Imperial College London

SPAT-Voulgarakis-1

Detection of Atmospheric Blocking Using Deep Learning

Supervisor:
Dr. Arnaud Czaja

CID:
01192437

Assessor:
Dr. Heather Graven

May 4, 2020

Word count: 9707

Abstract

We present a novel Ground Truth Dataset (GTD) containing the classification of European blocking events over 40 summer periods from 1979 to 2018. The GTD classifications were created by combining the classifications from three different blocking detection methods that utilise the 500hPa geopotential height gradient (GHG), the 500hPa geopotential height anomaly (Z500*) and the column-averaged potential vorticity anomaly (VPV*). The Z500* method was found to contribute 70.8% of the GTD blocking labels. A new automated blocking classification method was developed using a Convolutional Neural Network (CNN) algorithm that was trained on Z500* contour maps classified by the GTD into blocked and non-blocked patterns. The algorithm was found to successfully identify a variety of blocking structures, recognising $(55.0 \pm 18.8)\%$, $(78.1 \pm 10.4)\%$ and $(64.1 \pm 15.5)\%$ of the GHG, Z500* and VPV* specific blocking events respectively, thus outperforming any individual blocking detection method. The predicted events that did not match the GTD-labelled blocks were thoroughly analysed. Some were found at the beginning, middle or end of the GTD-labelled event and some were found to be unrelated to the GTD-labelled blocks. The former events present the potential to indicate the onset and decay stages of blocking patterns, whereas the latter events indicate where the currently established methods fail to capture the general features of atmospheric blocking. Therefore, the outputs from our CNN highlight the statistically significant events that could be the turning point in the study of blocking patterns. We conclude by recommending to combine more than one blocking detection method for any future development of new classification models. This will help such models to acquire a deeper understanding of general blocking climatology.

Contents

1 Introduction **4**

2 Theory **5**
2.1 Atmospheric Circulation Concepts 5
2.2 Assessment of Previous Blocking Detection Methods 6
2.3 Basic Concepts of Predictive Modeling and Machine Learning 8

3 Methods **8**
3.1 Ground Truth Dataset Compilation 8
3.2 The Algorithm 9
3.2.1 Evaluation of Algorithm Performance - F1 Scores 9
3.2.2 The Algorithm Structure 10
3.2.3 Input Data Engineering 13

4 Results **15**
4.1 Analysis of the Blocking Indices 15
4.2 CNN Performance Analysis 16
4.3 Analysis of the Domain Size Dependency 18
4.4 Variation of Input Data 20

5 Discussion of Results **21**
5.1 Agreement of Blocking Indices 21
5.2 Algorithm Performance 22
5.3 Dependency of CNN Performance on the Input Data Format 23

6 Conclusion **27**

Appendices **28**

A Hyper-Parameter Tuning for the Learning Rate **28**

B An Example of a Masked False Negative Event Detected by the CNN **28**

1 Introduction

Atmospheric blocking is a large scale meteorological event that obstructs the usual air circulation patterns. It manifests itself as a region defined by a persistent high pressure system that hinders the progression of synoptic weather systems over that region [1, 2]. Depending on the persistence of the anticyclonic high, the variability of the local weather could be reduced for a few days up to several weeks [3]. Thus, the blocks amplify the intensity of local weather conditions and can have drastic societal, ecological and health impacts. Specifically, the Euro-Atlantic blocks are known to enhance the likelihood of the continental heatwaves in summer, and the oceanic cold spells in winter [4]. The study of atmospheric blocking is challenging as this phenomenon is associated with a variety of pressure patterns, locations (i.e. mid-latitudes in the Northern and Southern Hemispheres (NH,SH)) and seasons. For example,

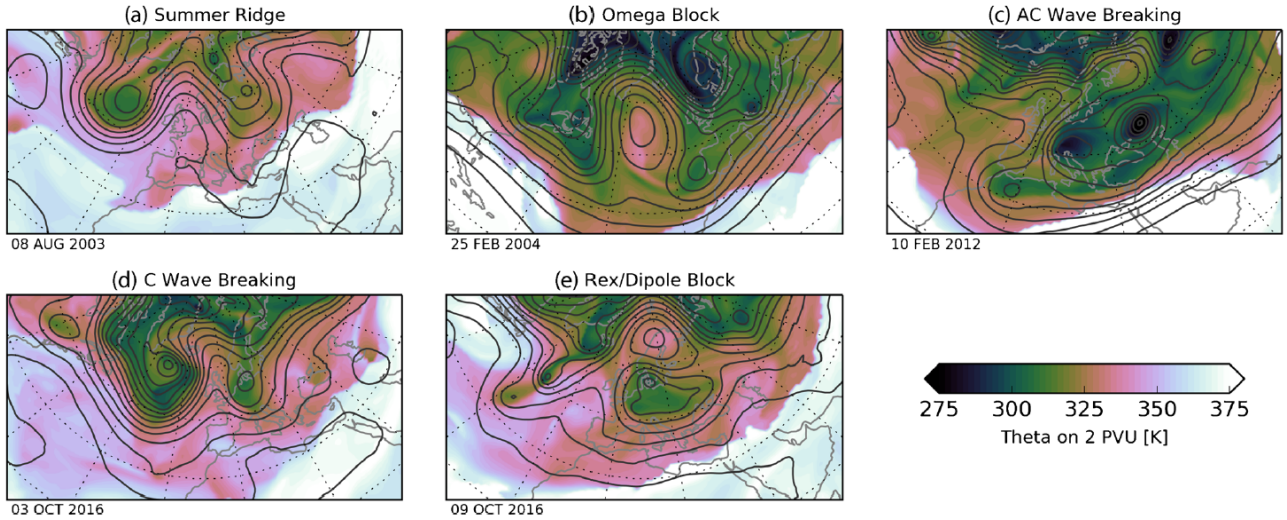

Figure 1: The variation of the North Atlantic blocking patterns taken from [4]. Snapshots of the potential temperature θ (coloured) and $Z500$ (contours).

blocking structure diversity over the mid-latitudes in NH is considerable (e.g. Figure 1). In the European region summer blocking is known to occur less frequently than winter blocking, indicating a strong seasonal dependency of the phenomenon and making its prediction more difficult [5]. Moreover, the physical processes involved in the onset, maintenance and decay stages of blocking are still the topics of ongoing research.

Our aim is to build a classification model that accurately detects summer blocking events in Europe using deep neural networks, a subset of Machine Learning (ML). The ERA5 reanalysis datasets at $1^\circ \times 1^\circ$ spatial resolution from the European Center of Medium-Range Weather Forecasts are used for this project [6]. The daily midnight snapshots of geopotential height (Z500) and potential vorticity (PV) were used. The data was taken for 40 June-July-August (JJA) periods (1st of June 1979 - 31st of August 2018) for $75^\circ N - 30^\circ N$ in latitude and $10^\circ W - 40^\circ E$ in longitude, taking the reference longitude to be the Greenwich meridian. The meteorological terms used in this project are discussed in Section 2.1. Unlike Verdecchia and Visconti 1996 [7], we perform Neural Network (NN) training on classification from more than one objective blocking detection method, which are discussed in Section 2.2, followed by a quick introduction to ML in Section 2.3. We also perform manual filtering of all blocking classifications produced by these methods before creating a Ground Truth Dataset (GTD) of historic blocking events, described in Section 3.1. The architecture of the Convolutional NN (CNN) and techniques used to improve its performance such as architecture optimasation, cross-validation and input data engineering are discussed in Section 3.2. Finally the results and their discussion are presented in Chapters 4 and 5, respectively.

2 Theory

2.1 Atmospheric Circulation Concepts

Atmospheric Blocking

The American Meteorological Society (AMS) defines atmospheric blocking in the NH using three conditions: (1) persistent obstruction of the usual West-to-East air flow in the mid-latitudes, (2) pronounced meridional (i.e. along the line of latitude) flow and (3) anticyclonic circulation (clockwise in the NH) at high latitudes accompanying cyclonic circulation at low latitudes [8]. This definition is often used for the evaluation of blocking detection methods as it covers more than one structural manifestation of a blocked air flow. In this project we consider three established algorithms that satisfy different AMS criteria and are adapted from the original papers: Tibaldi and Molteni 1990 (henceforth TM90) [9], Dole and Gordon 1983 (DG83) [10] and Schwierz et.al. 2004 (S04) [11]. Two methods, TM90 and DG83, are based on geopotential height (Z500) variable and the S04 method is based on vertically-averaged potential vorticity (VPV).

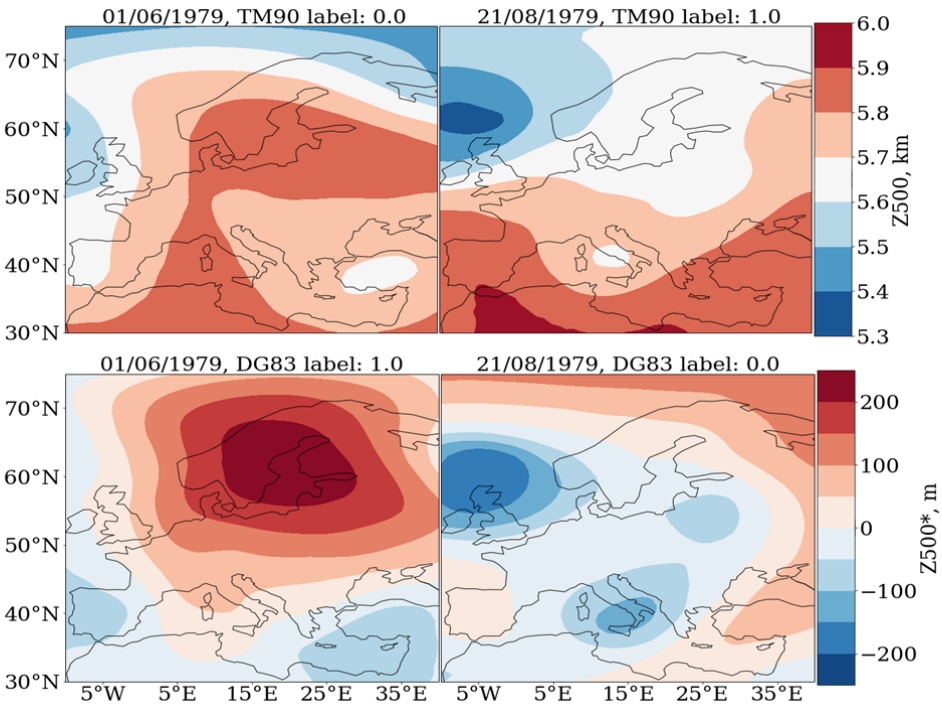

Figure 2: Examples of Z500 (top) and Z500* (bottom) contours plotted on top of the European coast-line (black solid line) and dated 01/06/1979 (left) and 21/08/1979 (right).

