Forecasting solar flares using magnetogram-based predictors and machine learning

Kostas Florios

Research Center for Astronomy and Applied Mathematics, Academy of Athens

Weekly Seminar in Astronomy and Applied Mathematics,
RCAAM/AA, November 29, 2016

Outline

- Forecasting solar flares using NASA Solar Dynamics Observatory (SDO)
- Predictors for solar Active Regions (AR), using available magnetograms, taken by Helioseismic and Magnetic Imager (HMI)
- Space-weather HMI Active Region Patches (SHARPs) with cut-out magnetograms in near real-time
- Predictors used: decay index, B effective, Non-neutralized Current etc.
- Predictor values for a representative sample of SHARP data (2012-2016)
- Machine Learning (ML) in order to predict Solar Flares Occurence

Outline (contd.)

- We predict >M1 class flares and C-class flares in the next 24h after the SHARP dataset has been issued
- Prediction methods are Multi-Layer Perceptrons, Support Vector Machines and Random Forests
- Comparison to statistical approaches
- Random Forests are the prediction technique of choice, while the second best method is the Multi-Layer Perceptron
- A Monte Carlo showed that the best performing method is Random Forest with
  - ACC=0.98±0.00, TSS=0.61±0.05 and HSS=0.50±0.04 (>M1 class)
  - ACC=0.88±0.00, TSS=0.63±0.02 and HSS=0.51±0.02 (C-class)
- We use fifteen B_{los} and B_r based magnetogram predictors

Notation for flares and Instrumentation

- The Geostationary Operational Environmental Satellite operated by USA National Oceanic and Atmospheric Administration (NOAA) constantly monitors the Sun.
- Based on these measurements, solar flares are categorized in classes, namely A,B,C,M,X (in increasing strength).
- Each class is ten times stronger than the previous.
- Observable effects are produced for class C and above, while M and X are the most energetic.
- The HMI (Scherrer et al., 2012) instrument onboard the SDO (Pesnell et al., 2012) is producing, in near-real time, the SHARP Product (M. Bobra et al., 2014) which is used in this study

Machine Learning and the FLARECAST project

Several researchers have recently used machine learning techniques aiming to effectively forecast solar flares (M. G. Bobra & Couvidat, 2015; Boucheron, Al-Ghraibah, & McAteer, 2015; Al-Ghraibah, A., Boucheron, L. E., & McAteer, R. T. J., 2015; Ahmed et al., 2013; Li, Cui, He, & Wang, 2008; Song et al., 2009; Yu, Huang, Wang, & Cui, 2009; Yuan, Shih, Jing, & Wang, 2010; Colak & Qahwaji, 2009; Wang, Cui, Li, Zhang, & Han, 2008).

- The analysis presented here is part of the HORIZON2020 FLARECAST project.
- Predictors calculated from SHARP data combined with ML algorithms in order to forecast flare events within a 24h window for C and $>$M1 classes.
- The prediction is binary, meaning that a given flare class is considered to either happen or not during the 24h following the prediction.

Sampling and Contribution

Sampling

Representative sample which includes 25% of the days in the years (2012-2016) at a cadence of 6h.

Contribution

1. The utilization of novel magnetogram-based predictors in a multi-parameter Solar Flares Occurrence prediction model.
2. The utilization of machine learning techniques, such as MLP, SVM and especially, for first time, Random Forests (RF) using the predictor variables to forecast Solar Flares Events with $>$M1 class and C-class.

Predictors

Multiple predictors calculation procedures were employed to compute values of these predictors starting from SDO/HMI/SHARP cut-out magnetograms.

Active Regions

Predictors characterize Active Regions (AR) of the Sun and the generated Flares Events are always associated with a given Active Region of the Sun.

- Magnetic polarity inversion line
- Decay index
- B effective
- Non-neutralized Current
- Ising energy
- Fourier spectral power index
- Schrijver's R value

Visualization of Active Regions

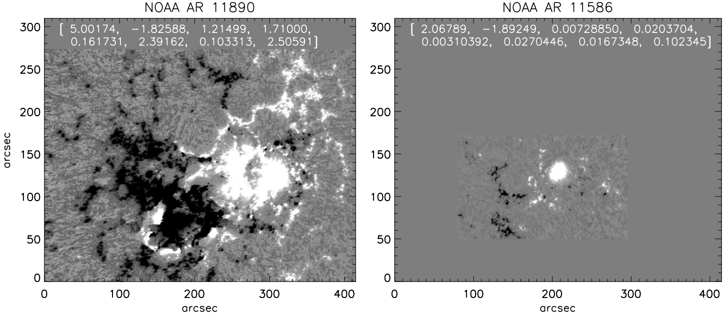


Figure: Two SHARP frames depicting AR with different flaring activity within the 24 h time window. NOAA AR 11890 (left) produced 4 C-, 2 M- and 1 X-class flares while NOAA AR 11586 (right) produced none. The two AR are scaled so as to retain their original relative size and, for comparison, the vectors of the eight predictors are also plotted on the frames. The names of all $K = 8$ predictors [logR, FSPI, TLMPIL, DI, Beff, IsinEn1, IsinEn2, NNC] are defined in Section 2. High values of the predictors indicate a powerful AR (left) and low values of the predictors indicate a quiet AR (right).

Flares association with Active Regions

- We developed an algorithm that searches for X-ray solar flares observed with GOES Satellites operated by NOAA.
- The algorithm finds GOES X-ray flares in the next 24h from a given SHARP property measurement:
  1. by matching NOAA AR numbers of AR properties with those of flares if NOAA AR numbers are available, and also
  2. by comparing the longitude and latitude ranges of ARs with locations of flares if NOAA AR numbers are not given.
- AR's rotation due to the differential solar rotation is also considered.

Binary prediction - Contingency Tables

Table: Contingency table 2 by 2

	ACTUAL	
PREDICT	NO	YES
NO	TN	FN
YES	FP	TP

Table: Contingency table 2 by 2 (alternative symbols)

	ACTUAL	
PREDICT	NO	YES
NO	d	c
YES	b	a

In what follows, let

- *ACC* denote Accuracy;
- *TSS* denote True Skill Statistic (Hanssen & Kuipers, 1965);
- *HSS* denote Heidke Skill Score (Heidke, 1926);

where TP=true positive, TN=true negative, FP=false positive, FN=false negative, and n=sample size, we have the following definitions:

Definition (Accuracy Metric)

$$Accuracy, ACC = (TN + TP)/n$$

The meaning of ACC is the Proportion Correct. PC = (correct forecasts of both event and non-event) / (total sample size).

Binary prediction - Quality Metrics (contd.)

Definition (TSS Metric)

$$True\ Skill\ Statistic,\ TSS = \frac{a}{(a+c)} - \frac{b}{(b+d)}$$

If we define H=POD=a/(a+c) and F=POFD=b/(b+d), then simply the vertical distance in the H-F plane is the TSS=H-F. The TSS has a range of -1 to +1, with 0 representing no skill. Negative values would be associated with "perverse" forecasts.

Definition (HSS Metric)

$$Heidke\ Skill\ Score,\ HSS = 2 \cdot \frac{(a \cdot d - b \cdot c)}{((a+c)\cdot(c+d) + (a+b)\cdot(b+d))}$$

The HSS measures the fractional improvement over the standard forecast. The "standard forecast" is usually the number correct by chance. The range of the HSS is $-\infty$ to 1. A perfect forecast obtains a HSS of 1.

- Let l inputs I_l and bias B_1.
  Let j hidden nodes H_j and bias B_2.
  Let i outputs O_i. Typically $i = 1$.
- The statistical model of a MLP neural network for binary outcome is based on (MacKay, 2003).
- The *multilayer perceptron* is a feedforward network. It has **input** neurons, **hidden** neurons and **output** neurons. The hidden neurons may be arranged in a sequence of layers.
- Such a feedforward network defines a nonlinear parameterized mapping from input $\boldsymbol{x}$ to an output $\boldsymbol{y} = \boldsymbol{y}(\boldsymbol{x}; \boldsymbol{w}, \boldsymbol{A})$. The output is a continuous function of the input and of the parameters $\boldsymbol{w}$; the architecture of the net, i.e., the functional form of the mapping, is denoted by $\boldsymbol{A}$.

In the case of classification, the mapping for a network with one hidden layer has the form:

$$\begin{aligned} a_j^{(1)} &= \sum_{l=1}^{L} w_{jl}^{(1)} x_l + \theta_j^{(1)}; \quad h_j = f^{(1)}(a_j^{(1)}) \\ a_i^{(2)} &= \sum_{j=1}^{J} w_{ij}^{(2)} h_j + \theta_i^{(2)}; \quad y_i = f^{(2)}(a_i^{(2)}) \end{aligned} \qquad \text{(MLP)}$$

where for example $f^{(1)}(a) = \frac{1}{(1+exp(-a))}$ and $f^{(2)}(a) = \frac{1}{(1+exp(-a))}$.

- Here l runs over the inputs $x_1, \ldots, x_L$, j runs over the hidden units, and i runs over the outputs
- The 'weights' w and 'biases' θ together make up the parameter vector $\boldsymbol{w}$
- The sigmoid function $f^{(1)}$ at the hidden layer gives the neural network greater flexibility than linear classification

MLP illustration

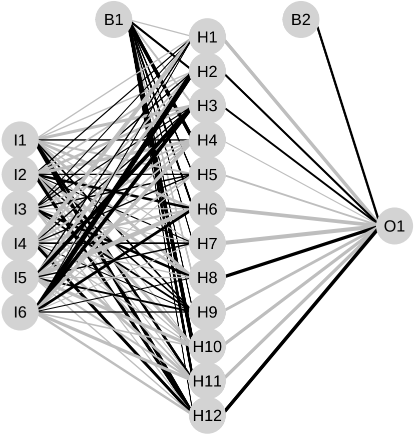


Figure: MLP neural network with 6 inputs, 12 hidden nodes, 1 output. Bold lines indicate large positive weights $\boldsymbol{w}$.

How a classification network is traditionally trained

This network is trained using a dataset $D = \{\boldsymbol{x^{(n)}}, \boldsymbol{t^{(n)}}\}$ by adjusting $\boldsymbol{w}$ so as to minimize a negative log-likelihood e.g.

$$G(\boldsymbol{w}) = -\Big(\sum_{n=1}^{N} \boldsymbol{t}^{(n)} \cdot ln(\boldsymbol{y}(\boldsymbol{x}^{(n)}; \boldsymbol{w})) + (1 - \boldsymbol{t}^{(n)}) \cdot ln(1 - \boldsymbol{y}(\boldsymbol{x}^{(n)}; \boldsymbol{w}))\Big) \tag{1}$$

The function G is the negative log-likelihood and it is minimized with respect to $\boldsymbol{w}$.

The Support Vector Machines (SVM) variant we have used is the ε-Support Vector Regression (ε-SVR) according to the widely used library LIBSVM (Chang & Lin, 2011; Meyer et al., 2003).

