (coral), the “44.5° Cut” methodology (blue), the ensemble methodology (yellow), or the uncorrected data with the mean heliocentric angle (green). The red dashed line emphasizes no difference from the uncorrected classifier. See the caption in Figure 6 for the meaning of the triangles and circles in the box plots.

The results for both tests are shown in Figure 9. Here we display a boxplot noting the TPR, FPR, TSS, and HSS₂ deviations from no corrections for the corrected data (maroon), the “44.5° Cut” methodology (salmon pink), the ensemble methodology (teal), and the uncorrected data plus mean θ (dark blue). These differences are calculated across the same training/testing partition pairs. Once again, our alternate methods for considering projection effects have little impact on performance. The ensemble methodology shows a slight increase in TSS scores with a complimentary decrease in HSS₂. The changes are on the order of only a few percent. Finally, when including the mean θ as an additional feature to the uncorrected classifier, performance is nearly unchanged across all tested metrics.

5. SUMMARY

Vector magnetograms suffer from projection effects as active regions migrate toward the limb. This can cause problems in solar flare forecasting, as active regions may appear stronger or weaker than they truly are due to artificial long-term θ -dependent trends present within the data. In this work,

we perform several correction techniques to remedy this problem and study their impact on flare forecasting. From our analysis, we have made the following conclusions:

1. Correcting for projection effects using a data-centered methodology increases both the true positive and false positive prediction rates. However, these changes are minimal.
2. Beyond heliocentric angles of 44.5° , “good” and “bad” changes to our forecasts after corrections are applied tend to level out. If we would like to maximize performance improvements, a combination of the corrected and uncorrected classifiers should be used. For data with mean heliocentric angles below 44.5° , the classifier trained on uncorrected data should be used. For heliocentric angles above 44.5° , the corrected classifier should be used.
3. Classifier-based correction methodologies showed no clear improvements over data-centered corrections.
4. A more sophisticated correction methodology is likely needed to see greater improvements in our results.
5. These findings could indicate significant limitations in using magnetogram data for flare forecasting. This further emphasizes the need for new data sources and prediction techniques to continue the development of the space weather forecasting community.

Finally, it could be intriguing to approach this problem from a modeling perspective in the future. Simulating an active region and feeding it through the Solar Dynamics Observatory Helioseismic and Magnetic Imager pipeline (similar to [Kitiashvili et al. 2015](#)) would allow us to better constrain the projection effects for each magnetic field parameter. We could force the simulated active regions into a particular shape, size, and strength, allowing us to generate a more robust selection of “benchmark” active regions.

6. ACKNOWLEDGEMENTS

We want to thank the anonymous reviewer for their valuable comments, which greatly improved the clarity of the text. This work was supported by NASA FINESST grant 80NSSC23K1639, NASA LWS grant 80NSSC22K0272, and NSF FDSS grant 1936361. Viacheslav Sadykov acknowledges the NASA COFFIES DSC grant NNH18ZDA001N-DRIVE.

REFERENCES

- | | |
|---|--|
| <p>Ahmadzadeh, A., Aydin, B., Georgoulis, M. K., et al. 2021, <i>The Astrophysical Journal Supplement Series</i>, 254, 23, doi: 10.3847/1538-4365/abec88</p> <p>Alipour, N., Mohammadi, F., & Safari, H. 2019, <i>The Astrophysical Journal Supplement Series</i>, 243, 20, doi: 10.3847/1538-4365/ab289b</p> <p>Angryk, R., Martens, P., Aydin, B., et al. 2020a, SWAN-SF, V1, Harvard Dataverse, doi: 10.7910/DVN/EBCFKM</p> | <p>Angryk, R. A., Martens, P. C., Aydin, B., et al. 2020b, <i>Scientific data</i>, 7, 227</p> <p>Badillo, S., Banfai, B., Birzele, F., et al. 2020, <i>Clinical pharmacology & therapeutics</i>, 107, 871</p> <p>Bloomfield, D. S., Higgins, P. A., McAteer, R. J., & Gallagher, P. T. 2012, <i>The Astrophysical Journal Letters</i>, 747, L41</p> <p>Bobra, M. G., & Couvidat, S. 2015, <i>The Astrophysical Journal</i>, 798, 135</p> <p>Chen, Y.-W., & Lin, C.-J. 2006, <i>Feature extraction: foundations and applications</i>, 315</p> |
|---|--|

- Ding, X., Liu, J., Yang, F., & Cao, J. 2021, *Journal of the Franklin Institute*, 358, 10121
- Falconer, D. A., Tiwari, S. K., Moore, R. L., & Khazanov, I. 2016, *The Astrophysical Journal Letters*, 833, L31
- Goodwin, G. T., Sadykov, V. M., & Martens, P. C. 2024, *The Astrophysical Journal*, 964, 163, doi: [10.3847/1538-4357/ad276c](https://doi.org/10.3847/1538-4357/ad276c)
- Gu, Q., Li, Z., & Han, J. 2011, in *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence, UAI'11* (Arlington, Virginia, USA: AUAI Press), 266–273
- Hoeksema, J. T., Liu, Y., Hayashi, K., et al. 2014, *Solar Physics*, 289, 3483
- Huwlyer, C., & Melchior, M. 2022, *Astronomy and Computing*, 41, 100668, doi: [10.1016/j.ascom.2022.100668](https://doi.org/10.1016/j.ascom.2022.100668)
- Kitiashvili, I. N., Couvidat, S., & Lagg, A. 2015, *The Astrophysical Journal*, 808, 59, doi: [10.1088/0004-637X/808/1/59](https://doi.org/10.1088/0004-637X/808/1/59)
- Krista, L. D., & Chih, M. 2021, *The Astrophysical Journal*, 922, 218, doi: [10.3847/1538-4357/ac2840](https://doi.org/10.3847/1538-4357/ac2840)
- Leka, K., Dissauer, K., Barnes, G., & Wagner, E. L. 2023, *The Astrophysical Journal*, 942, 84
- Leka, K., Wagner, E. L., Griñón-Marín, A. B., Bommier, V., & Higgins, R. E. 2022, *Solar Physics*, 297, 121
- Liu, Y., Hoeksema, J., Scherrer, P., et al. 2012, *Solar Physics*, 279, 295
- Nishizuka, N., Sugiura, K., Kubo, Y., Den, M., & Ishii, M. 2018, *The Astrophysical Journal*, 858, 113
- Panos, B., & Kleint, L. 2020, *The Astrophysical Journal*, 891, 17, doi: [10.3847/1538-4357/ab700b](https://doi.org/10.3847/1538-4357/ab700b)
- Panos, B., Kleint, L., & Zbinden, J. 2023, *Astronomy & Astrophysics*, 671, A73, doi: [10.1051/0004-6361/202244835](https://doi.org/10.1051/0004-6361/202244835)
- Scherrer, P. H., Schou, J., Bush, R., et al. 2012, *Solar Physics*, 275, 207
- Waskom, M. L. 2021, *Journal of Open Source Software*, 6, 3021, doi: [10.21105/joss.03021](https://doi.org/10.21105/joss.03021)
- Weglarczyk, S. 2018, *ITM Web Conf.*, 23, 00037, doi: [10.1051/itmconf/20182300037](https://doi.org/10.1051/itmconf/20182300037)
- Woods, M. M., Dalda, A. S., & Pontieu, B. D. 2021, *The Astrophysical Journal*, 922, 137, doi: [10.3847/1538-4357/ac2667](https://doi.org/10.3847/1538-4357/ac2667)