The geopotential height Z is defined as the height of the surface with constant pressure relative to sea level. Half of the atmosphere's mass is located above the pressure surface at 500hPa, the altitude of which - generally around $Z500 = 5.5$ km - can vary depending on temperature. It follows from the expansion of hot air that warmer air columns will be at higher pressures - and hence have a larger Z500 - then colder, lower pressure air columns. In the NH, high pressure systems have clockwise circulation (i.e. anticyclonic) and low pressure systems circulate anticlockwise (i.e. cyclones). Potential Vorticity (PV), expressed in isentropic coordinates (i.e. along constant potential temperature θ), is defined as:

$$PV = -g(f(\phi) + \zeta_\theta)\frac{\partial \theta}{\partial p} \tag{1}$$

where $f(\phi)$ is the planetary vorticity, which is a function of latitude ϕ, and ζ_θ the component of relative vorticity that is directed along the normal to the isentropic surface [12]. The planetary vorticity is the component of the Earth's rotation along the local normal and the relative vorticity is the fluid's circulation relative to earth. The $g\frac{\partial\theta}{\partial p}$ term determines the height of the air column and depends on the gravitational constant g, θ and pressure p. Isentropic PV (IPV)

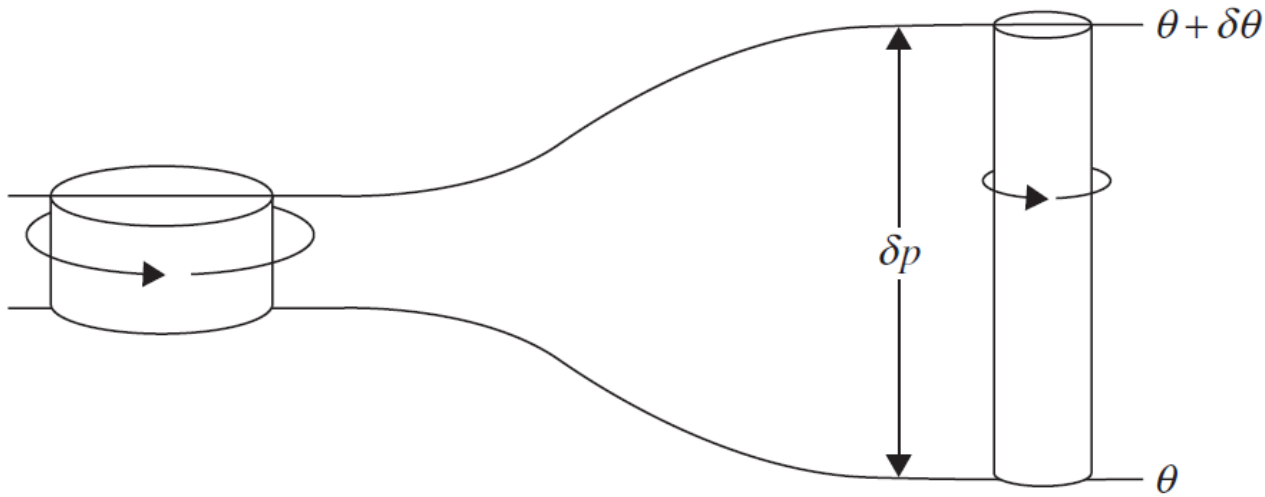

Figure 3: The conservation of isentropic PV for a column of air moving along a constant latitude. Taken from [12].

is conserved following a given air parcel, therefore any changes in ζ_θ at a constant ϕ and θ, cause a stretching or a shrinking of the air column (i.e. an increase in ∂p as shown on Figure 3). Hence, blocking anticyclones and other blocking structures that are reflected on isobaric Z500 maps are said to correspond with distinctive features on the IPV maps [13]. IPV is measured in PV units (PVU), where 1 PVU $= 1 \times 10^{-6}$ K m^{2}kg^{-1}s^{-1}. The minus sign in Eq.(1) indicates that a high pressure system will have a negative PV value.

2.2 Assessment of Previous Blocking Detection Methods

The TM90 method utilises the total Z500-field by computing the Z500 spatial gradient (GHG) along the North (GHGN) and South (GHGS) directions from the central latitude, $\phi_c = 50°N$, as follows:

$$GHGN = \frac{Z500(\phi_n) - Z500(\phi_c)}{\Delta\phi} \quad ; \quad GHGS = \frac{Z500(\phi_c) - Z500(\phi_s)}{\Delta\phi} \tag{2}$$

where the $\Delta\phi$ threshold can be varied, $\phi_s = \phi_c - \Delta\phi$ and $\phi_n = \phi_c + \Delta\phi$. The DG83 and S04 methods consider the anomaly fields which are defined by departures from the Long Term Daily Mean (LTDM). The LTDM is calculated for each gridpoint and each of the 365 days in a year over the 40 years (described by [14]). The anomaly threshold α for both variables was set to be 1.5 times the standard deviation of that variable, V^*, from its LTDM, σ_{V^*}:

$$\alpha = \begin{cases} 1.5 \times \sigma_{V*} & \text{, if } \alpha \geq \alpha_{min} \\ \alpha_{min} & \text{, otherwise} \end{cases} \tag{3}$$

where a minimum threshold α_{min} for the gridpoints with little or no blocking activity was also introduced. The limitations, tendencies and modifications to each method, as adopted from Pinheiro et al 2019 [15], are described in Table 1.

Table 1: Summary of Method Definitions and Limitations. Adopted from [15].

Method	TM90 [9]	DG83 [10]	S04 [11]
Variable	Z500 latitudinal gradient GHGN,GHGS Eq.(2)	Z500 anomaly (Z500*) + latitudinal scailing	Vertically averaged PV anomaly (VPV*)
Definition	Change in Z500 over $\Delta\phi = 20°$ above and below the ϕ_c	Positive Z500* wrt climatological mean	Reversal of flow wrt climatological mean
Blocking condition	GHGN < -10 m/deg lat GHGS > 0 m/deg lat over 4 days per gridpoint	Z500* $\geq$ 100m over 10 days, per gridpoint	VPV* ≤ -1.2 PVU, with at least 70% overlap between contours over 5 days
Changes to Methods [15]	Extended to latitudes between 27.5°-72.5°, 5 days persistence and countour overlap	Varying anomaly threshold, α Eq.(3) 5 days persistence and countour overlap	Varying anomaly threshold, α Eq.(3)
Limitations	Based on total fields. No reference to mean climatology. Does not necessarily satisfy AMS(1).	Possibility of over-prediction of blocking events. Satisfies all AMS cretiria.	Can identify unobstructed flow as blocked. Somewhat satisfies AMS(2) and AMS(3).

The TM90 method looks at how much the geopotential height changes over the $\Delta\phi = 20°$ of latitude in both meridional directions from the central latitude, ϕ_c. Both blocking conditions have to be met for a pattern to be identified as blocked, see Table 1. This method is based on the assumption that geopotential height surfaces are latitudinally equidistant from each other. This assumption does not appreciate the pronounced non-linearity of the blocking structures and is shown not to be satisfied for most Z500 contours in the top row of Figure 2. This method is also limited to the Z500 values that change significantly over the 20° in latitude. If the $Z500(\phi_c + 20°) \approx Z500(\phi_c)$, the $GHG \approx 0$ meaning that this method would not be able to recognise more complex blocking structures for which such a condition is met. This effect is partially taken into account by shifting the latitudinal band (i.e. the ϕ_c, ϕ_n and ϕ_s) by $\delta\phi = \pm 2.5°$, thus obtaining three meridional gradients for GHGN and GHGN. Even though this change extends the 1-D TM90 approach to 2-D, it is still not sufficient for something as diverse as blocking shapes, especially for larger domains than the one shown in Figure 2. The DG83 method emphasises the areas of anomalously high pressure indicated by the positive Z500*, thus highlighting the 2-D anticyclonic structures. This method is widely used as the Z500* contours give a good visual representation of blocked pressure patterns, as shown in the bottom row of Figure 2. It also satisfies all the criteria of the AMS definition of blocking and therefore has a tendency to over-predict blocking phenomena. Similarly to the DG83 method, the S04 approach is based on anomalies that exceed a given threshold value, α Eq.(3). It considers the negative anomalies, in the NH, in the 150-500hPa column-averaged PV values and is said to capture the 3-D structure of atmospheric circulation patterns. However, the use of vertically-averaged PV (VPV) values, as opposed to IPV, no longer guarantees the conservation of the flow properties for the S04-label computation. Each blocking detection method thus focuses on a different subset of the blocking features and therefore none of them represent the general blocking climatology individually.