Let us consider a set of training points, $(x_1, z_1), \ldots, (x_\ell, z_\ell)$, where $x_i \in R^n$ is a feature vector and $z_i \in R^1$ is the target output. Using given parameters $C > 0$ and $\varepsilon > 0$, the standard form of support vector regression (e.g. see Vapnik, 1998) is

$$\text{minimize} \quad \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{i=1}^{\ell}\xi_i + C\sum_{i=1}^{\ell}\xi_i^*$$

subject to:

$$\begin{aligned} \boldsymbol{w}^T\phi(\boldsymbol{x}_i) + b - z_i &\leq \varepsilon + \xi_i, \qquad i = 1, 2, \ldots, \ell, \\ z_i - \boldsymbol{w}^T\phi(\boldsymbol{x}_i) - b &\leq \varepsilon + \xi_i^*, \qquad i = 1, 2, \ldots, \ell, \\ \xi_i, \xi_i^* &\geq 0, \qquad i = 1, 2, \ldots, \ell. \end{aligned} \tag{ε-SVR}$$

- The optimization in (ε-SVR) model is performed by changing the decision variables: $\boldsymbol{w}$,b,ξ and ξ^*.
- Actually, LIBSVM solves the dual of (ε-SVR) which depends on a quantity: $Q_{ij} = K(x_i, x_j) = \phi(\boldsymbol{x_i})\phi(\boldsymbol{x_j})$. which is called *kernel* of the SVM.
- We have used the Radial Basis Function (RBF) (or Gaussian) kernel which is defined as: $K(x, x') = exp(-\gamma||x - x'||^2)$.

Random Forests

The pseudocode of the Random Forest approach is as follows, taken from (Bessonov, 2016):
Randomly build an ensemble of trees

1. **Bootstrap** a sample of data, start building a tree
2. **Create a node** by
    1. **Randomly** selecting m variables from M
    2. Keep **m** constant (except for terminal nodes)
3. **Split** the node based on **m** variables
4. **Grow** a tree until no more splits are possible
5. **Repeat steps 1-4 n times**
   $\rightarrow$ Generate an **ensemble of trees**
6. **Calculate** variable importance for each predictor variable **X**

Random Forests (contd.)

- The Random Forest (RF) is a novel Machine Learning methodology, introduced by Breiman in (Breiman, 2001), which has not previously been used in Solar Flares Prediction at least in published works - to the best of our knowledge.
- The Random Forest approach is an ensemble of "tree predictors" where we let each tree "vote" for most popular class.
- It offers a significant performance improvement to other available classification algorithms (e.g. see Fernández-Delgado et al., 2014).
- The RF approach relies on randomness and involves for variable selection the concept of purity of a split and the GINI index.

Multi-layer Perceptrons

The Multi-Layer Perceptrons were implemented using the R programming language and the ***nnet*** package (Venables & Ripley, 2002). The options used were:

1. *linout=FALSE* to ensure that sigmoid activation functions are used at the output node,
2. *entropy=TRUE* to ensure that the negative log-likelihood objective function is minimized during the training phase (and not the default Sum of Squares Error (*SSE*) criterion),
3. *size=iNode* where *iNode* for both >M1 class flares and for C-class flares was chosen with a tuning procedure.

The tuning of the MLP was mostly needed in the >M1 class flares case, which was found harder to predict than the C-class flares, but it was also performed in the C-class flares case.

Support Vector Machines
The Support Vector Machines were implemented using the R programming language and the ***e*1071** package (Meyer et al., 2015). The options used were:

1. *probability=TRUE* in order to obtain probability estimates for every element of the training set during training, and likewise, probability estimates during the prediction step for the elements of the testing set.

The tuning of the SVM was mostly needed in the >M1 class flares case, which was found harder to predict than the C-class flares, but it was also performed in the C-class flares case.

Random forests

The Random forests were implemented using the R programming language and the ***randomForest*** package (Liaw & Wiener, 2002). The options used were:

1. *importance=TRUE* to create importance information for every predictor,
2. *na.action=na.omit* to exclude records of predictors with missing values, which appeared in preliminary versions of the dataset (and actually are not present in the final version of the dataset utilized).

Random forests predicted the >M1 and C-class class flares well enough with their other default values for hyperparameters, so no additional tuning was performed for Random Forests method.

The non-machine learning (or statistical) methods considered are namely:

- linear regression
- probit regression
- logit regression

The interested reader is referred to (Greene, 2002) for a description of these estimators for the coefficients of the binary choice model in statistics/econometrics. These methods can of course be used for predictions, as in our study. The statistical estimators were also implemented in R.

Sampling:

- We use a representative sample of the 2012-2016 SHARP dataset.
- We have randomly sampled 25% of the days in the period 2012-2016, and for every given day we have computed the predictors at a cadence of 6 hours, namely at 00:00, 06:00, 12:00, 18:00 in every day.

Types of results:

1. We construct and test prediction models based on machine learning for the >M1 class flares.
2. We create and evaluate prediction models based on the statistical models for the >M1 class flares.
3. We analyse C-class flares with machine learning.
4. Finally, we predict C-class flares with the statistical estimators.

We present at first a single combination of training/testing set results, and then summary statistics for 200 replications.

Computational experiments (contd.)

What:

(a) Skill Scores profiles ACC, TSS and HSS as a function of the threshold probability,
(b) Receiver Operating Characteristic (ROC) curves for the prediction step and
(c) Reliability diagrams for the prediction step.

How:

(a) own code in R
(b) ROCR package (Sing, Sander, Beerenwinkel, & Lengauer, 2005)
(c) verification package (NCAR, 2015)

The computational system is an an Intel Core i5-4590 CPU at 3.30GHz with 16.0GB RAM and operating system Windows 7, 64−bit. All algorithms were implemented and run using the R programming language 3.2.0 (R Core Team, 2016) and the RStudio 1.0.4 IDE.

Data description

- K=8 predictors [logR, FSPI, TLMPIL, DI, Beff, IsinEn1, IsinEn2, NNC]
- These are computed using either
  - the line-of-sight Magnetograms B_{los} of SHARP data (M. Bobra et al., 2014) or
  - the radial component B_r of the SHARP data (M. Bobra et al., 2014).
- In total $2 \times 7 + 1 = 15$ predictors [1].
- The sample comprises $N = 9,324$ observations which are randomly split into $N_1 = 4,662$ observations for the training set and $N_2 = 4,662$ observations for the testing set [2].

[1] For predictor *NNC*, there is a unique algorithm version for computation

[2] Results of simulations based on the B_r or B_{los} only predictors, thus with models using only 8 predictors are available upon request

Data pre-processing

We have standardized all predictor variables to have mean equal to 0 and standard deviation equal to 1. This helps train the ML algorithms better.

$$x[i,j] = \frac{(x[i,j] - m_j)}{s_j} \quad i = 1, ..., N, \; j = 1, K$$

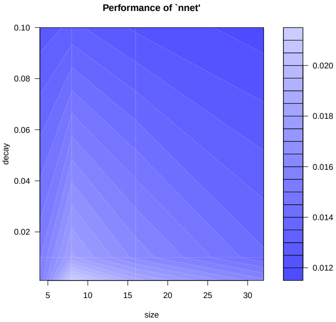
$$m_j = \mathrm{mean\ of\ j-th\ column\ of\ X,\ \ j = 1, K}$$

$$s_j = \mathrm{sd\ of\ j-th\ column\ of\ X,\ \ j = 1, K}$$

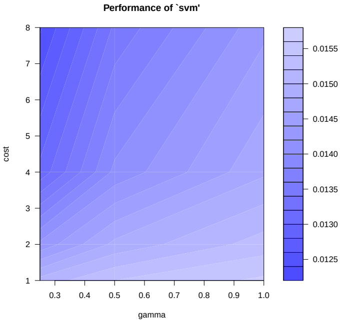
Tuning of ML algorithms (general)

The algorithms MLP, SVM and RF have the following critical hyperparameters which are tuned with a 10-fold cross-validation exploiting only the training set at one of its realizations.

- nnet.
  size, number of hidden neurons, $\in \{4, 8, 16, 32\}$
  decay, weight decay parameter, $\in \{10^{-3}, 10^{-2}, 10^{-1}\}$
- svm.
  gamma, parameter in the RBF (or Gaussian) kernel, $\in \{0.25, 0.50, 1\}$
  cost, cost of constraints violation, it is the 'C'-constant of the regularization term in the Lagrange formulation, $\in \{1, 2, 4, 8\}$
- randomForest.
  mtry, Number of variables randomly sampled as candidates at each split, $\in \{2\}$
  ntree, Number of trees to grow, $\in \{500\}$

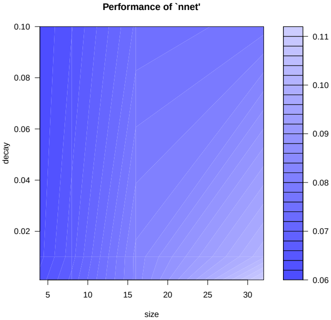


(a) MLP tuning

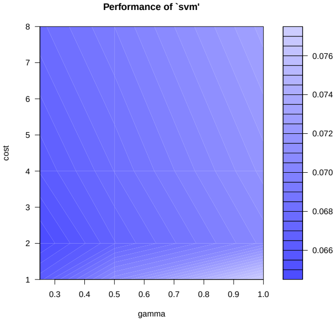


(b) SVM tuning

Figure: Algorithms tuning for MLP and SVM for >M1 class flares



(a) MLP tuning



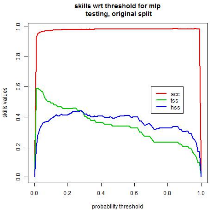
(b) SVM tuning

Figure: Algorithms tuning for MLP and SVM for C-class flares

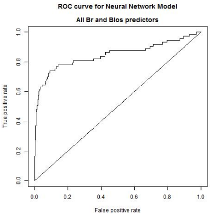
Results of tuning ML algorithms

(a) >M1 class flares Figure 3.
The selected values are size=32 and decay=0.1 for the MLP and gamma=0.25 and cost=8 for the SVM. These values are used throughout the remaining of the paper for the >M1 class flares.

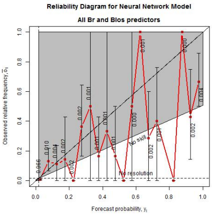
(b) C-class class flares Figure 4.
Now, the selected values are size=4 and decay=0.1 for the MLP and gamma=0.25 and cost=2 for the SVM.