2.3 Basic Concepts of Predictive Modeling and Machine Learning

A predictive model is a mathematical tool developed to predict an outcome from the input data [16]. Hence there are three key components of such a model: the input data, x, the fitting function, m, and the output, y. This statement in an equation form, $y = m(x)$, indicates that the only components that can be altered in order to achieve the desired y are m (algorithm optimisation , Section 3.2.2) and x (data engineering, Section 3.2.3). A classification task uses a supervised machine learning approach in which the computer program learns from x and its known classifications y. The algorithm then uses this learning to classify new observations, i.e. to predict $y =$ on previously unseen observations x'. It is therefore practical to split your main dataset (X and Y) into two subsets: the train set (x,y), which is used for algorithm learning (or training), and the test set(x',y'), used for validating the algorithm's performance. During training the x is fit with randomly initialised coefficients that results in a set of predictions y_p. The deviation of y_p from the y is calculated (i.e. the loss) and is used to update the coefficients. This cycle is repeated until the global minimum of the loss function is found. Optimising a set of hyper-parameters such as learning rate, number of epochs and batch size can help the algorithm to reach a global minimum of the loss function faster. The learning rate defines the speed at which the model updates its coefficients. If the value is too large it can overshoot the global minimum, and if it is too small and the algorithm will take a long time to converge [17]. The number of epochs defines the number of times the whole training data is shown to the algorithm while learning. The batch size is the number of sub-samples that are shown to the algorithm at a time. Choosing the optimal set of these hyper-parameters is important as it helps to minimise loss and improve the algorithm's performance (known as hyper-parameter tuning, described in Section 3.2.2).

3 Methods

3.1 Ground Truth Dataset Compilation

The blocking indices from the three methods described in Section 2.2 were combined together to create the GTD, which is shown in Figure 4. This allowed us to take the vast variety of blocking structures into account. A region was identified as blocked if at least one of the methods was satisfied. That region had to also be identified as blocked during the manual labelling process. This process involved the inspection of the Z500* contours by a group of three physics students and three academics. Three things were assessed before the manual classification could be performed: the size of the positive Z500*, the changes of the size and location of the anomaly (i.e. persistence and overlap criteria) and the area coverage of the domain. The inputs from an individual were found to be at least in 95% agreement with the inputs from each of the other individuals. These inputs for each contour, i, were then added together for every event and averaged. When the average was larger than 0.5, the pattern was labeled as blocked, otherwise it was labeled as non-blocked. This allowed the bias (i.e. the knowledge of significant historic blocking events) to be eliminated whilst providing a valid determination of blocking from the academics. The GTD totalled in 600 blocked days and was used to train the CNN, described in the next section. Out of 600 labels, 425 were taken from the DG83 method, 135 were taken from TM90 and 133 from S04, respectively contributing 70.8%, 22.5% and 22.2% of the blocked labels to the GTD.

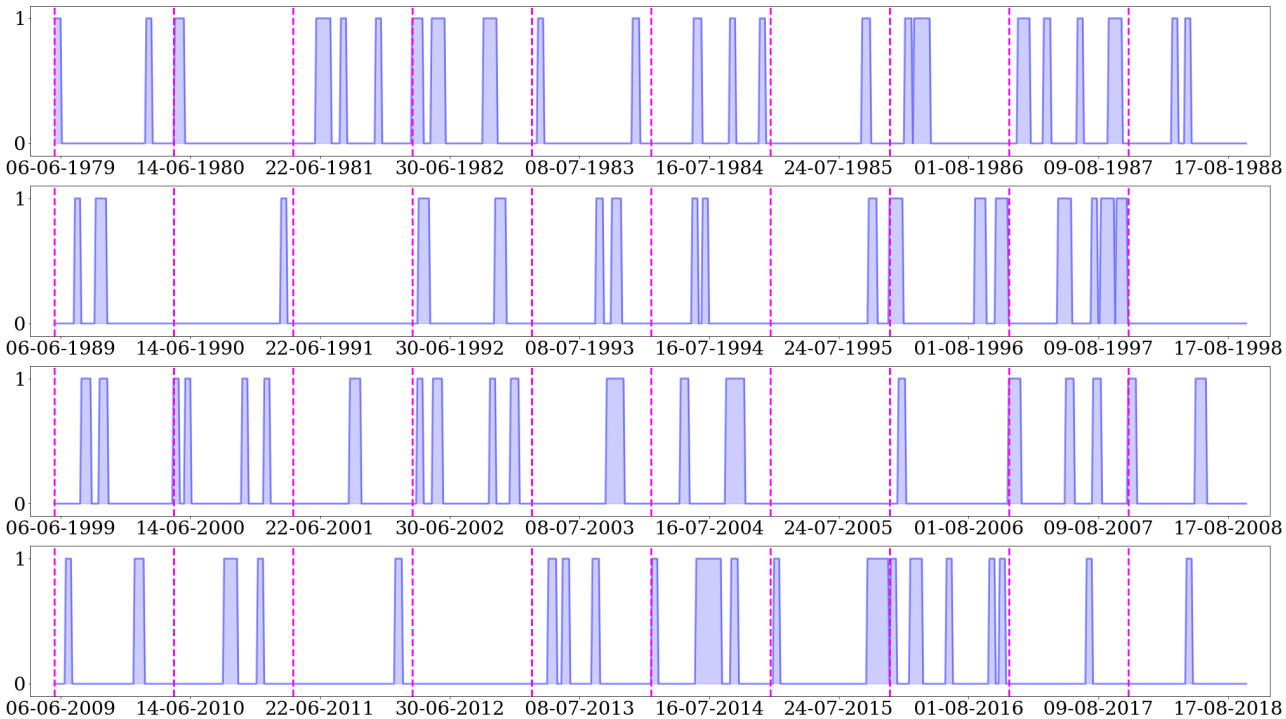

Figure 4: The historic dataset of European blocking classification for JJA periods between the 1st of June 1979 and the 31st of August 2018. A blocking event is indexed as $i = 1$, non-blocked $i = 0$.

3.2 The Algorithm

3.2.1 Evaluation of Algorithm Performance - F1 Scores

The input dataset consists of 3680 observations, 600 of which are labelled as blocked by the GTD. The bias towards the non-blocked days in the data is known as the imbalanced class problem in ML. The data has a disproportionate number of observations for each class, 16.3% of blocked days and 83.7% of clear (non-blocked) days. If the standard accuracy metric is used and there is no detected blocking by the algorithm, the algorithm's performance will have 83.7% accuracy. Thus, the simple accuracy score is misleading and indicates that a different metric should be used to gain a more insightful understanding of the algorithm's performance. Therefore, we propose using F1 scores, as it penalises the classifier if it always predicts the negative (non-blocked) class on the unseen observations [18]. It does so by considering precision, $\mathcal{P}$, and recall, $\mathcal{R}$, which describe a blocked event as a positive and a non-blocked event as a negative. Precision, recall and the F1 score f_1 are calculated as follows:

$$\mathcal{P} = \frac{d_{TP}}{(d_{TP} + d_{FP})} \quad \mathcal{R} = \frac{d_{TP}}{(d_{TP} + d_{FN})} \quad \text{, combined to give:} \quad f_1 = \frac{2\mathcal{P}\mathcal{R}}{\mathcal{P} + \mathcal{R}} \tag{4}$$

where $\mathcal{P}$ is defined as the number of correct (true) positive results (i.e. correctly detected blocked days), d_{TP}, over the number of all detected positives, d_P. d_P is the sum of positives that were detected as blocked (True Positives, TP) and positives that were detected to be non-blocked (False Positives, FP), $d_P = d_{TP} + d_{FP}$. $\mathcal{R}$ is defined as d_{TP} divided by the number of samples that have been detected as blocked, d_b. This consists of d_{TP} and the number of negatives that were identified as blocked by the algorithm (False Negatives, FN), d_{FN}, giving $d_b = d_{TP} + d_{FN}$. Both $\mathcal{P}$ and $\mathcal{R}$ take values between 0 and 1. Hence the F1 score, f_1 Eq.(4) has a range of $0 \leq f_1 \leq 1$. For the example mentioned above, where the algorithm predicts all events to be not blocked (i.e. 3680 negatives). This would mean that $d_{FP} = 600$, $d_{TP} = d_{FN} = 0$

and $d_{TN} = 3680 - 600 = 3080$ giving us $\mathcal{P} = \mathcal{R} = f_1 = 0$, which more accurately describes the ability of the algorithm to detect blocking. The opposite example of the algorithm only predicting blocking would give us $d_{TP} = 600$, $d_{FN} = 3080$ and $d_{TN} = d_{FP} = 0$, giving us $\mathcal{P} = 1$, $\mathcal{R} = 0.165$ and $f_1 = 0.163 \times 2/1.163 = 0.280$. Hence, the algorithm's performance can only be improved by reducing d_{FN} and d_{FP}, which will result in $\mathcal{P} \to 1$ and $\mathcal{R} \to 1$, giving the best performance $f_1 = 1$. It is worth noting that in an imbalanced dataset, the correctly detected blocked day is favoured by the F1 score metric, as opposed to the correctly detected non-blocked day.

3.2.2 The Algorithm Structure

The CNN Architecture

Following the discussion in Section 1 the detection of blocking patterns is not trivial and can be challenging, even for a specialist in the field. With an outstanding ability of discovering niche structures in high dimensional data, deep learning algorithms have outperformed other ML models, especially in image recognition tasks [19]. In this project a CNN classification algorithm is used as, unlike a general artificial neural network, it utilises a combination of convolutional and linear layers, described below. This makes the CNN more suitable for pattern recognition tasks as the convolutional layers are especially good at picking out the features of an object, regardless of the object's position in an image [20].

One of the typical CNN structures is shown in Figure 5 (a) and (c), which consists of an input layer (e.g. Z500* maps), 2 convolutional layers, 2 Max pooling layers, linear layers and an output layer (i.e. the classification). Every convolutional layer takes a 2D input array of numbers (image). Then a filter, starting from the top of the image, performs a dot product with a chunk of the image and outputs a single number. As the filter slides through the image, it creates a smaller 2D array: a feature map (see Figure 5 (a)). The feature map is then taken through a max pooling layer which picks out the top feature (the pixel with the largest value, see Figure 5 (b)) out of every filter subarray. As the original image goes through the convolutional and max pooling layers, it is reduced in size as the model learns the features of the image. This part of the algorithm is the traditional pattern recognition structure [21]. The output of the last 2D layer is turned into a 1D array of numbers that is passed to the linear layers, consisting of single nodes as shown on Figure 5 (c). Each node in one layer is connected to all the nodes in the subsequent layer, which enables the algorithm to learn how different features are correlated in an image. The final output layer is the algorithm's decision of whether the image belongs to the blocked ($i = 1$) or the non-blocked class ($i = 0$). The linear layers represent the classification part of the algorithm [21].