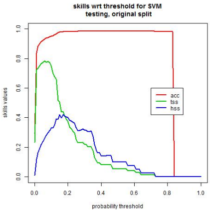
(a) MLP, SSP



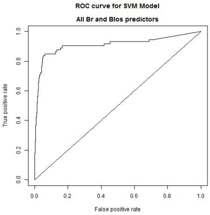
(b) MLP, ROC



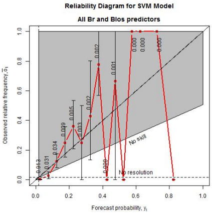
(c) MLP, RD



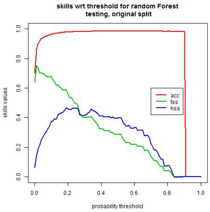
(d) SVM, SSP



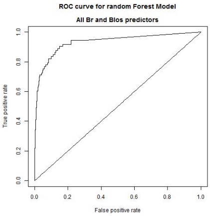
(e) SVM, ROC



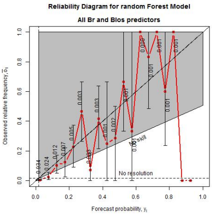
(f) SVM, RD



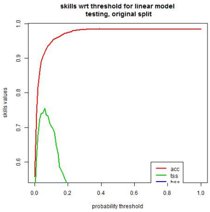
(a) RF, SSP



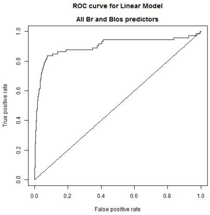
(b) RF, ROC



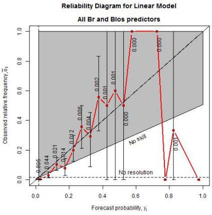
(c) RF, RD



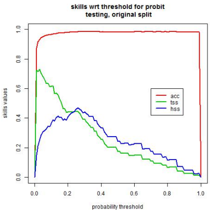
(d) LM, SSP



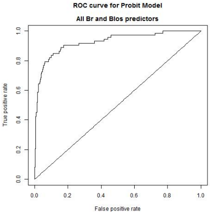
(e) LM, ROC



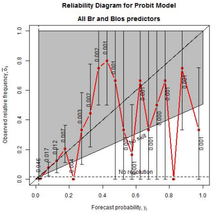
(f) LM, RD



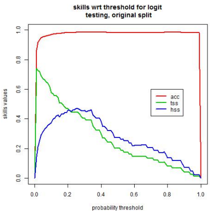
(a) PR, SSP



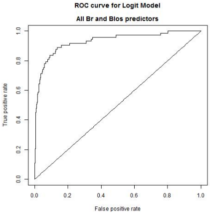
(b) PR, ROC



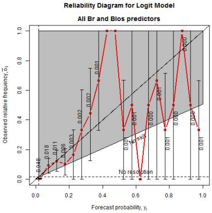
(c) PR, RD



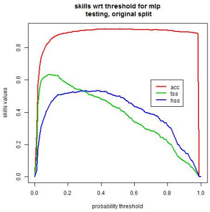
(d) LG, SSP



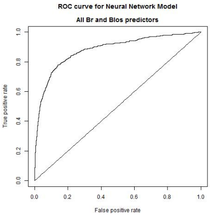
(e) LG, ROC



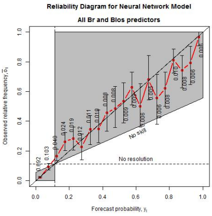
(f) LG, RD



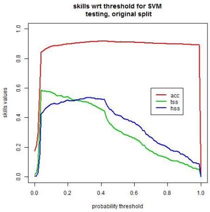
(a) MLP, SSP



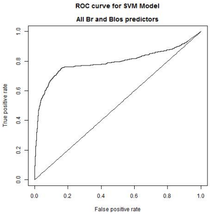
(b) MLP, ROC



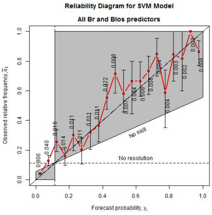
(c) MLP, RD



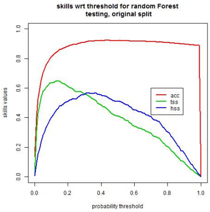
(d) SVM, SSP



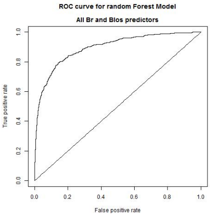
(e) SVM, ROC



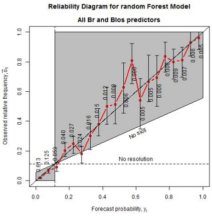
(f) SVM, RD



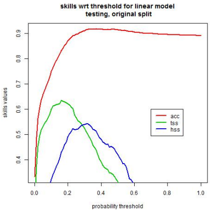
(a) RF, SSP



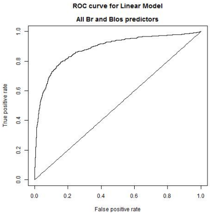
(b) RF, ROC



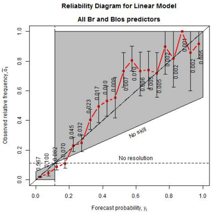
(c) RF, RD



(d) LM, SSP



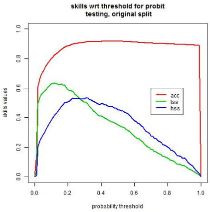
(e) LM, ROC



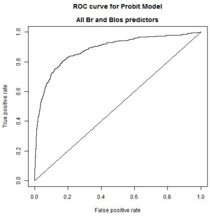
(f) LM, RD

Prediction of C class flares events using Statistics

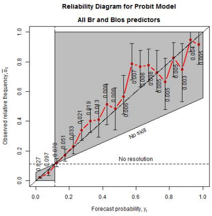
Probit regression, Logit regression



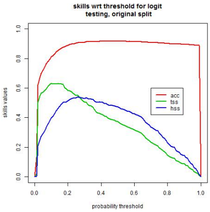
(a) PR, SSP



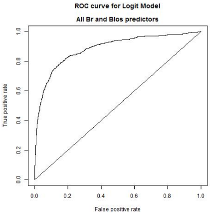
(b) PR, ROC



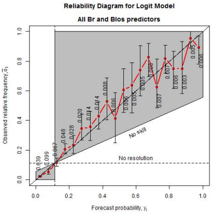
(c) PR, RD



(d) LG, SSP



(e) LG, ROC



(f) LG, RD

Monte Carlo simulation for >M1 class flares

Skill Scores Average Values

Table: Scenario 1, based on 200 datasets: Using SHYKJG1 dataset (>M1 class flares): Method 0 is Neural Network. Method 1 is Linear Regression. Method 2 is Probit Regression. Method 3 is Logit Regression. Method 4 is Random Forest Regression. Method 5 is Support Vector Regresion. Acronym A means ACC, acronym T means TSS and acronym H means HSS metric.

		0			1			2			3			4			5	
%	A	T	H	A	T	H	A	T	H	A	T	H	A	T	H	A	T	H
00	0.02	0.00	0.00	0.54	0.48	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.68	0.64	0.06	0.27	0.19	0.01
05	0.97	0.59	0.37	**0.90**	**0.75**	**0.19**	0.95	0.71	0.29	0.95	0.71	0.30	**0.94**	**0.75**	**0.28**	0.92	0.74	0.23
10	0.98	0.54	0.42	0.95	0.70	0.30	0.97	0.63	0.37	0.97	0.63	0.38	0.96	0.69	0.37	0.96	0.69	0.33
15	0.98	0.50	0.44	0.97	0.62	0.38	0.98	0.55	0.42	0.98	0.55	0.44	0.97	0.65	0.44	0.98	0.47	0.40
20	0.98	0.47	0.44	0.98	0.48	0.41	0.98	0.45	0.43	0.98	0.47	0.45	**0.98**	**0.61**	**0.50**	0.98	0.36	0.40
25	0.98	0.45	0.45	0.98	0.36	0.39	0.98	0.39	0.43	0.98	0.41	0.44	0.98	0.56	0.53	0.98	0.28	0.35
30	**0.98**	**0.42**	**0.44**	0.98	0.26	0.34	0.98	0.34	0.41	0.99	0.36	0.42	**0.99**	**0.51**	**0.54**	0.99	0.22	0.31
35	0.98	0.40	0.44	0.99	0.19	0.28	0.99	0.29	0.38	0.99	0.32	0.40	0.99	0.45	0.52	0.99	0.17	0.26
40	0.98	0.38	0.44	0.99	0.13	0.21	0.99	0.25	0.34	0.99	0.28	0.37	0.99	0.38	0.48	0.99	0.13	0.22
45	0.99	0.36	0.43	0.98	0.09	0.16	0.99	0.22	0.31	0.99	0.25	0.34	0.99	0.32	0.43	0.99	0.11	0.18
50	0.99	0.34	0.42	0.98	0.07	0.12	0.99	0.19	0.29	0.99	0.22	0.32	0.99	0.27	0.38	0.99	0.08	0.14
55	0.99	0.32	0.41	0.98	0.05	0.09	0.99	0.16	0.26	0.99	0.19	0.29	0.99	0.22	0.32	0.98	0.06	0.11
60	0.99	0.30	0.39	0.96	0.04	0.07	0.99	0.14	0.22	0.99	0.17	0.27	0.99	0.17	0.27	0.97	0.05	0.08
65	0.99	0.28	0.38	0.95	0.03	0.06	0.99	0.11	0.19	0.99	0.15	0.24	0.99	0.13	0.22	0.94	0.03	0.05
70	0.99	0.26	0.36	0.88	0.02	0.04	0.99	0.09	0.16	0.99	0.12	0.20	0.99	0.10	0.16	0.85	0.02	0.03
75	0.99	0.24	0.34	0.82	0.02	0.03	0.98	0.08	0.13	0.99	0.10	0.17	0.97	0.06	0.11	0.70	0.01	0.02
80	0.99	0.22	0.32	0.73	0.01	0.02	0.97	0.06	0.11	0.98	0.08	0.14	0.86	0.04	0.07	0.58	0.01	0.02
85	0.99	0.19	0.29	0.64	0.01	0.01	0.97	0.05	0.09	0.97	0.06	0.11	0.60	0.02	0.03	0.46	0.01	0.01
90	0.99	0.15	0.25	0.58	0.01	0.01	0.95	0.04	0.07	0.96	0.04	0.08	0.21	0.00	0.01	0.33	0.01	0.01
95	0.99	0.11	0.18	0.50	0.00	0.01	0.92	0.02	0.04	0.90	0.03	0.05	0.00	0.00	0.00	0.17	0.00	0.00
100	0.00	0.00	0.00	0.45	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table: Scenario 1, based on 200 datasets: Using SHYKJG1 dataset (>M1 class flares): Method 0 is Neural Network. Method 1 is Linear Regression. Method 2 is Probit Regression. Method 3 is Logit Regression. Method 4 is Random Forest Regression. Method 5 is Support Vector Regresion. Acronym A means ACC, acronym T means TSS and acronym H means HSS metric.

		0			1			2			3			4			5	
%	A	T	H	A	T	H	A	T	H	A	T	H	A	T	H	A	T	H
00	0.04	0.03	0.00	0.02	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.02	0.01	0.05	0.05	0.00
05	0.00	0.05	0.03	**0.01**	**0.03**	**0.02**	0.01	0.04	0.03	0.01	0.04	0.03	**0.01**	**0.05**	**0.02**	0.01	0.04	0.02
10	0.00	0.06	0.04	0.01	0.04	0.03	0.00	0.05	0.04	0.00	0.05	0.04	0.00	0.05	0.03	0.01	0.05	0.04
15	0.00	0.06	0.04	0.00	0.05	0.04	0.00	0.05	0.04	0.00	0.06	0.04	0.00	0.05	0.04	0.00	0.08	0.04
20	0.00	0.06	0.04	0.00	0.05	0.04	0.00	0.05	0.04	0.00	0.06	0.04	**0.00**	**0.05**	**0.04**	0.00	0.06	0.05
25	0.00	0.06	0.04	0.00	0.05	0.04	0.00	0.05	0.04	0.00	0.06	0.05	0.00	0.05	0.04	0.00	0.06	0.06
30	**0.00**	**0.06**	**0.04**	0.00	0.05	0.04	0.00	0.05	0.05	0.00	0.05	0.05	**0.00**	**0.06**	**0.05**	0.00	0.05	0.05
35	0.00	0.06	0.05	0.00	0.05	0.05	0.00	0.05	0.05	0.00	0.05	0.05	0.00	0.06	0.05	0.00	0.04	0.05
40	0.00	0.06	0.05	0.00	0.05	0.07	0.00	0.05	0.05	0.00	0.05	0.05	0.00	0.06	0.05	0.00	0.04	0.05
45	0.00	0.06	0.05	0.00	0.04	0.06	0.00	0.05	0.05	0.00	0.05	0.05	0.00	0.06	0.06	0.00	0.03	0.05
50	0.00	0.06	0.05	0.00	0.03	0.05	0.00	0.05	0.05	0.00	0.04	0.05	0.00	0.06	0.06	0.00	0.03	0.05
55	0.00	0.06	0.05	0.07	0.03	0.04	0.00	0.05	0.06	0.00	0.05	0.05	0.00	0.05	0.06	0.07	0.03	0.04
60	0.00	0.06	0.05	0.14	0.02	0.04	0.00	0.05	0.06	0.00	0.04	0.06	0.00	0.04	0.06	0.10	0.02	0.04
65	0.00	0.05	0.05	0.17	0.02	0.03	0.00	0.05	0.07	0.00	0.05	0.06	0.00	0.04	0.06	0.20	0.02	0.03
70	0.00	0.05	0.05	0.30	0.02	0.03	0.00	0.04	0.06	0.00	0.05	0.06	0.00	0.04	0.06	0.34	0.01	0.02
75	0.00	0.05	0.06	0.37	0.01	0.03	0.07	0.04	0.06	0.00	0.04	0.07	0.14	0.04	0.06	0.45	0.01	0.02
80	0.00	0.05	0.06	0.43	0.01	0.02	0.10	0.03	0.06	0.00	0.04	0.06	0.33	0.03	0.05	0.49	0.01	0.02
85	0.00	0.05	0.07	0.47	0.01	0.02	0.12	0.03	0.05	0.10	0.03	0.06	0.48	0.02	0.04	0.49	0.01	0.02
90	0.00	0.05	0.07	0.49	0.01	0.01	0.18	0.02	0.04	0.14	0.03	0.05	0.40	0.01	0.01	0.47	0.01	0.02
95	0.00	0.04	0.06	0.49	0.01	0.01	0.25	0.02	0.03	0.28	0.02	0.03	0.07	0.00	0.00	0.37	0.01	0.01
100	0.00	0.00	0.00	0.49	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table: Scenario 1, based on 200 datasets: Using SHYKJG1 dataset (C-class flares): Method 0 is Neural Network. Method 1 is Linear Regression. Method 2 is Probit Regression. Method 3 is Logit Regression. Method 4 is Random Forest Regression. Method 5 is Support Vector Regresion. Acronym A means ACC, acronym T means TSS and acronym H means HSS metric.

		0			1			2			3			4			5	
%	A	T	H	A	T	H	A	T	H	A	T	H	A	T	H	A	T	H
00	0.00	0.00	0.00	0.31	0.20	0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.14	0.03	0.01	0.17	0.02	0.01
05	0.76	0.60	0.33	0.65	0.53	0.23	0.70	0.57	0.27	0.71	0.57	0.28	0.69	0.56	0.26	0.85	0.60	0.44
10	0.83	0.63	0.42	0.74	0.60	0.31	0.79	0.62	0.37	0.80	0.63	0.38	0.80	0.63	0.38	0.88	0.58	0.48
15	0.87	0.61	0.47	0.81	0.63	0.40	0.85	0.64	0.45	**0.85**	**0.64**	**0.46**	**0.85**	**0.64**	**0.46**	0.89	0.57	0.50
20	0.89	0.59	0.50	0.86	0.63	0.48	0.88	0.61	0.51	0.89	0.61	0.51	**0.88**	**0.63**	**0.51**	0.90	0.55	0.51
25	0.90	0.56	0.52	0.90	0.59	0.52	0.90	0.57	0.53	0.90	0.57	0.53	0.90	0.60	0.54	0.90	0.53	0.52
30	0.91	0.53	0.53	0.91	0.53	0.54	0.91	0.53	0.54	0.91	0.53	0.54	0.91	0.57	0.55	0.91	0.51	0.53
35	**0.91**	**0.51**	**0.53**	0.92	0.47	0.52	0.92	0.48	0.53	0.92	0.49	0.53	**0.92**	**0.54**	**0.56**	0.91	0.49	0.53
40	0.92	0.48	0.53	0.92	0.41	0.49	0.92	0.43	0.50	0.92	0.45	0.51	0.92	0.51	0.56	0.92	0.47	0.52
45	0.92	0.45	0.52	0.92	0.36	0.46	0.92	0.39	0.48	0.92	0.41	0.49	0.92	0.47	0.54	0.92	0.40	0.49
50	0.92	0.43	0.51	0.92	0.32	0.43	0.92	0.36	0.46	0.92	0.38	0.48	0.92	0.43	0.52	0.92	0.33	0.44
55	0.92	0.40	0.49	0.91	0.27	0.38	0.92	0.33	0.44	0.92	0.35	0.46	0.92	0.40	0.50	0.91	0.29	0.40
60	0.92	0.37	0.47	0.91	0.22	0.32	0.92	0.30	0.41	0.92	0.32	0.43	0.92	0.36	0.47	0.91	0.25	0.35
65	0.92	0.34	0.45	0.91	0.18	0.27	0.91	0.26	0.37	0.92	0.29	0.40	0.92	0.32	0.43	0.91	0.22	0.32
70	0.92	0.31	0.42	0.90	0.14	0.23	0.91	0.22	0.32	0.91	0.25	0.36	0.92	0.28	0.39	0.91	0.18	0.28
75	0.91	0.28	0.38	0.90	0.11	0.18	0.91	0.18	0.27	0.91	0.21	0.31	0.91	0.24	0.35	0.90	0.15	0.24
80	0.91	0.23	0.33	0.90	0.09	0.15	0.90	0.14	0.23	0.91	0.17	0.27	0.91	0.19	0.29	0.90	0.13	0.20
85	0.87	0.16	0.25	0.90	0.07	0.12	0.90	0.11	0.18	0.90	0.14	0.21	0.90	0.14	0.23	0.90	0.10	0.16
90	0.75	0.07	0.12	0.90	0.06	0.10	0.90	0.08	0.14	0.90	0.10	0.16	0.90	0.09	0.15	0.90	0.08	0.13
95	0.50	0.02	0.03	0.89	0.05	0.08	0.90	0.05	0.09	0.90	0.06	0.10	0.89	0.04	0.07	0.90	0.05	0.09
100	0.00	0.00	0.00	0.89	0.04	0.07	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Monte Carlo simulation for C class flares

Standard deviations of the Skills Scores Values

Table: Scenario 1, based on 200 datasets: Using SHYKJG dataset (C-class flares): Method 0 is Neural Network. Method 1 is Linear Regression. Method 2 is Probit Regression. Method 3 is Logit Regression. Method 4 is Random Forest Regression. Method 5 is Support Vector Regresion. Acronym A means ACC, acronym T means TSS and acronym H means HSS metric.

		0			1			2			3			4			5	
%	A	T	H	A	T	H	A	T	H	A	T	H	A	T	H	A	T	H
00	0.00	0.00	0.00	0.04	0.04	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.01	0.01	0.00
05	0.01	0.01	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.02	0.01
10	0.01	0.01	0.02	0.01	0.01	0.02	0.01	0.01	0.02	0.01	0.01	0.02	0.01	0.01	0.01	0.00	0.02	0.02
15	0.01	0.02	0.02	0.01	0.01	0.02	0.01	0.01	0.02	**0.01**	**0.01**	**0.02**	**0.01**	**0.02**	**0.01**	0.00	0.02	0.02
20	0.01	0.02	0.02	0.01	0.02	0.02	0.01	0.02	0.01	0.01	0.02	0.01	**0.00**	**0.02**	**0.02**	0.00	0.02	0.02
25	0.00	0.02	0.01	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02
30	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02
35	**0.00**	**0.02**	**0.02**	0.00	0.02	0.01	0.00	0.02	0.02	0.00	0.02	0.01	**0.00**	**0.02**	**0.02**	0.00	0.02	0.02
40	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02
45	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.03	0.02
50	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02
55	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02
60	0.00	0.03	0.02	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02
65	0.00	0.03	0.02	0.00	0.02	0.02	0.00	0.03	0.03	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02
70	0.00	0.03	0.03	0.00	0.02	0.03	0.00	0.03	0.03	0.00	0.02	0.03	0.00	0.02	0.02	0.00	0.02	0.02
75	0.00	0.03	0.03	0.00	0.02	0.03	0.00	0.03	0.03	0.00	0.02	0.03	0.00	0.02	0.03	0.00	0.02	0.02
80	0.00	0.03	0.04	0.00	0.02	0.02	0.00	0.03	0.04	0.00	0.02	0.03	0.00	0.02	0.03	0.00	0.02	0.02
85	0.18	0.05	0.07	0.00	0.01	0.02	0.00	0.02	0.03	0.00	0.02	0.03	0.00	0.02	0.03	0.00	0.01	0.02
90	0.34	0.05	0.08	0.00	0.01	0.02	0.00	0.02	0.03	0.00	0.02	0.03	0.00	0.02	0.03	0.00	0.01	0.02
95	0.44	0.03	0.04	0.00	0.01	0.02	0.00	0.02	0.03	0.00	0.02	0.02	0.00	0.01	0.02	0.00	0.01	0.02
100	0.00	0.00	0.00	0.00	0.01	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

- We observe from Table 3 that regarding prediction of >M1 class flares, the maximum value of HSS observed is with method Random Forest and for probability threshold 30% with value HSS=0.54. The corresponding metrics values are ACC=0.99$\pm$0.00, TSS=0.51$\pm$0.06 and HSS=0.54$\pm$0.05 at threshold 30% for the Random Forest.
- Considering the threshold where the maximum TSS is observed, we get the optimal results for method Random Forest and threshold 5% with values ACC=0.94$\pm$0.01, TSS=0.75$\pm$0.05 and HSS=0.28$\pm$0.02.

- For the range of thresholds 5% to 30% the method Random Forest yields increasing values of HSS and decreasing values of TSS. So, it is not clear which the optimal threshold value would be for a Decision Maker (DM) who is simultaneously interested in optimizing both TSS and HSS.
- For example, an appealing forecasting model is Random Forest with threshold 20% and metrics ACC=0.98 ± 0.00, TSS=0.61 ± 0.05 and HSS=0.50 ± 0.04 in Table 3.

Discussion on Monte Carlo for C class flares

- We observe from Table 5 that regarding prediction of C-class flares, the maximum value of HSS observed is with method Random Forest and for probability threshold 35% with value HSS=0.56. The corresponding metrics values are ACC=0.92±0.00, TSS=0.54±0.02 and HSS=0.56±0.02 at threshold 35% for the Random Forest.
- Considering again the threshold where the maximum TSS is observed, we get the optimal results for method Random Forest and threshold 15% with values ACC=0.85±0.01, TSS=0.64±0.02 and HSS=0.46±0.01.

- For the range of thresholds 15% to 35% the method Random Forest yields increasing values of HSS and decreasing values of TSS.
- For example, an appealing forecasting model is Random Forest with threshold 20% and metrics ACC=0.88±0.00, TSS=0.63±0.02 and HSS=0.51±0.02 in Table 5.
- These results are competitive with recent findings in the literature for C-class flares predictability in the range of 0.50–0.55 for TSS and 0.40–0.45 for HSS (see Al-Ghraibah, A. et al., 2015; Boucheron et al., 2015).