The complexity of the CNN can be changed by alternating the algorithm's architecture (i.e. the number of convolutional, max pooling or linear layers), the size of the layers and the size of the filters, thus optimising the model's ability to classify blocking [22]. F1 scores are used for grading the algorithm's performance, therefore, the optimal algorithm structure will result in the highest F1 score. The CNN architecture has an almost infinite number of degrees of freedom which means that the random picking of the model's internal parameters does not guarantee the optimal model structure. However, even if the optimal architecture is found, the CNN can still perform poorly on previously unseen data. In such a case, the complex model is said to have over-fitted the training dataset, meaning that it has learned a small part of the blocking features instead of the general trend. A cross-validation technique can be implemented to find the optimal CNN architecture and can also be used to indicate whether or not our algorithm is over-fitting the data [23].

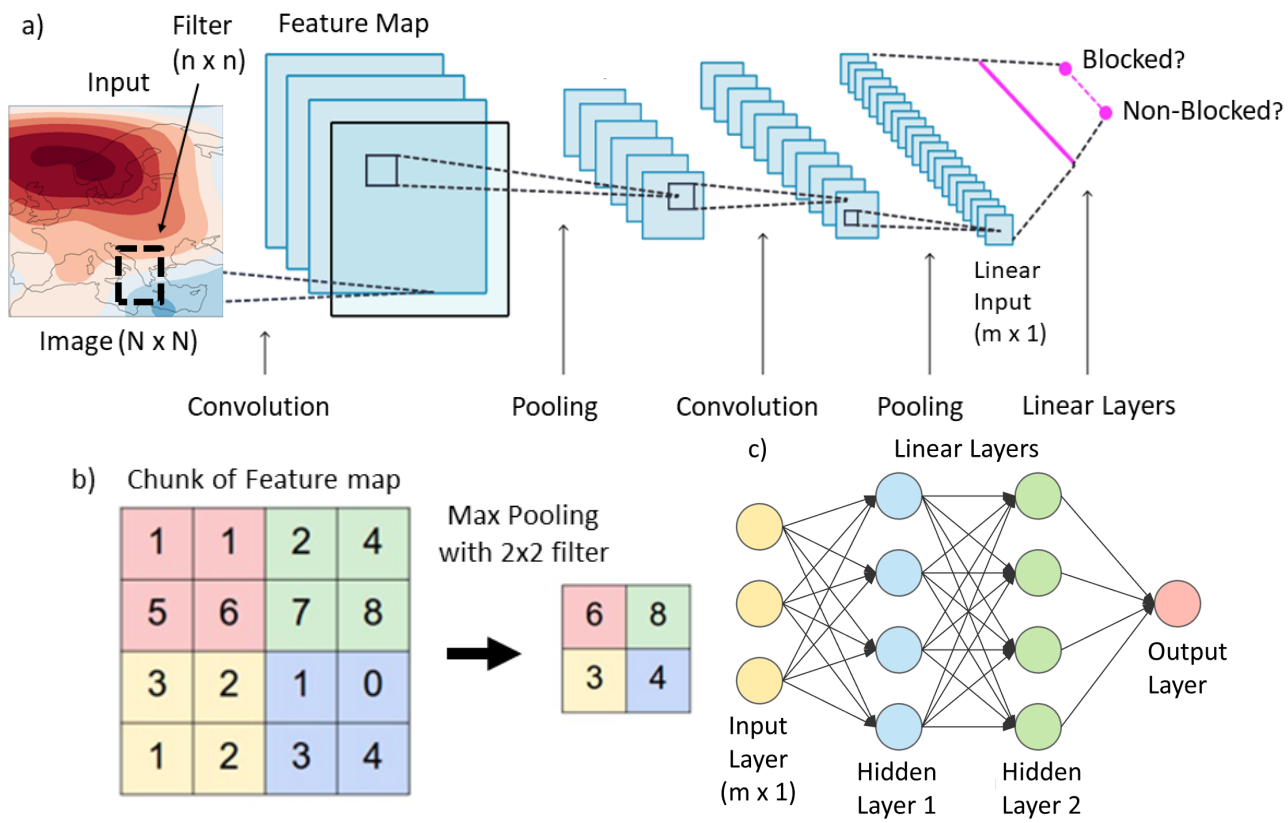

Figure 5: The structure of CNN algorithm including the 3 convolutional and 2 max pooling layers (a), pattern recognition part, and the linear layers (c), the classification part, are shown. Each convolutional layer results in a feature map after performing convolution between the input $N \times N$ image with an $n \times n$ filter. The dimension of the feature map is smaller than the input image. Each max pooling layer results in further reduction of the input array as shown in (b). The output of the pattern recognition part of the algorithm in (a) is the input of the linear layers in (c). The output of the linear layers is the output of the whole algorithm, resulting in classification of the input pattern. Adapted from [21].

Cross-Validation

The K-Fold Cross-Validation technique is used to find the optimal model architecture. The main dataset was split into $k+1$ equal subsets. The last subset, henceforth referred to as the test set, was left for final validation (see Figure 6). The remaining k subsets were used for cross-validated training. Splitting the data into k subsets allows to train the algorithm on k different combinations (folds) of these subsets as shown in Figure 6. For each fold, the model is trained on $k-1$ subsets (purple) and then validated on the unused subset (blue), as shown in Figure 6. F1 scores were recorded for each validation set, giving a value per fold. The most optimal number of folds used for cross-validation was found to be $k = 3$. It provided the most consistent validation set F1 scores across the three folds, meaning that each subset is representative of the blocking patterns we are trying to classify.

Once the number of folds was identified, the 3-fold validation was used to pick the remaining hyper-parameters and the model's architecture. Hyper-parameters tuning was performed for the learning rate, the number of epochs and the batch size (see Section 2.3). Similarly to the k-fold number optimisation, the model's performance was evaluated with the validation set and with the test set for every fold. Thus the total of the three F1 scores for each validation and testing stages was recorded. The mean, $\bar{f}_1$, and standard deviation, σ, of the F1 scores for the validation sets, $\bar{f}_{1v}$ and σ_v, and the test set, $\bar{f}_{1t}$ and σ_t, were found for each choice of

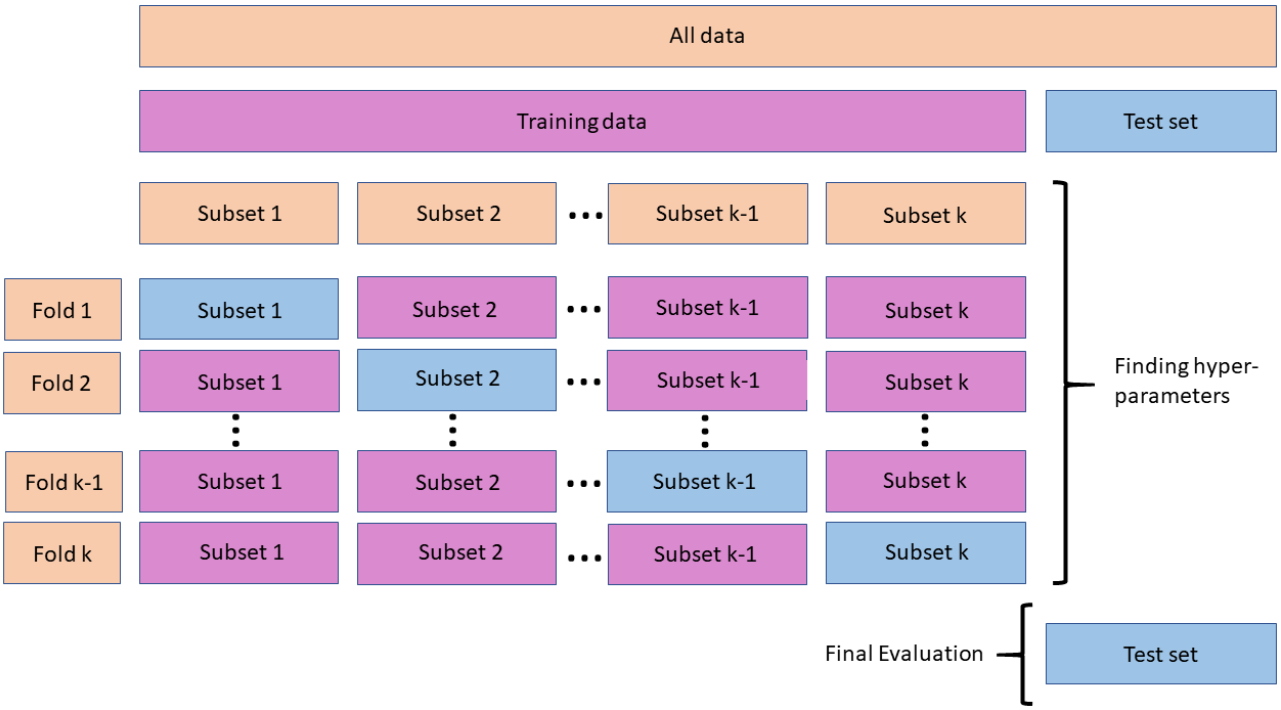

Figure 6: Structure of data splitting for K-Fold Cross-Validation. The main dataset (orange) is split into train (purple) and test(blue) sets with k:1 ratio. The training set is used for cross-validation, and split into k equal subsets. Each fold represents a training of the algorithm on a different combination of training and validation subsets.