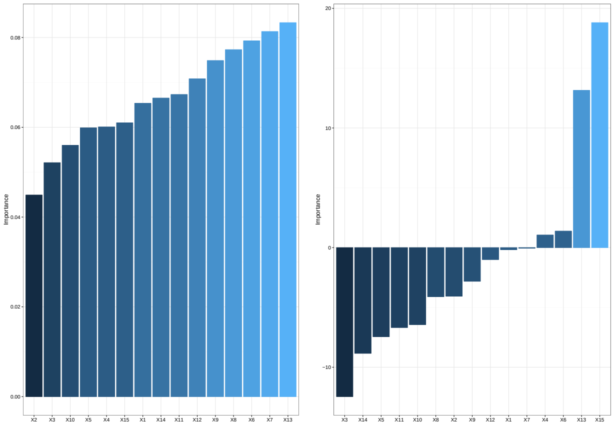
Nomenclature of predictors in the following slides

Param	Meaning	Number
logRBl	Schrijver's R value based on B_{los}	x1
logRBr	Schrijver's R value based on B_r	x2
FSPIBl	Fourier spectral power index based on B_{los}	x3
FSPIBr	Fourier spectral power index based on B_r	x4
TLMPILBl	Magnetic polarity inversion line based on B_{los}	x5
TLMPILBr	Magnetic polarity inversion line based on B_r	x6
DIBl	Decay index based on B_{los}	x7
DIBr	Decay index based on B_r	x8
BeffBl	B effective value based on B_{los}	x9
BeffBr	B effective value based on B_r	x10
IsinEn1Bl	Ising energy (partitioned) based on B_{los}	x11
IsinEn1Br	Ising energy (partitioned) based on B_r	x12
IsinEn2Bl	Ising energy (Ahmed et al.) based on B_{los}	x13
IsinEn2Br	Ising energy (Ahmed et al.) based on B_r	x14
NNC	Non-neutralized Current	x15

Table: Names of input variables in SHYKJG1 dataset, (N=9,324, K=15)

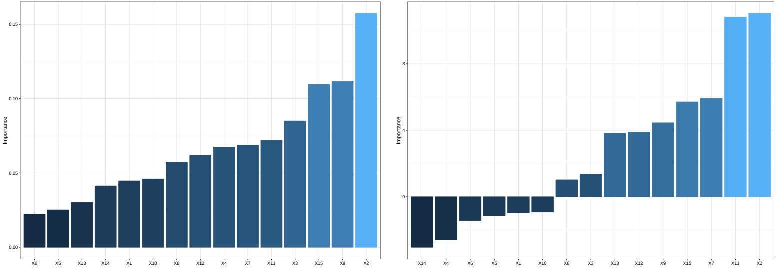
Relative Importance of predictors

Ranking predictors within the Neural Networks paradigm using (Beck, 2015)



(a) Garson, MX class (b) Olden, MX class

Figure: Ranking predictors within the MLP paradigm using algorithms (Garson, 1991) and (Olden et al., 2004) for MX class flares training sets.

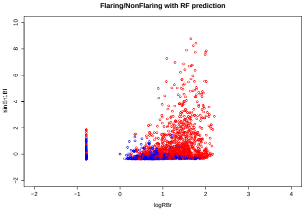


(a) Garson, C class (b) Olden, C class

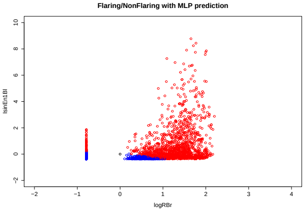
Figure: Ranking predictors within the MLP paradigm using algorithms (Garson, 1991) and (Olden et al., 2004) for C class flares training sets.

Keeping the 2 most important predictors (Olden)

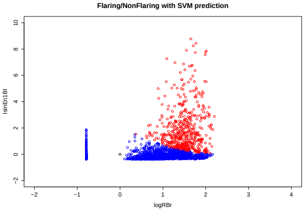
Building prediction models and visualing results (5% probability threshold)



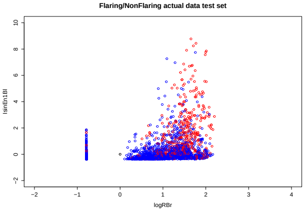
(a) RF, C class, 5%



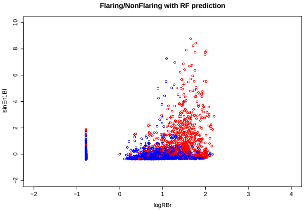
(b) MLP, C class, 5%



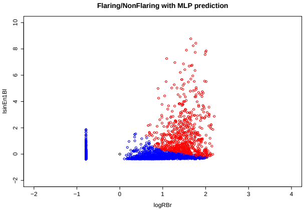
(c) SVM, C class, 5%



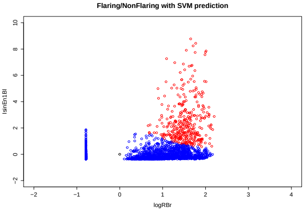
(d) Actual data, C class



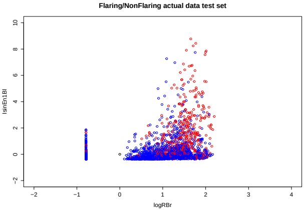
(a) RF, C class, 20%



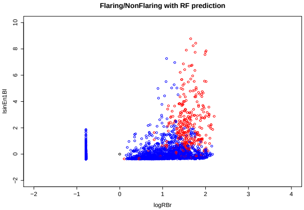
(b) MLP, C class, 20%



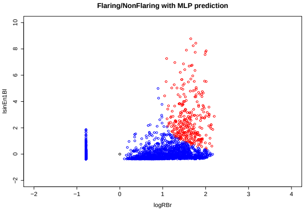
(c) SVM, C class, 20%



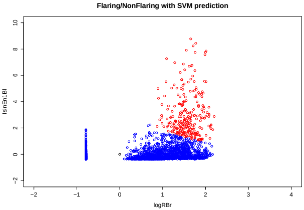
(d) Actual data, C class



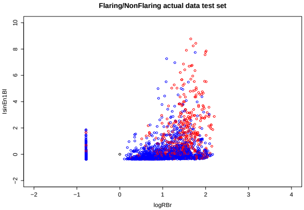
(a) RF, C class, 50%



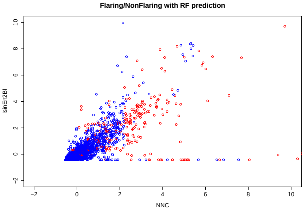
(b) MLP, C class, 50%



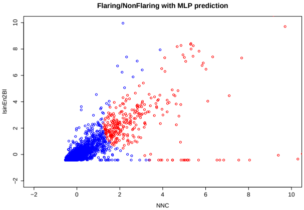
(c) SVM, C class, 50%



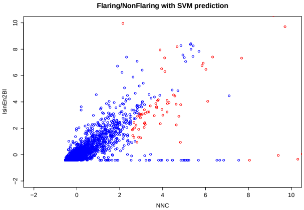
(d) Actual data, C class



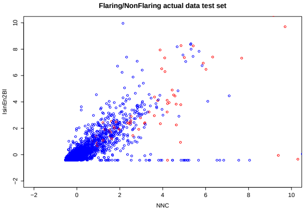
(a) RF, MX class, 5%



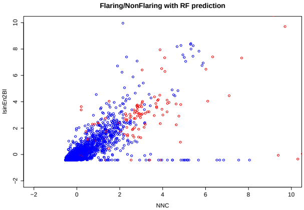
(b) MLP, MX class, 5%



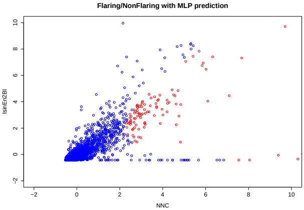
(c) SVM, MX class, 5%



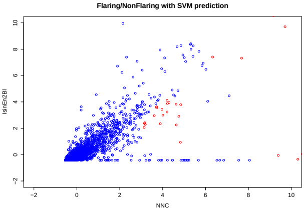
(d) Actual data, MX class



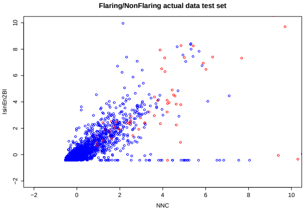
(a) RF, MX class, 20%



(b) MLP, MX class, 20%



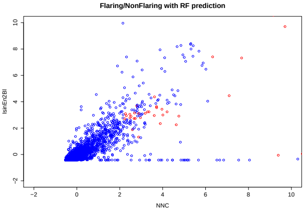
(c) SVM, MX class, 20%



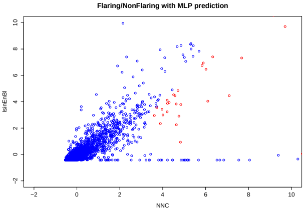
(d) Actual data, MX class

Keeping the 2 most important predictors (Olden)

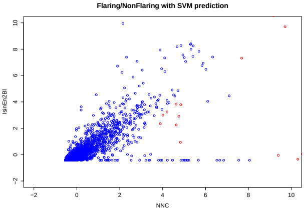
Building prediction models and visualing results (50% probability threshold)



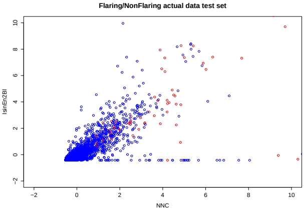
(a) RF, MX class, 50%



(b) MLP, MX class, 50%



(c) SVM, MX class, 50%



(d) Actual data, MX class

Papers under preparation

- Solar flares forecasting using magnetogram-based predictors and machine learning $\leftarrow$ close to submission
- $\star$Recurrent neural networks training using evolutionary algorithms: The case of solar flares forecasting$\star$ $\leftarrow$ algorithm development
- $\star$Multiobjective two-class classification using evolutionary algorithms: The case of solar flares forecasting$\star$ $\leftarrow$ algorithm development

Discussion

- We presented a new approach for the effective prediction of >M1 and C-class solar flares.
- Machine learning methods such as MLP, SVM and random forests were used
- The predictor variables were based on the SDO/HMI/SHARP NASA data product available since 2012.
- The sample was representative of the solar activity during (2012-2016) since 25% of the days were included in the sample
- The intra-day cadence used was 6h
- Forecast horizon was 24h

Discussion (2)

- The results show that the random forest methodology seems to be overall the prediction method of choice.
- $>$M1 class flares are more difficult to predict since the relative frequency of $>$M1 class flares events in the original sample (N=9324, K=15) was only 1.6% (146 events)
- C-class flares are easier to predict since the relative frequency of C class flares events in the original sample is 11.0% (1024 events)
- RandomForest is capable of predicting effectively $>$M1 class flares and especially C-class flares, a fact which is mirrored on the reliability diagrams obtained and the values of the skill scores

Discussion (3)

- Regarding >M1 class flares, SVM and MLP needed additional tuning
- Statistical methods (linear regression, probit, logit) produced acceptable forecasting performance
- Regarding the C-class flares, performance is very good for almost all forecasting methods
- The winner in the C-class flares is again the RandomForest
- A Monte Carlo showed that results are robust while using different random seeds to define training/testing sets
- The standard deviations of the skill scores are low in the Monte Carlo.

Summary

- RandomForest is better than MLP and SVM, and this is promising for future research since RandomForest is used for Solar Flares prediction for first time in the present work
- Also, it is inline with a body of work reporting excellent performance of Random Forest in several classification benchmarks (e.g. see Fernández-Delgado et al., 2014)
- Importantly, the predictor Non-Neutralized Current proposed in this study is found to be the most statistically significant predictor together with Schrijver's R value and Ising Energy

Future research

- We plan to enlarge the sample on which our analysis is done, by reducing the cadence from 6h to 3h or even 1h (the limit is 12min)
- We plan to investigate the time-series aspects of our data, using recurrent neural networks possibly trained with evolutionary algorithms
- Finally, the present work is intended to be integrated in the FLARECAST online forecasting tool

Thank you

This research has been financed by the European Union's Horizon2020 research and innovation programme under grant agreement No.640216 for the "Flare Likelihood And Region Eruption foreCASTing" (FLARECAST) project.

Ahmed, O. W., Qahwaji, R., Colak, T., Higgins, P. A., Gallagher, P. T., & Bloomfield, D. S. (2013). Solar flare prediction using advanced feature extraction, machine learning, and feature selection. *Solar Physics*, *283*(1), 157–175.

Al-Ghraibah, A., Boucheron, L. E., & McAteer, R. T. J. (2015). An automated classification approach to ranking photospheric proxies of magnetic energy build-up. *Astronomy and Astrophysics*, *579*, A64.

Beck, M. (2015). NeuralNetTools: Visualization and Analysis Tools for Neural Networks [Computer software manual]. Retrieved from http://CRAN.R-project.org/package=NeuralNetTools (R package version 1.4.0)

Bessonov, K. (2016). *GBIO002-1 Genetics and Bioinformatics 2015-16 (educational material).* Retrieved from http://www.montefiore.ulg.ac.be/~kbessonov/ ([Online; last accessed 10-October-2016])

Bobra, M., Sun, X., Hoeksema, J., Turmon, M., Liu, Y., Hayashi, K., . . . Leka, K. (2014). The Helioseismic and Magnetic

Imager (HMI) Vector Magnetic Field Pipeline: SHARPs - Space-Weather HMI Active Region Patches. *Solar Physics*, *289*(9), 3549-3578.

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.

Boucheron, L. E., Al-Ghraibah, A., & McAteer, R. T. J. (2015). Prediction of solar flare size and time-to-flare using support vector machine regression. *The Astrophysical Journal*, *812*(1), 51.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5-32.

Chang, C.-C., & Lin, C.-J. (2011). LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, *2*, 27:1–27:27. (Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm)

Colak, T., & Qahwaji, R. (2009). Automated solar activity prediction: A hybrid computer platform using machine learning and solar imaging for automated prediction of solar flares. *Space Weather*, *7*(6), S06001.

Fernández-Delgado, M., Cernadas, E., Barro, S., & Amorim, D. (2014). Do we need hundreds of classifiers to solve real world classification problems? *Journal of Machine Learning Research*, *15*, 3133-3181.

Garson, G. D. (1991, April). Interpreting neural-network connection weights. *AI Expert*, *6*(4), 46–51.

Greene, W. H. (2002). *Econometric analysis* (5th ed.). Prentice Hall.

Hanssen, A., & Kuipers, W. (1965). *On the relationship between the frequency of rain and various meteorological parameters:(with reference to the problem of objective forecasting)*. Staatsdrukkerij-en Uitgeverijbedrijf.

Heidke, P. (1926). Berechnung des erfolges und der güte der windstärkevorhersagen im sturmwarnungsdienst. *Geografiska Annaler*, *8*, 301-349.

Li, R., Cui, Y., He, H., & Wang, H. (2008). Application of support vector machine combined with k-nearest neighbors in solar flare and solar proton events forecasting. *Advances in Space Research*, *42*(9), 1469–1474.

Liaw, A., & Wiener, M. (2002). Classification and regression by randomforest. *R News*, *2*(3), 18-22.

MacKay, D. J. (2003). *Information theory, inference, and learning algorithms*. Cambridge University Press.

Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., & Leisch, F. (2015). e1071: Misc functions of the department of statistics, probability theory group (formerly: E1071), tu wien [Computer software manual]. (R package version 1.6-7)

Meyer, D., Leisch, F., & Hornik, K. (2003). The support vector machine under test. *Neurocomputing*, *55*(1), 169–186.

NCAR. (2015). verification: Weather forecast verification utilities [Computer software manual]. (NCAR - Research Applications Laboratory, R package version 1.42)

Olden, J. D., Joy, M. K., & Death, R. G. (2004). An accurate comparison of methods for quantifying variable importance in artificial neural networks using simulated data. *Ecological Modelling*, *178*(3–4), 389 - 397.

Pesnell, W. D., Thompson, B. J., & Chamberlin, P. C. (2012, January). The Solar Dynamics Observatory (SDO). *Solar Physics*, *275*, 3-15. doi: 10.1007/s11207-011-9841-3

R Core Team. (2016). R: A language and environment for statistical computing [Computer software manual]. Vienna, Austria.

Scherrer, P. H., Schou, J., Bush, R. I., Kosovichev, A. G., Bogart, R. S., Hoeksema, J. T., . . . Tomczyk, S. (2012, January). The Helioseismic and Magnetic Imager (HMI) Investigation for the Solar Dynamics Observatory (SDO). *Solar Physics*, *275*, 207-227. doi: 10.1007/s11207-011-9834-2

Sing, T., Sander, O., Beerenwinkel, N., & Lengauer, T. (2005). Rocr: visualizing classifier performance in r. *Bioinformatics*, *21*(20), 7881.

Song, H., Tan, C., Jing, J., Wang, H., Yurchyshyn, V., & Abramenko, V. (2009). Statistical assessment of photospheric magnetic features in imminent solar flare predictions. *Solar Physics*, *254*(1), 101–125.

Vapnik, V. (1998). *Statistical learning theory*. Wiley.

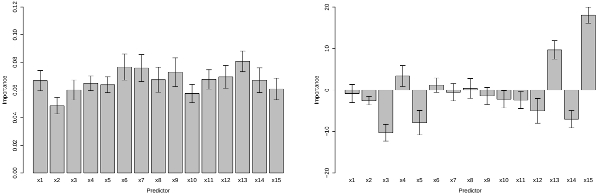
Venables, W. N., & Ripley, B. D. (2002). *Modern applied statistics with s* (Fourth ed.). New York: Springer.

Wang, H., Cui, Y., Li, R., Zhang, L., & Han, H. (2008). Solar flare forecasting model supported with artificial neural network techniques. *Advances in Space Research*, *42*(9), 1464 - 1468.

Yu, D., Huang, X., Wang, H., & Cui, Y. (2009). Short-term solar flare prediction using a sequential supervised learning method. *Solar Physics*, *255*(1), 91–105.

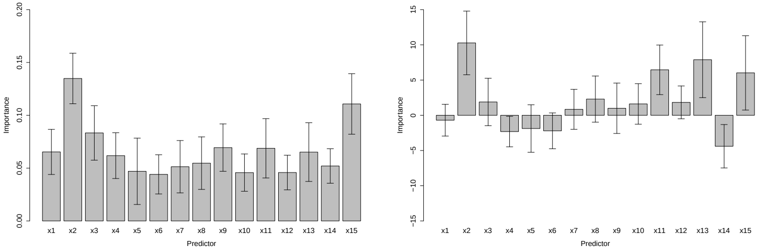
Yuan, Y., Shih, F. Y., Jing, J., & Wang, H.-M. (2010). Automated flare forecasting using a statistical learning technique. *Research in Astronomy and Astrophysics*, *10*(8), 785.

BACKUP SLIDES



(a) Garson, MX class (b) Olden, MX class

Figure: Ranking predictors within the MLP paradigm using algorithms (Garson, 1991) and (Olden et al., 2004) for MX class flares training sets **using 30 runs**.



(a) Garson, C class (b) Olden, C class

Figure: Ranking predictors within the MLP paradigm using algorithms (Garson, 1991) and (Olden et al., 2004) for C class flares training sets **using 30 runs**.

Nomenclature

Param	Meaning	Number
logRBl	Schrijver's R value based on B_{los}	x1
logRBr	Schrijver's R value based on B_r	x2
FSPIBl	Fourier spectral power index based on B_{los}	x3
FSPIBr	Fourier spectral power index based on B_r	x4
TLMPILBl	Magnetic polarity inversion line based on B_{los}	x5
TLMPILBr	Magnetic polarity inversion line based on B_r	x6
DIBl	Decay index based on B_{los}	x7
DIBr	Decay index based on B_r	x8
BeffBl	B effective value based on B_{los}	x9
BeffBr	B effective value based on B_r	x10
IsinEn1Bl	Ising energy (partitioned) based on B_{los}	x11
IsinEn1Br	Ising energy (partitioned) based on B_r	x12
IsinEn2Bl	Ising energy (Ahmed et al.) based on B_{los}	x13
IsinEn2Br	Ising energy (Ahmed et al.) based on B_r	x14
NNC	Non-neutralized Current	x15

Table: Names of input variables in SHYKJG1 dataset, (N=9,324, K=15)

MX class flares Estimation results for Linear regression, Probit and Logit (replication 200^{th})

Table: Linear model original split training results for >M1 class flares.

Param	Estimate	Std. Error	t value	$Pr(>\|t\|)$
INTCPT	0.0157	0.0016	10.0565	0.0000
logRBl	−0.0159	0.0031	−5.0970	0.0000
logRBr	−0.0108	0.0035	−3.0672	0.0022
FSPIBl	0.0016	0.0040	0.3915	0.6954
FSPIBr	−0.0009	0.0033	−0.2744	0.7838
TLMPILBl	−0.0021	0.0028	−0.7304	0.4652
TLMPILBr	0.0059	0.0025	2.3539	0.0186
DIBl	0.0031	0.0027	1.1358	0.2561
DIBr	0.0161	0.0024	6.8246	0.0000
BeffBl	0.0150	0.0032	4.7269	0.0000
BeffBr	0.0044	0.0026	1.7126	0.0869
IsinEn1Bl	0.0201	0.0035	5.7768	0.0000
IsinEn1Br	−0.0462	0.0042	−11.0557	0.0000
IsinEn2Bl	0.0088	0.0030	2.9003	0.0037
IsinEn2Br	0.0016	0.0024	0.6722	0.5015
NNC	0.0665	0.0055	12.0702	0.0000

MX class flares Estimation results for Linear regression, Probit and Logit (replication 200^{th})

Table: Probit model original split training results for >M1 class flares.