Table 2: Hyper-parameter Tuning for the Number of Epochs and Batch Size

# of Epochs	$\bar{f}_{1_v}$	σ_v	$\bar{f}_{1_t}$	σ_t	Batch size	$\bar{f}_{1_v}$	σ_v	$\bar{f}_{1_t}$	σ_t
10	0.434	0.308	0.432	0.306	100	0.723	0.044	0.701	0.050
20	0.720	0.057	0.693	0.010	200	0.712	0.035	0.714	0.029
40	0.717	0.025	0.717	0.027	250	0.466	0.330	0.473	0.336
60	0.710	0.044	0.721	0.041	350	0.687	0.067	0.696	0.018
80	0.638	0.042	0.645	0.072	500	0.710	0.037	0.677	0.017
100	0.689	0.023	0.659	0.037	750	0.651	0.090	0.572	0.175
500	0.447	0.066	0.403	0.057	1000	0.700	0.023	0.694	0.031
1000	0.465	0.063	0.491	0.070	1200	0.223	0.315	0.223	0.312

hyper-parameter, see Table 2. The chosen hyper-parameters were picked such that $\bar{f}_{1_v}$ closely matches $\bar{f}_{1_t}$ and $\bar{f}_{1_v} \approx \bar{f}_{1_t}$ (henceforth $\bar{f}_1$) are as high as possible. Both σ_v and σ_t needed to be as low as possible so that the model was not over-fitting. The most optimal number of epochs, batch size and learning rate were found to be 40, 200 and 1×10^{-4} respectively (as indicated in gray shading on Table 2 and Table 11 in Appendix A) .

3.2.3 Input Data Engineering

Overlap and Persistence Criteria Implementation

Input data engineering is used to aid better learning for a given model. Regardless of model complexity, it can be challenging to detect the physical features of blocking, such as persistence and overlap criteria, when raw data is taken as an input. The window method was implemented on the input data by stacking a $d = 3$ number of Z500* contour maps before and after the labelled contour. This transforms our main dataset of 3680 Z500* contours (1/day) to 3440 lots of $(2d + 1 = 7)$-day windows (shown in Figure 7). The middle day in the window is used

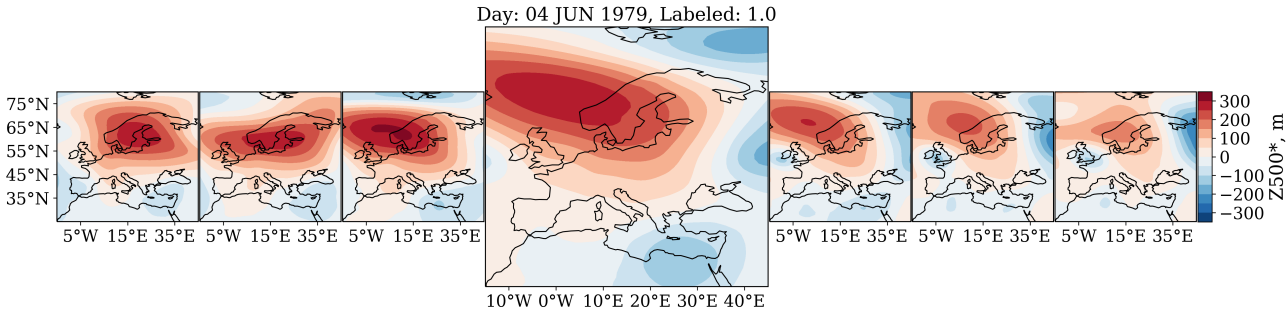

Figure 7: An example of an input sample that was converted from one Z500* map to 7-day Z500* map, stacked along the latitude axis. The label of the sample is defined by the middle day.

as the label for the whole pattern, which means that $2d = 6$ labels are abolished from every summer period. Out of $40 \times 6 = 240$ abolished indices, 34 had $i = 1$ label, leaving our GTD with only 566 blocked days out of 3440. The contour's domain size and the type of input data (Z500* to IPV) can also be changed.

Data Re-sampling

Even though applying the F1 metric helps evaluate the algorithm's performance on an imbalanced dataset, the algorithm can still struggle to predict blocking in the first place. Most machine learning algorithms are designed to maximise accuracy and reduce error, performing best when the number of samples in each class is approximately equal [18]. Different re-sampling techniques can be used to either increase the number of blocked days or decrease the number of non-blocked days in the dataset. The former is known as over-sampling or up-sampling of the minority class and is defined as adding more randomly selected copies of the minority class samples. The latter is the under-sampling or down-sampling of the majority class and is defined as randomly removing samples from the majority class. Since the total number of observations in our dataset is almost seven times the number of blocked days, when randomly up-sampling data some of the blocked observations will be copied more than seven times. The algorithm will see that and assign more importance to the most frequently observed samples, which can impair its ability to learn the general features of blocking and cause over-fitting. When down-sampling is used, the total number of observations is reduced to double the number of samples in the minority class, $566 \times 2 = 1132$. This can influence the algorithm's ability to distinguish between a blocked and non-blocked event in the future if the information that is randomly removed is valuable. The decrease in the number of observations available can also cause the algorithm to under-fit the data. It is not possible to distinguish between samples based on their importance for blocking detection before training. However, if valuable samples are to be removed, the output F1 scores across multiple algorithmic runs are guaranteed to be low and inconsistent.

To eliminate such inconsistencies (or noise) the algorithm was run for 2×10 times fully as

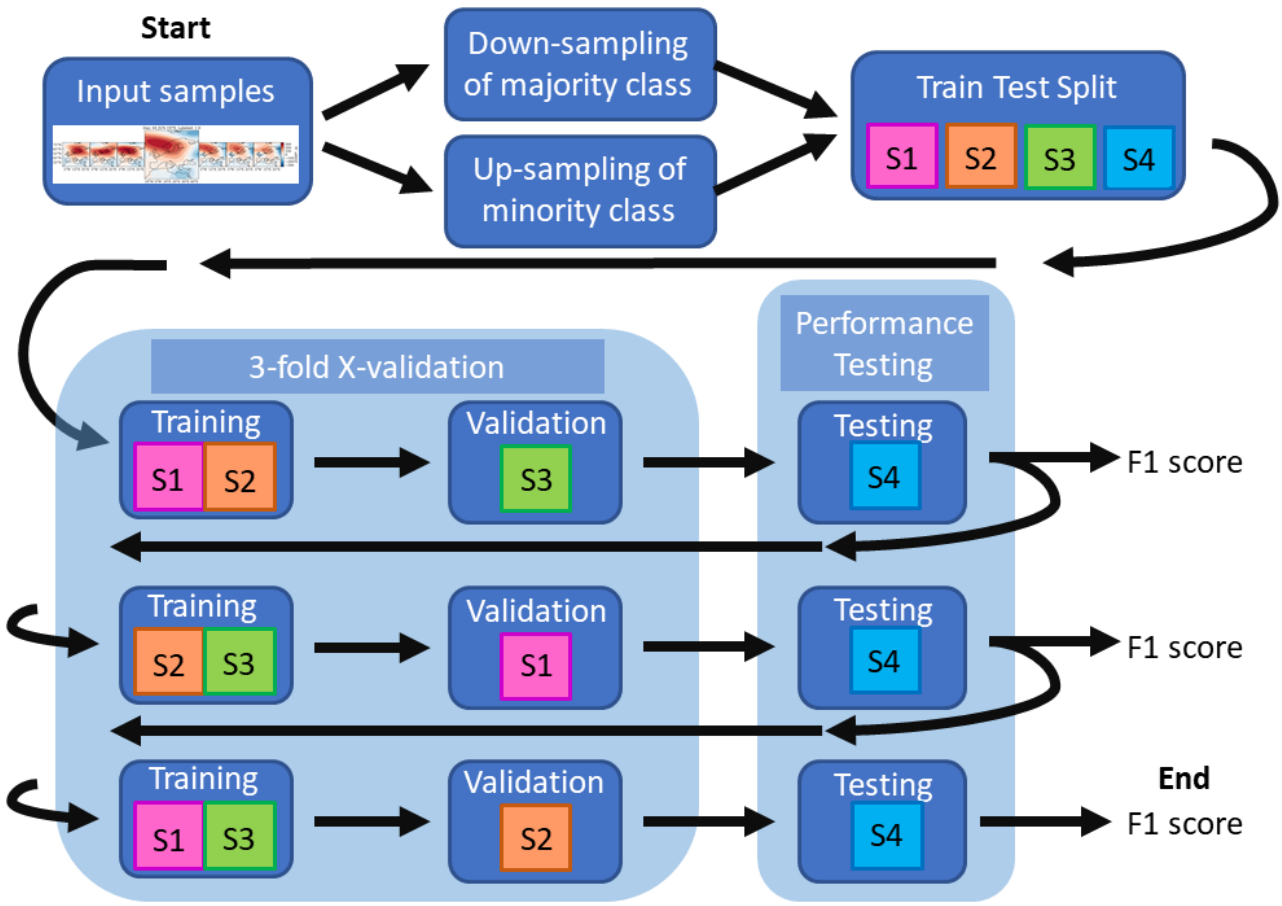

Figure 8: The protocol for one full run of the algorithm is shown. At the start the 7-day window samples are either the down-sampled or up-sampled. The resulting samples are split into 3-fold cross-validated training (S1, S2, S3) and testing (S4) sets. An array of predicted labels and an F1 score are given after every cross-validated training. The full run of the algorithm results in three F1 scores and three arrays of predicted labels.

Table 3: Average 1-Run F1 Scores, $\bar{f}_1$, for Under-sampled Data

	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10
Lot 1	0.455	0.665	0.490	0.454	0.692	0.473	0.465	0.686	0.697	0.712
Lot 2	0.485	0.664	0.457	0.474	0.459	0.669	0.672	0.474	0.581	0.711

shown in Figure 8. Each run resulted in three F1 scores, which were averaged to give $\bar{f}_1$. 2×10 $\bar{f}_1$ (see Tables 4 and 3) were averaged to give the total average F1 score, $\bar{\mathcal{F}}_1$. The $\bar{\mathcal{F}}_1$ therefore gives a better indication of the algorithm's performance independent of the noise introduced by re-sampling.

Both up-sampling and down-sampling techniques were used separately for two lots of 10 full runs to determine the optimal re-sampling technique. The 20-run average, $\bar{\mathcal{F}}_1$, was found to be 0.572 for down-sampled data and 0.715 for up-sampled data. From Tables 3 and 4, it is seen that using up-sampling techniques results in more consistent and higher average F1 scores, $\bar{f}_1$. Thus, the up-sampling method was used for further analysis and at least ten complete runs were performed for all future investigations.