Param	Estimate	Std. Error	z value	$Pr(>\|z\|)$
INTCPT	−2.9811	0.1692	−17.6236	0.0000
logRBl	0.2804	0.2051	1.3673	0.1715
logRBr	−0.0789	0.1885	−0.4187	0.6754
FSPIBl	0.1677	0.1597	1.0500	0.2937
FSPIBr	−0.1920	0.1648	−1.1646	0.2442
TLMPILBl	−0.2194	0.1110	−1.9764	0.0481
TLMPILBr	0.1287	0.0643	2.0013	0.0454
DIBl	0.0878	0.0600	1.4623	0.1437
DIBr	0.1241	0.0493	2.5191	0.0118
BeffBl	0.1596	0.0682	2.3408	0.0192
BeffBr	−0.0380	0.0840	−0.4529	0.6506
IsinEn1Bl	0.1203	0.0727	1.6554	0.0978
IsinEn1Br	−0.2441	0.0926	−2.6357	0.0084
IsinEn2Bl	0.0590	0.0524	1.1243	0.2609
IsinEn2Br	0.0898	0.0625	1.4375	0.1506
NNC	0.3712	0.1012	3.6693	0.0002

MX class flares Estimation results for Linear regression, Probit and Logit (replication 200^{th})

Table: Logit model original split training results for >M1 class flares.

Param	Estimate	Std. Error	z value	$Pr(>\|z\|)$
INTCPT	−6.8476	0.6357	−10.7717	0.0000
logRBl	1.1186	0.6536	1.7113	0.0870
logRBr	0.0754	0.5522	0.1365	0.8914
FSPIBl	0.3804	0.3577	1.0637	0.2875
FSPIBr	−0.5479	0.3794	−1.4440	0.1487
TLMPILBl	−0.4881	0.2338	−2.0879	0.0368
TLMPILBr	0.2659	0.1401	1.8979	0.0577
DIBl	0.1803	0.1143	1.5773	0.1147
DIBr	0.2289	0.0947	2.4174	0.0156
BeffBl	0.3083	0.1335	2.3100	0.0209
BeffBr	−0.0824	0.1652	−0.4988	0.6179
IsinEn1Bl	0.2887	0.1448	1.9930	0.0463
IsinEn1Br	−0.4526	0.1843	−2.4551	0.0141
IsinEn2Bl	0.0828	0.0984	0.8419	0.3999
IsinEn2Br	0.2203	0.1241	1.7754	0.0758
NNC	0.5896	0.1937	3.0444	0.0023

C class flares Estimation results for Linear regression, Probit and Logit (replication 200^{th})

Table: Linear model original split training results for C-class flares.

Param	Estimate	Std. Error	t value	$\Pr(>\|t\|)$
INTCPT	0.1065	0.0037	28.6525	0.0000
logRBl	0.0133	0.0075	1.7896	0.0736
logRBr	0.0460	0.0084	5.4702	0.0000
FSPIBl	0.0848	0.0095	8.9220	0.0000
FSPIBr	−0.0523	0.0079	−6.6188	0.0000
TLMPILBl	−0.0188	0.0067	−2.7876	0.0053
TLMPILBr	−0.0049	0.0060	−0.8193	0.4127
DIBl	0.0145	0.0064	2.2685	0.0233
DIBr	0.0213	0.0056	3.7951	0.0001
BeffBl	0.0235	0.0076	3.1072	0.0019
BeffBr	0.0052	0.0062	0.8401	0.4009
IsinEn1Bl	−0.0029	0.0083	−0.3473	0.7284
IsinEn1Br	−0.0141	0.0100	−1.4173	0.1565
IsinEn2Bl	0.0345	0.0073	4.7540	0.0000
IsinEn2Br	0.0029	0.0057	0.5050	0.6136
NNC	0.1040	0.0131	7.9241	0.0000

C class flares Estimation results for Linear regression, Probit and Logit (replication 200^{th})

Table: Probit model original split training results for C-class flares.

Param	Estimate	Std. Error	z value	$Pr(>\|z\|)$
INTCPT	−1.6368	0.0395	−41.3883	0.0000
logRBl	0.1863	0.0630	2.9562	0.0031
logRBr	0.3561	0.0695	5.1270	0.0000
FSPIBl	0.4847	0.0797	6.0812	0.0000
FSPIBr	−0.3242	0.0684	−4.7394	0.0000
TLMPILBl	−0.1273	0.0583	−2.1834	0.0290
TLMPILBr	−0.0157	0.0366	−0.4288	0.6681
DIBl	0.0687	0.0413	1.6639	0.0961
DIBr	0.0804	0.0346	2.3246	0.0201
BeffBl	0.0803	0.0469	1.7117	0.0870
BeffBr	0.0562	0.0367	1.5321	0.1255
IsinEn1Bl	−0.0827	0.0525	−1.5753	0.1152
IsinEn1Br	−0.0409	0.0599	−0.6835	0.4943
IsinEn2Bl	0.0898	0.0474	1.8940	0.0582
IsinEn2Br	0.0216	0.0360	0.6007	0.5480
NNC	0.3885	0.0859	4.5247	0.0000

C class flares Estimation results for Linear regression, Probit and Logit (replication 200^{th})

Table: Logit model original split training results for C-class flares.

Param	Estimate	Std. Error	z value	$Pr(>\|z\|)$
INTCPT	−3.0961	0.0967	−32.0302	0.0000
logRBl	0.4191	0.1363	3.0751	0.0021
logRBr	0.7879	0.1502	5.2460	0.0000
FSPIBl	0.9149	0.1564	5.8481	0.0000
FSPIBr	−0.6200	0.1365	−4.5413	0.0000
TLMPILBl	−0.1990	0.1053	−1.8902	0.0587
TLMPILBr	0.0039	0.0634	0.0613	0.9511
DIBl	0.0938	0.0733	1.2788	0.2010
DIBr	0.1421	0.0606	2.3472	0.0189
BeffBl	0.0824	0.0824	1.0001	0.3173
BeffBr	0.0794	0.0647	1.2271	0.2198
IsinEn1Bl	−0.0952	0.0940	−1.0118	0.3116
IsinEn1Br	−0.1255	0.1072	−1.1706	0.2418
IsinEn2Bl	0.1596	0.0874	1.8263	0.0678
IsinEn2Br	0.0397	0.0650	0.6114	0.5409
NNC	0.7083	0.1578	4.4881	0.0000

NEW DATA: Monte Carlo simulation for >M1 class flares

Skill Scores Average Values

Table: Scenario 1, based on 200 datasets: Using SHYKJG1 dataset (>M1 class flares): Method 0 is Neural Network. Method 1 is Linear Regression. Method 2 is Probit Regression. Method 3 is Logit Regression. Method 4 is Random Forest Regression. Method 5 is Support Vector Regresion. Acronym A means ACC, acronym T means TSS and acronym H means HSS metric.

	0			1			2			3			4			5		
%	A	T	H	A	T	H	A	T	H	A	T	H	A	T	H	A	T	H
00	0.35	0.34	0.02	0.56	0.51	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.81	0.79	0.11	0.44	0.36	0.02
05	0.97	0.65	0.41	**0.91**	**0.81**	**0.22**	0.95	0.79	0.31	0.95	0.78	0.33	**0.94**	**0.80**	**0.30**	0.95	0.76	0.31
10	0.98	0.59	0.45	0.95	0.78	0.33	0.97	0.72	0.39	0.97	0.71	0.41	0.96	0.74	0.38	0.97	0.71	0.40
15	0.98	0.56	0.47	0.97	0.70	0.41	0.97	0.65	0.44	0.98	0.64	0.45	0.97	0.70	0.45	0.98	0.63	0.45
20	0.98	0.53	0.48	0.98	0.58	0.45	0.98	0.59	0.47	0.98	0.57	0.48	**0.98**	**0.66**	**0.51**	0.98	0.48	0.46
25	0.98	0.50	0.48	0.98	0.44	0.44	0.98	0.51	0.48	0.98	0.51	0.49	0.98	0.62	0.55	0.98	0.42	0.45
30	**0.98**	**0.48**	**0.49**	0.99	0.34	0.40	0.99	0.44	0.47	0.99	0.45	0.48	**0.99**	**0.56**	**0.57**	0.99	0.37	0.44
35	0.99	0.45	0.48	0.99	0.26	0.35	0.99	0.38	0.45	0.99	0.40	0.46	0.99	0.51	0.56	0.99	0.32	0.41
40	0.99	0.43	0.47	0.99	0.20	0.29	0.99	0.33	0.42	0.99	0.35	0.44	0.99	0.44	0.53	0.99	0.27	0.37
45	0.99	0.41	0.47	0.99	0.15	0.24	0.99	0.29	0.39	0.99	0.32	0.41	0.99	0.38	0.48	0.99	0.22	0.32
50	0.99	0.39	0.46	0.99	0.12	0.20	0.99	0.26	0.36	0.99	0.29	0.39	0.99	0.31	0.43	0.99	0.18	0.27
55	0.99	0.37	0.45	0.99	0.10	0.18	0.99	0.22	0.33	0.99	0.26	0.36	0.99	0.26	0.38	0.99	0.14	0.23
60	0.99	0.35	0.44	0.99	0.09	0.16	0.99	0.19	0.29	0.99	0.23	0.34	0.99	0.22	0.33	0.99	0.11	0.18
65	0.99	0.33	0.42	0.99	0.08	0.14	0.99	0.17	0.27	0.99	0.20	0.31	0.99	0.18	0.28	0.99	0.08	0.15
70	0.99	0.30	0.41	0.98	0.06	0.11	0.99	0.14	0.23	0.99	0.18	0.28	0.99	0.14	0.24	0.98	0.06	0.11
75	0.99	0.28	0.39	0.96	0.05	0.09	0.99	0.12	0.20	0.99	0.16	0.25	0.99	0.11	0.19	0.96	0.04	0.08
80	0.99	0.25	0.36	0.90	0.04	0.07	0.99	0.10	0.17	0.99	0.13	0.22	0.98	0.07	0.13	0.87	0.03	0.05
85	0.99	0.22	0.33	0.79	0.03	0.05	0.99	0.08	0.14	0.99	0.10	0.18	0.89	0.04	0.08	0.77	0.02	0.04
90	0.99	0.19	0.29	0.66	0.02	0.03	0.97	0.06	0.11	0.98	0.07	0.13	0.55	0.01	0.03	0.57	0.01	0.02
95	0.99	0.14	0.23	0.54	0.01	0.02	0.91	0.04	0.07	0.94	0.05	0.09	0.07	0.00	0.00	0.30	0.00	0.01
100	0.00	0.00	0.00	0.47	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table: Scenario 1, based on 200 datasets: Using SHYKJG1 dataset (>M1 class flares): Method 0 is Neural Network. Method 1 is Linear Regression. Method 2 is Probit Regression. Method 3 is Logit Regression. Method 4 is Random Forest Regression. Method 5 is Support Vector Regresion. Acronym A means ACC, acronym T means TSS and acronym H means HSS metric.

		0			1			2			3			4			5	
%	A	T	H	A	T	H	A	T	H	A	T	H	A	T	H	A	T	H
00	0.20	0.21	0.01	0.06	0.07	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.01	0.02	0.08	0.09	0.01
05	0.00	0.05	0.04	**0.01**	**0.03**	**0.02**	0.01	0.04	0.03	0.00	0.04	0.03	**0.01**	**0.04**	**0.03**	0.00	0.04	0.03
10	0.00	0.06	0.04	0.00	0.04	0.03	0.00	0.05	0.03	0.00	0.05	0.03	0.00	0.05	0.03	0.00	0.05	0.04
15	0.00	0.06	0.04	0.00	0.05	0.04	0.00	0.06	0.04	0.00	0.06	0.04	0.00	0.05	0.04	0.00	0.06	0.04
20	0.00	0.06	0.04	0.00	0.07	0.04	0.00	0.06	0.04	0.00	0.06	0.04	**0.00**	**0.05**	**0.04**	0.00	0.07	0.04
25	0.00	0.06	0.04	0.00	0.07	0.05	0.00	0.06	0.04	0.00	0.06	0.04	0.00	0.06	0.04	0.00	0.06	0.05
30	**0.00**	**0.06**	**0.05**	0.00	0.06	0.05	0.00	0.06	0.04	0.00	0.06	0.04	**0.00**	**0.06**	**0.04**	0.00	0.06	0.05
35	0.00	0.06	0.05	0.00	0.05	0.05	0.00	0.06	0.05	0.00	0.06	0.05	0.00	0.06	0.05	0.00	0.06	0.06
40	0.00	0.06	0.05	0.00	0.06	0.07	0.00	0.06	0.06	0.00	0.06	0.05	0.00	0.06	0.05	0.00	0.06	0.06
45	0.00	0.06	0.05	0.00	0.06	0.08	0.00	0.06	0.06	0.00	0.06	0.05	0.00	0.07	0.06	0.00	0.05	0.06
50	0.00	0.06	0.05	0.00	0.05	0.07	0.00	0.05	0.05	0.00	0.05	0.05	0.00	0.06	0.06	0.00	0.05	0.06
55	0.00	0.06	0.05	0.00	0.04	0.05	0.00	0.05	0.06	0.00	0.05	0.05	0.00	0.06	0.06	0.00	0.04	0.06
60	0.00	0.06	0.06	0.00	0.03	0.04	0.00	0.05	0.06	0.00	0.05	0.05	0.00	0.05	0.06	0.00	0.04	0.06
65	0.00	0.06	0.06	0.00	0.03	0.04	0.00	0.05	0.06	0.00	0.05	0.05	0.00	0.04	0.06	0.00	0.03	0.05
70	0.00	0.06	0.06	0.10	0.03	0.05	0.00	0.04	0.06	0.00	0.04	0.05	0.00	0.04	0.06	0.07	0.03	0.05
75	0.00	0.06	0.06	0.15	0.03	0.05	0.00	0.04	0.06	0.00	0.04	0.06	0.00	0.04	0.06	0.15	0.02	0.04
80	0.00	0.06	0.06	0.28	0.03	0.05	0.00	0.04	0.06	0.00	0.04	0.06	0.10	0.04	0.06	0.32	0.02	0.04
85	0.00	0.05	0.06	0.39	0.02	0.04	0.00	0.03	0.05	0.00	0.04	0.06	0.29	0.03	0.05	0.41	0.02	0.03
90	0.00	0.05	0.07	0.46	0.02	0.03	0.12	0.03	0.05	0.07	0.03	0.06	0.49	0.02	0.03	0.49	0.01	0.02
95	0.00	0.04	0.06	0.49	0.01	0.03	0.26	0.03	0.05	0.22	0.03	0.05	0.26	0.00	0.01	0.45	0.01	0.01
100	0.00	0.00	0.00	0.49	0.01	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table: Scenario 1, based on 200 datasets: Using SHYKJG1 dataset (C-class flares): Method 0 is Neural Network. Method 1 is Linear Regression. Method 2 is Probit Regression. Method 3 is Logit Regression. Method 4 is Random Forest Regression. Method 5 is Support Vector Regresion. Acronym A means ACC, acronym T means TSS and acronym H means HSS metric.

		0			1			2			3			4			5	
%	A	T	H	A	T	H	A	T	H	A	T	H	A	T	H	A	T	H
00	0.00	0.00	0.00	0.54	0.48	0.17	0.00	0.00	0.00	0.00	0.00	0.00	0.48	0.42	0.14	0.20	0.04	0.01
05	0.81	0.72	0.42	0.68	0.62	0.28	0.76	0.68	0.36	0.77	0.69	0.37	0.78	0.71	0.39	0.89	0.69	0.54
10	0.86	0.74	0.50	0.76	0.69	0.36	0.83	0.73	0.45	0.83	0.74	0.46	0.84	0.75	0.48	0.91	0.66	0.58
15	0.88	0.72	0.55	0.83	0.74	0.45	0.87	0.73	0.52	**0.88**	**0.74**	**0.54**	**0.87**	**0.75**	**0.53**	0.91	0.63	0.59
20	0.90	0.70	0.57	0.88	0.74	0.54	0.90	0.71	0.57	0.90	0.72	0.58	**0.89**	**0.74**	**0.57**	0.92	0.61	0.59
25	0.91	0.68	0.59	0.91	0.71	0.59	0.91	0.67	0.59	0.91	0.68	0.60	0.91	0.71	0.60	0.92	0.59	0.59
30	0.91	0.65	0.59	0.92	0.63	0.60	0.92	0.62	0.59	0.92	0.63	0.60	0.92	0.68	0.61	0.92	0.57	0.59
35	**0.92**	**0.61**	**0.60**	0.92	0.56	0.58	0.92	0.57	0.58	0.92	0.59	0.60	**0.92**	**0.64**	**0.62**	0.93	0.55	0.59
40	0.92	0.58	0.59	0.92	0.49	0.56	0.92	0.53	0.57	0.92	0.54	0.58	0.93	0.60	0.62	0.93	0.53	0.58
45	0.93	0.54	0.58	0.92	0.43	0.52	0.92	0.48	0.55	0.93	0.50	0.57	0.93	0.56	0.61	0.93	0.50	0.57
50	0.93	0.51	0.57	0.92	0.38	0.49	0.92	0.45	0.53	0.93	0.47	0.55	0.93	0.52	0.59	0.93	0.48	0.56
55	0.93	0.47	0.56	0.92	0.33	0.45	0.92	0.41	0.51	0.93	0.43	0.53	0.93	0.47	0.57	0.93	0.44	0.53
60	0.93	0.44	0.54	0.92	0.28	0.40	0.92	0.38	0.49	0.93	0.41	0.51	0.93	0.43	0.54	0.92	0.39	0.50
65	0.93	0.41	0.52	0.91	0.23	0.34	0.92	0.35	0.47	0.93	0.38	0.49	0.93	0.39	0.51	0.92	0.35	0.47
70	0.92	0.38	0.49	0.91	0.19	0.29	0.92	0.32	0.43	0.92	0.35	0.47	0.92	0.35	0.48	0.92	0.32	0.43
75	0.92	0.34	0.46	0.91	0.15	0.24	0.92	0.28	0.39	0.92	0.31	0.43	0.92	0.31	0.43	0.92	0.28	0.39
80	0.92	0.31	0.43	0.90	0.12	0.20	0.91	0.24	0.34	0.92	0.26	0.38	0.92	0.27	0.38	0.91	0.24	0.35
85	0.92	0.26	0.38	0.90	0.11	0.17	0.91	0.19	0.29	0.91	0.21	0.32	0.91	0.21	0.32	0.91	0.19	0.29
90	0.89	0.19	0.28	0.90	0.09	0.15	0.90	0.15	0.23	0.91	0.16	0.25	0.91	0.15	0.24	0.91	0.15	0.23
95	0.65	0.06	0.10	0.90	0.07	0.13	0.90	0.11	0.17	0.90	0.11	0.17	0.90	0.07	0.11	0.90	0.10	0.16
100	0.00	0.00	0.00	0.90	0.06	0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

NEW DATA: Monte Carlo simulation for C class flares

Standard deviations of the Skills Scores Values

Table: Scenario 1, based on 200 datasets: Using SHYKJG dataset (C-class flares): Method 0 is Neural Network. Method 1 is Linear Regression. Method 2 is Probit Regression. Method 3 is Logit Regression. Method 4 is Random Forest Regression. Method 5 is Support Vector Regresion. Acronym A means ACC, acronym T means TSS and acronym H means HSS metric.

		0			1			2			3			4			5	
%	A	T	H	A	T	H	A	T	H	A	T	H	A	T	H	A	T	H
00	0.00	0.00	0.00	0.01	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.02	0.01	0.02	0.02	0.00
05	0.01	0.01	0.02	0.01	0.01	0.01	0.03	0.06	0.04	0.01	0.01	0.01	0.01	0.01	0.01	0.00	0.02	0.02
10	0.01	0.01	0.02	0.01	0.01	0.02	0.03	0.07	0.04	0.01	0.01	0.02	0.01	0.01	0.01	0.00	0.02	0.02
15	0.01	0.02	0.02	0.01	0.01	0.02	0.02	0.07	0.04	**0.01**	**0.01**	**0.02**	**0.01**	**0.01**	**0.01**	0.00	0.02	0.02
20	0.00	0.02	0.01	0.01	0.01	0.02	0.03	0.07	0.05	0.00	0.01	0.01	**0.00**	**0.01**	**0.01**	0.00	0.02	0.01
25	0.00	0.02	0.01	0.00	0.02	0.01	0.03	0.06	0.05	0.00	0.02	0.01	0.00	0.02	0.01	0.00	0.02	0.02
30	0.00	0.02	0.01	0.00	0.02	0.01	0.03	0.06	0.05	0.00	0.02	0.01	0.00	0.02	0.01	0.00	0.02	0.02
35	**0.00**	**0.02**	**0.01**	0.00	0.02	0.01	0.03	0.05	0.05	0.00	0.02	0.01	**0.00**	**0.02**	**0.01**	0.00	0.02	0.01
40	0.00	0.02	0.01	0.00	0.02	0.02	0.03	0.05	0.05	0.00	0.02	0.01	0.00	0.02	0.01	0.00	0.02	0.02
45	0.00	0.03	0.02	0.00	0.02	0.02	0.03	0.05	0.05	0.00	0.02	0.02	0.00	0.02	0.01	0.00	0.02	0.02
50	0.00	0.03	0.02	0.00	0.02	0.02	0.03	0.04	0.04	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.02	0.02
55	0.00	0.03	0.02	0.00	0.02	0.02	0.03	0.04	0.04	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.03	0.02
60	0.00	0.03	0.02	0.00	0.02	0.02	0.03	0.04	0.04	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.03	0.03
65	0.00	0.03	0.02	0.00	0.02	0.03	0.03	0.05	0.04	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.03	0.03
70	0.00	0.03	0.03	0.00	0.02	0.03	0.03	0.05	0.04	0.00	0.02	0.02	0.00	0.02	0.02	0.00	0.03	0.03
75	0.00	0.03	0.03	0.00	0.02	0.03	0.03	0.05	0.05	0.00	0.03	0.03	0.00	0.02	0.02	0.00	0.03	0.03
80	0.00	0.03	0.03	0.00	0.02	0.02	0.03	0.06	0.05	0.00	0.03	0.03	0.00	0.02	0.03	0.00	0.03	0.03
85	0.00	0.03	0.04	0.00	0.02	0.02	0.03	0.06	0.06	0.00	0.03	0.04	0.00	0.02	0.03	0.00	0.02	0.03
90	0.14	0.06	0.08	0.00	0.01	0.02	0.03	0.07	0.06	0.00	0.03	0.05	0.00	0.02	0.03	0.00	0.02	0.03
95	0.40	0.06	0.09	0.00	0.01	0.02	0.03	0.07	0.07	0.00	0.03	0.04	0.00	0.02	0.03	0.00	0.02	0.03
100	0.00	0.00	0.00	0.00	0.01	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00