Table 4: Average 1-Run F1 Scores, $\bar{f}_1$, for Over-sampled Data

	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10
Lot 1	0.701	0.702	0.705	0.657	0.464	0.449	0.640	0.670	0.657	0.654
Lot 2	0.693	0.479	0.708	0.688	0.675	0.680	0.718	0.726	0.658	0.670

4 Results

4.1 Analysis of the Blocking Indices

The three methods (TM90, DG83, S04) were used to calculate the blocking labels using the Z500, Z500* and VPV* data for 40 years of European summers (92 day periods). The blocking labels for each method consisted of an array, $\mathcal{X}$, of 3680 labels (or indices, i), with each label being either $i = 1$ or $i = 0$. The three index arrays ($\mathcal{X}_{TM90}$, $\mathcal{X}_{DG83}$, $\mathcal{X}_{S04}$, shown on Figure 9) were compared to each other and to the GTD, $\mathcal{X}_{GTD}$. The total number of blocked days,

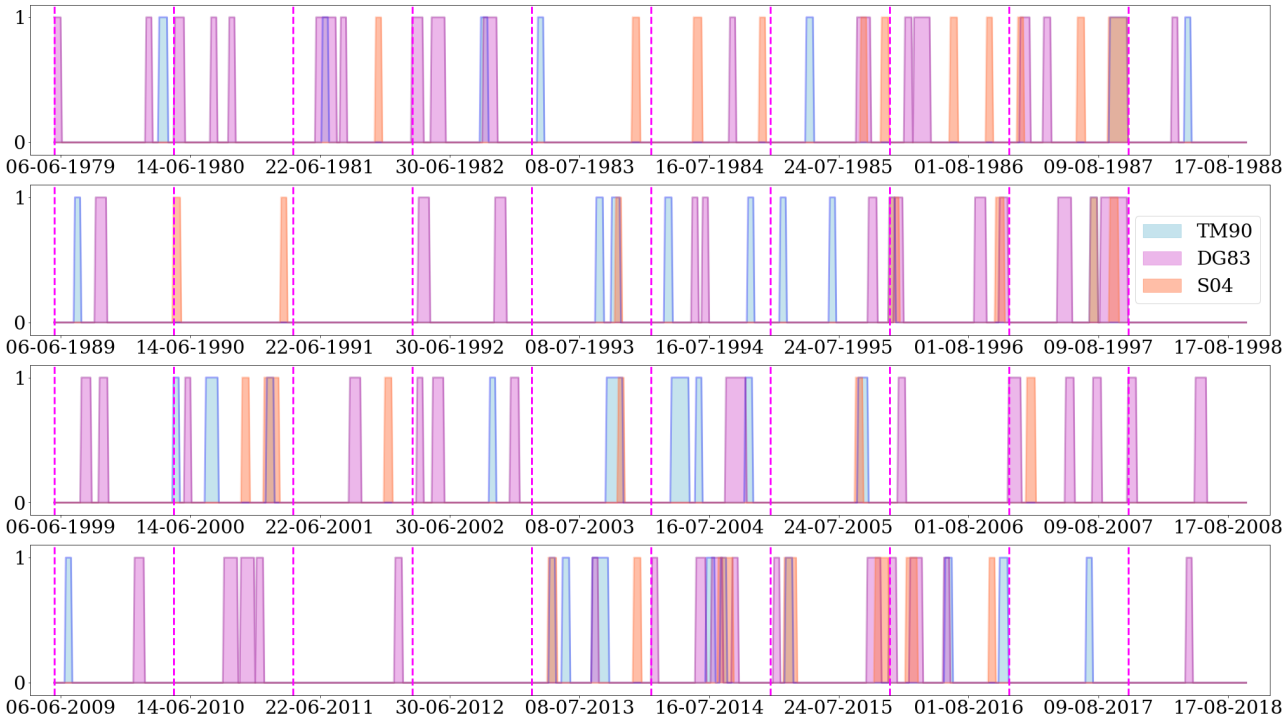

Figure 9: Shows blocked $i = 1$ and non-blocked $i = 0$ JJA Z500* contours as indicated by $\mathcal{X}_{TM90}$ (blue), $\mathcal{X}_{DG83}$ (purple) and $\mathcal{X}_{S04}$ (orange) between 1979 and 2018. The 01/06 for each summer is indicated by the pink vertical line.

d_m, as identified by any one method $m \in \{TM90, DG83, S04\}$ were found to be $d_{TM90} = 235$, $d_{DG83} = 462$ and $d_{S04} = 217$. The probability of any one day in that period being identified as blocked, $P(\mathcal{X}_m) = d_m/3680$, by any method, $\mathcal{X}_m \in \{\mathcal{X}_{TM90}, \mathcal{X}_{DG83}, \mathcal{X}_{S04}\}$ is shown in Table 5. All three methods were found to agree only on 3 days out of 3680. 794 out of the 3680 days were identified as blocked by at least one of the methods. The probability of a blocked day being labeled so by any one method, $P_b(\mathcal{X}_m) = d_m/794$, is shown in Table 5. More than half of the blocked days in the GTD come from the DG83 method (see Section 3.1), which was found to be the least conservative blocking labeling method. Therefore, the labels defined in the GTD are significantly influenced by the DG83 method.

Table 5: Statistic of Individual Methods

d_{TM90}	d_{DG83}	d_{S04}	$P(\mathcal{X}_{TM90})$	$P(\mathcal{X}_{DG83})$	$P(\mathcal{X}_{S04})$	$P_b(\mathcal{X}_{TM90})$	$P_b(\mathcal{X}_{DG83})$	$P_b(\mathcal{X}_{S04})$
235	462	217	0.064	0.126	0.059	0.296	0.582	0.273

The arrays of indices $\mathcal{X}_m$ from different methods, m, were also analysed in pairs and were found to infrequently agree with each other (see overlaps on Figure 9). Out of 794 blocked days, the number of days in which any two methods agree with each other ($d_{m \cap n \neq m}$) were calculated. The method agreement probabilities, $P(\mathcal{X}_m \cap \mathcal{X}_{n \neq m}) = d_{m \cap n \neq m}/794$, were obtained as shown in Table 6. The conditional probabilities, $P(\mathcal{X}_m | \mathcal{X}_{n \neq m})$, of each permutation of methods were also calculated using the full array of 3680 labels from each method. The correlation between the method indices, $C(\mathcal{X}_m, \mathcal{X}_{n \neq m})$, was found using the Pearson Correlation Coefficient and the results are shown in Table 6. None of the method pairs were found to have a significant index

Table 6: Method Pair Analysis

Method Pair	Method Intersection	Conditional Probability	Correlation Coefficient
TM90 and DG83	$P(\mathcal{X}_{TM90} \cap \mathcal{X}_{DG83})$ $= 0.031$	$P(\mathcal{X}_{TM90} \| \mathcal{X}_{DG83}) = 0.054$ $P(\mathcal{X}_{DG83} \| \mathcal{X}_{TM90}) = 0.106$	$C(\mathcal{X}_{TM90}, \mathcal{X}_{DG83})$ $=$ -0.019
DG83 and S04	$P(\mathcal{X}_{DG83} \cap \mathcal{X}_{S04})$ $= 0.049$	$P(\mathcal{X}_{DG83} \| \mathcal{X}_{S04}) = 0.180$ $P(\mathcal{X}_{S04} \| \mathcal{X}_{DG83}) = 0.084$	$C(\mathcal{X}_{DG83}, \mathcal{X}_{S04})$ $= 0.049$
TM90 and S04	$P(\mathcal{X}_{S04} \cap \mathcal{X}_{TM90})$ $= 0.074$	$P(\mathcal{X}_{S04} \| \mathcal{X}_{TM90}) = 0.251$ $P(\mathcal{X}_{TM90} \| \mathcal{X}_{S04}) = 0.272$	$C(\mathcal{X}_{S04}, \mathcal{X}_{TM90})$ $= 0.262$

overlap, with the highest overlap probability being 0.074 for the TM90 and S04 method pair. This pair also had the highest conditional probabilities. The $P(\mathcal{X}_{S04} | \mathcal{X}_{TM90})$ and $P(\mathcal{X}_{TM90} | \mathcal{X}_{S04})$ values were also the most consistent with each other out of all the conditional probability pairs. Furthermore, $C(\mathcal{X}_{S04}, \mathcal{X}_{TM90})$ was found to be highest, albeit still too small to indicate any significant dependency (Table 6). The number of matching indices in the TM90 and S04 method pair, $d_{TM90 \cap S04} = 59$, represented $(59/d_{TM90}) \times 100 = 25\%$ of blocked days in TM90 and $(59/d_{S04}) \times 100 = 27\%$ of the S04 blocked days (Table 5). The methods in the DG83 and TM90 pair agreed the least with only $d_{TM90 \cap DG83} = 25$ matching indices, giving the lowest conditional probabilities and almost no correlation (Table 6).

4.2 CNN Performance Analysis

The GTD was used to train the CNN classifier for the detection of historic blocking using the Z500* contours as the initial input data. The European domain was initially defined over the $75°N - 30°N$ in latitude and $10°W - 40°E$ in longitude. The 3600 countour maps were pre-processed as described in Section 3.2.3 and 3440 7-day samples were used for re-sampling. The up-sampling of the minority class was used to achieve an equal number of samples in each class, which brought the total number of re-sampled observations and labels to 5748. These observations were split chronologically for cross-validated training, 75% of 5748, and testing, 25% of 5748, as described in Section 3.2.2. Thus, the algorithm used the first 30 years of observations and labels for learning and the last 10 years of unseen observations for testing,

see Section 2.3. Testing was performed at the end of every fold and an array of 1437 predicted labels, $x_{k,CNN}$, (algorithm's output) was then compared to the actual labels. Each full run resulted in three F1 scores, as described in Figure 8, and three arrays of predicted labels: $x_{1,CNN}$, $x_{2,CNN}$ and $x_{3,CNN}$. The average of the F1 scores, $\bar{f}_1$, and the average of the three arrays, $\mathcal{X}_{CNN}$, were obtained for each full run of the algorithm. Ten full runs were performed to obtain reliable results as described in Section 3.2.3.

Table 7: Statistics of CNN Predicted Labels

Iteration #	$d_{CNN,tot}$	p_{TM90}	p_{DG83}	p_{S04}	p_{GTD}	$\bar{f}'_1$
1	370	$\frac{36}{62} = 0.581$	$\frac{94}{111} = 0.846$	$\frac{34}{57} = 0.596$	$\frac{114}{139} = 0.820$	0.711
2	280	$\frac{18}{62} = 0.290$	$\frac{74}{111} = 0.667$	$\frac{28}{57} = 0.491$	$\frac{91}{139} = 0.655$	0.762
3	330	$\frac{30}{61} = 0.492$	$\frac{86}{111} = 0.774$	$\frac{32}{56} = 0.571$	$\frac{104}{138} = 0754.$	0.753
4	267	$\frac{23}{62} = 0.371$	$\frac{71}{111} = 0.639$	$\frac{31}{57} = 0.544$	$\frac{94}{139} = 0.676$	0.748
5	860	$\frac{62}{62} = 1$	$\frac{110}{111} = 0.990$	$\frac{56}{56} = 1$	$\frac{138}{138} = 1$	0.578
6	458	$\frac{43}{62} = 0.694$	$\frac{96}{110} = 0.872$	$\frac{48}{57} = 0.842$	$\frac{122}{138} = 0.884$	0.700
7	377	$\frac{35}{62} = 0.565$	$\frac{87}{108} = 0.805$	$\frac{38}{57} = 0.667$	$\frac{112}{136} = 0.824$	0.741
8	362	$\frac{37}{61} = 0.607$	$\frac{89}{111} = 0.801$	$\frac{36}{57} = 0.632$	$\frac{106}{138} = 0.768$	0.739
9	268	$\frac{25}{62} = 0.403$	$\frac{75}{111} = 0.675$	$\frac{27}{57} = 0.474$	$\frac{90}{139} = 0.647$	0.767
10	370	$\frac{30}{60} = 0.500$	$\frac{81}{110} = 0.736$	$\frac{34}{57} = 0.596$	$\frac{105}{137} = 0.766$	0.732
Mean	$\bar{d}_{CNN,tot}$ = 394 ±165	$\bar{p}_{TM90}$ = 0.550 ±0.188	$\bar{p}_{DG83}$ = 0.781 ±0.104	$\bar{p}_{S04}$ = 0.641 ±0.155	$\bar{p}_{GTD}$ = 0.779 ±0.104	$\bar{\mathcal{F}}'_1$ = 0.723 ±0.052

In order to compare the $\mathcal{X}_{CNN}$ to the labels given by the blocking methods ($\mathcal{X}_{TM90}$, $\mathcal{X}_{DG83}$, $\mathcal{X}_{TM90}$) it had to be re-sampled back into the original indices (i.e. one index per original observation in the test set), giving an array of $0.25 \times 3440 = 860$ predicted label, $\mathcal{X}'_{CNN}$. $\mathcal{X}'_{CNN}$ was used to compare to the last 860 indices from $\mathcal{X}_{GTD}$, $\mathcal{X}_{TM90}$, $\mathcal{X}_{DG83}$ and $\mathcal{X}_{TM90}$ (henceforth $\mathcal{X}'_m$, where $m \in \{GTD, TM90, DG83, S04\}$) for ten algorithm runs and the comparison statistics are shown on Table 7. The number of blocked days in $\mathcal{X}'_{CNN}$ that matched blocked indices in $\mathcal{X}'_{GTD}$, $\mathcal{X}'_{TM90}$, $\mathcal{X}'_{DG83}$, $\mathcal{X}'_{S04}$ were found, $d_{CNN \cap GTD}$, $d_{CNN \cap TM90}$, $d_{CNN \cap DG83}$ and $d_{CNN \cap S04}$ respectively. The total number of blocked days in $\mathcal{X}'_{GTD}$, $\mathcal{X}'_{TM90}$, $\mathcal{X}'_{DG83}$, $\mathcal{X}'_{S04}$ were found: $d_{GTD,tot}$,

$d_{TM90,tot}$, $d_{DG83,tot}$ and $d_{S04,tot}$. The method agreement proportion, $p_m = d_{CNN \cap m}/d_{m,tot}$, was calculated to determine how many method specific days were detected as blocked by the algorithm, see Table 7. For example, if there are $d_{DG83,tot} = 111$ DG83-blocked days in the test set, and $d_{CNN \cap DG83} = 94$ of these days were predicted as blocked by the algorithm, the DG83 agreement proportion, p_{DG83}, is $94/111 = 0.846$. The table also shows the total number of blocked days that were predicted by the CNN, $d_{CNN,tot}$, and the new F1 scores that were calculated using $\mathcal{X}'_{CNN}$ and $\mathcal{X}'_{GTD}$, $\bar{f}'_1$. The bottom row on Table 7 gives the averages of the quantities presented in the table columns over the ten full runs. Note that the highest $\bar{p}_m$, shown on the bottom row, was found for the DG83 method and its value was even higher than the $\bar{p}_{GTD}$, albeit only slightly. The average number of total predicted blocked days was found to be $\bar{d}_{CNN,tot} = 394$, which is 2.86, times higher than the expected number of blocked days in the test set as indicated by GTD, taking $\bar{d}_{GTD,tot} = 139$. The over-prediction ratio was found to be 1.93 in the best case (Iteration # 4) and 6.23 in the worst case, where all days in the test set were predicted to be blocked (Iteration # 5). In all Iterations, the number of predicted blocked labels was found to be greater than the expected number of blocked labels, ie the CNN was found to always over-predict, indicating presence of False Negative (FN) labels.

The number of FNs, $d_{FN} = d_{CNN,tot} - d_{CNN \cap GTD}$, i.e. days that were predicted blocked by the CNN that were not actually blocked as defined by the GTD, was found for every iteration of the algorithm (see Table 8). The number of FN labels that matched with method specific blocked days, $d_{FN \cap m}$ was calculated for each method. The proportion of FNs that agreed with the blocked indices as indicated by a specific method m, $p_{FN \cap m} = d_{FN \cap m}/d_{FN}$, was calculated for each method for ten iterations, as shown on Table 8. For example, the number of FNs was found to be $d_{FN} = 256$, of which $d_{FN \cap TM90} = 8$ days were found to be TM90-blocked days. Thus, the proportion of FNs that matched with TM90 is found to be $p_{FN \cap TM90} = 8/256 = 0.0312$. The bottom row on Table 8 shows the average of the column quantity over the ten full runs of the algorithm. The average number of FNs was found to be $\bar{d}_{FN} = 287 \pm 153$. The biggest overlap between false negative indices was found to be with the S04 indices, $\bar{p}_{FN \cap S04} = 0.034 \pm 0.007$, albeit very low. This leads to the conclusion that the vast majority of FNs predicted by the algorithm did not relate to any particular method. The FN indices that did match with the blocked days of individual methods were found to be grouped into, at most, two distinct events rather than coming from a collection of randomly spread individual days (see discussion in Section 5.2).

The ten lots of predicted labels were added and averaged to give averaged predicted labels, $\bar{\mathcal{X}}'_{CNN}$. In $\bar{\mathcal{X}}'_{CNN}$, every label that was above 0.5 was rounded up to $i = 1$ index and the rest were labeled $i = 0$. The FN indices were found by comparing $\bar{\mathcal{X}}'_{CNN}$ to $\mathcal{X}'_{GTD}$ and resulted in total of 224 days. Almost every FN index that did not match with any method specific blocked indices was found to belong to at least one of the 35 groups, henceforth FN events (see Figure 10). Note that if the algorithm's learning was poor, the FNs would be more likely to be randomly spread rather than be grouped together into a collection of days. Out of the 35 FN events, 14 groups were found at either the beginning, middle or end of a GTD-labeled blocking event. The remaining 21 events were split into 6, whose duration was lower than five days, and 15, whose duration ranges between five to fourteen days (see Figure 10). The average FN event duration was found to be 6.06 ± 3.50 days. The true classification of these events is discussed further in Section 5.2.

4.3 Analysis of the Domain Size Dependency

A similar analysis was done to the one mentioned in Section 4.2, with the exception of the domain size of the input contours. The extended European domain was used, defined to spread over $80°N - 25°N$ in latitude and $15°W - 45°E$ in longitude. This resulted in 1070 extra

Table 8: Statistics of False Negatives

Iteration #	d_{FN}	$p_{FN\cap TM90}$	$p_{FN\cap DG83}$	$p_{FN\cap S04}$
1	256	$\frac{8}{256} = 0.031$	$\frac{8}{256} = 0.031$	$\frac{8}{256} = 0.031$
2	189	$\frac{2}{189} = 0.011$	$\frac{2}{189} = 0.011$	$\frac{5}{189} = 0.026$
3	226	$\frac{5}{226} = 0.022$	$\frac{6}{226} = 0.027$	$\frac{7}{226} = 0.031$
4	173	$\frac{1}{173} = 0.006$	$\frac{2}{173} = 0.012$	$\frac{5}{173} = 0.029$
5	722	$\frac{15}{722} = 0.021$	$\frac{18}{722} = 0.023$	$\frac{26}{722} = 0.034$
6	336	$\frac{5}{336} = 0.015$	$\frac{14}{336} = 0.042$	$\frac{17}{336} = 0.051$
7	265	$\frac{3}{265} = 0.011$	$\frac{7}{265} = 0.026$	$\frac{9}{265} = 0.034$
8	256	$\frac{7}{256} = 0.027$	$\frac{10}{256} = 0.039$	$\frac{11}{256} = 0.043$
9	178	$\frac{4}{178} = 0.022$	$\frac{2}{178} = 0.011$	$\frac{6}{178} = 0.034$
10	265	$\frac{2}{265} = 0.008$	$\frac{5}{265} = 0.019$	$\frac{7}{265} = 0.026$
Mean	$\bar{d}_{FN}$ =287 ±153	$\bar{p}_{FN\cap TM90}$ = 0.017 ±0.008	$\bar{p}_{FN\cap DG83}$ = 0.024 ±0.011	$\bar{p}_{FN\cap S04}$ = 0.034 ±0.007

gridpoints to be considered for every Z500* contour map. Ten full runs of the algorithm were performed and each resulted in an array of predicted labels, $\mathcal{X}^{ext}_{CNN}$. These arrays were re-sampled back into the original number of indices in the test set, $\mathcal{X}'^{ext}_{CNN}$ and each was compared to the last 860 labels in the method indices, $\mathcal{X}'_m$ where $m \in \{GTD, TM90, DG83, S04\}$. The same quantities were obtained as the ones shown in Table 7 and Table 8 from this comparison and the averages of these quantities over ten full runs of the algorithm are shown on Table 9. The average of the total number of predicted blocked days, $\bar{d}^{ext}_{CNN,tot} = 329 \pm 70$, is found to be smaller than for the input data with the original domain size shown Table 7. The difference between $\bar{d}_{CNN,tot}$ and $\bar{d}^{ext}_{CNN,tot}$ was found to be within both $1\sigma_{CNN,tot}$ and $1\sigma^{ext}_{CNN,tot}$, where $\sigma_{CNN,tot} > \sigma^{ext}_{CNN,tot}$. Also, the $\bar{p}^{ext}_m$ was found to decrease for every method, m, compared to the $\bar{p}_m$ values shown on the bottom row of Table 7, within $1\sigma_m$ for their respective standard deviation values. The highest $\bar{p}_m$ was found to be for the DG83 method, same as before. The average of the F1 scores, $\bar{\mathcal{F}}'_{1,ext}$ is found to decrease compared to $\bar{\mathcal{F}}'_1$ by 0.296 which is within both $6\sigma_{\bar{\mathcal{F}}'_1}$ and $14\sigma^{ext}_{\bar{\mathcal{F}}'_1}$. The decrease in the average agreement proportions and the decrease in the average F1 score are the consequences of a smaller number of blocked days being predicted.

Table 9: Statistics of CNN Predicted Labels with Extended Domain

$\bar{p}^{ext}_{TM90}$	$\bar{p}^{ext}_{DG83}$	$\bar{p}^{ext}_{S04}$	$\bar{p}^{ext}_{GTD}$	$\bar{\mathcal{F}}'_{1,ext}$
0.478 ± 0.098	0.760 ± 0.104	0.559 ± 0.116	0.722 ± 0.103	0.427 ± 0.022
$\bar{d}^{ext}_{CNN,tot}$	$\bar{d}^{ext}_{FN}$	$\bar{p}^{ext}_{FN\cap TM90}$	$\bar{p}^{ext}_{FN\cap DG83}$	$\bar{p}^{ext}_{FN\cap S04}$
329 ± 70	229 ± 56	0.086 ± 0.038	0.026 ± 0.007	0.034 ± 0.006

Thus, the average number of FNs has also decreased from 287 ± 153 (Table 8) to 229 ± 56 with the extended domain (Table 9). Also, the standard deviation of the number of total predicted blocked days and the number of FNs has decreased twofold and threefold respectively indicating less fluctuation in the algorithm's predictions with the extended domain.

The $\bar{\mathcal{X}}'^{ext}_{CNN}$ was found by averaging the indices in $\mathcal{X}'^{ext}_{CNN}$ across ten iteration. Similarly to $\bar{\mathcal{X}}'_{CNN}$, the indices in $\bar{\mathcal{X}}'^{ext}_{CNN}$ were given a $i = 1$ label if the index value was above 0.5 and $i = 0$ if it was below that value. The predicted FNs in $\bar{\mathcal{X}}'^{ext}_{CNN}$ (216 days in total) were found to be grouped into almost exactly the same events as the FN in $\bar{\mathcal{X}}'_{CNN}$ (total of 224 days). Out of 34 FN events, 11 were found to take place at either the beginning, middle, or end of a GTD-labelled blocking event. 6 events were found to be groups of blocked indices covering less than five days and thus not classifying as a blocking event according to the persistence criteria. The remaining 17 events were found to have durations ranging from five to eighteen days. All groups of FN events detected in the extended domain were found to take place at roughly the same time as the FN events detected in the original domain. The changes in total number of FN indices was found to be due to the changes in group sizes, i.e. the duration of these events. The average duration of the 34 events was 5.88 ± 4.00 days. Another thing to note is that the proportion of FNs that matched with the blocked indices from the TM90 method was found to be $\bar{p}^{ext}_{FN\cap TM90} = 0.086$, which is five times higher than $\bar{p}_{FN\cap TM90}$. This indicates that extending the domain might have caused the algorithm to detect more days that were excluded from the GTD during manual labeling, but were identified as blocked by the TM90 method.

4.4 Variation of Input Data

The same investigation as performed in Sections 4.2 and 4.3 was done to test the ability of the algorithm to detect blocking patterns using the 300 K isentropic PV data. The original domain size was used to obtain ten arrays of predicted labels, which were in turn used to obtain the average quantities as shown in Table 10. The $\bar{d}^{pv}_{CNN,tot} = 253 \pm 134$, and the $\bar{p}^{pv}_{m}$ for every method, m, decreased compared to $\bar{d}_{CNN,tot} = 394 \pm 165$ and $\bar{p}_m$. In some quantities this decrease exceeded three standard deviations (see the bottom row of Table 7 for comparison). The standard deviation of the average number of predicted blocked days and of average number of FNs have both decreased slightly in value, from $\sigma_{FN} = 153$ to $\sigma^{pv}_{FN} = 112$ for the latter. The decrease in the F1 score from $\bar{\mathcal{F}}'_1 = 0.723 \pm 0.052$ to $\bar{\mathcal{F}}'_{1,pv} = 0.187 \pm 0.092$ was within $11\sigma_{\bar{\mathcal{F}}'_1}$ and $6\sigma^{pv}_{\bar{\mathcal{F}}'_1}$. This indicates a greater fluctuation in the algorithm's output and a poorer blocking detection ability.

The arrays were also re-sampled and averaged to give $\bar{\mathcal{X}}'^{pv}_{CNN}$ which was used to investigate the similarity of the FN events between the Z500* observations in Section 4.2 and the IPV observations. The 221 FN indices were found to be grouped into 39 events. Out of these events only 7 matched the beginning, middle or end of a GTD-labelled event. Out of the remaining 32 events, 14 were shorter than five days in duration and 10 matched some of the FN events found in $\bar{\mathcal{X}}'_{CNN}$ and $\bar{\mathcal{X}}'^{ext}_{CNN}$ (described in Sections 4.2 and 4.3). A total of 8 new events were

Table 10: Statistics of CNN Predicted Labels with IPV Data

$\bar{p}^{pv}_{TM90}$	$\bar{p}^{pv}_{DG83}$	$\bar{p}^{pv}_{S04}$	$\bar{p}^{pv}_{GTD}$	$\bar{\mathcal{F}}'_{1,pv}$
0.353 ± 0.183	0.335 ± 0.182	0.413 ± 0.231	0.307 ± 0.164	0.187 ± 0.092
$\bar{d}^{pv}_{CNN,tot}$	$\bar{d}^{pv}_{FN}$	$\bar{p}^{pv}_{FN \cap TM90}$	$\bar{p}^{ext}_{FN \cap DG83}$	$\bar{p}^{pv}_{FN \cap S04}$
253 ± 134	210 ± 112	0.038 ± 0.019	0.015 ± 0.008	0.051 ± 0.027

found. The average duration of the 39 groups was 5.13 ± 3.61 days long, which was the smallest average out of the three with the smallest standard deviation value.

5 Discussion of Results

5.1 Agreement of Blocking Indices

Looking at the statistical analysis of blocking indices in Section 4.1 there are a few things that are worth noting. As can be seen in Table 5, the probability of identifying any one day as blocked is the highest when using the DG83 method, whereas the $P(TM90)$ is similar to $P(S04)$. This is expected as it is noted in Section 2.2 that the DG83 method has a tendency to over-predict blocking. The DG83 and TM90 method pair was found to have the lowest $P(DG83 \cap TM90)$, whereas the TM90 and S04 method pair was found to have the highest $P(S04 \cap TM90)$. This result indicates that the methods that utilise the same variable (i.e. Z500 or PV), but different fields (total vs anomaly) do not detect the same blocking features. The number of days the TM90 and S04 methods agree on was found to represent only 25% of the TM90 indices and 27% of the S04 indices. Albeit small, this agreement between the TM90 and S04 method pair is the biggest out of all three pairs. The pronounced agreement between the TM90 and S04 methods is further confirmed by the highest Pearson correlation coefficient, $C(DG83, S04)$ and the highest conditional probabilities $P(TM90|S04)$ and $P(S04|TM90)$ presented in Table 6. This is an interesting result as the TM90 is known to oversimplify the blocking structure, whereas the S04 tends to appreciate the full 3-D complexity of the phenomena. Changing the TM90's threshold ($\Delta\phi$) did not result in a significant increase in the number of matched blocked days with either the DG83 or S04 methods. Therefore, it is likely that if the TM90-assumptions are changed to better reflect the dimensions of the Z500 field, more blocks detected by the new TM90 will match the S04-blocking events. The TM90 method could be drastically improved by computing the integral of the derivative of the Z500 field along a straight path going from a starting point A to a finishing point B. That way the gradient is calculated by taking an infinite number of infinitesimal steps along the path, thus considering the full 2-D structure of the Z500 field. Moreover, if A and B are also chosen to be the highest and the lowest Z500 value in the domain respectively, it is possible that more blocking events will be found. It is also unexpected that the agreement between the DG83 and S04 method pair was small, despite the DG83 method's ability to satisfy all AMS criteria. This inter-method disagreement indicates that every method majorly focuses on a different type of blocking structure. Thus, even the AMS criteria do not cover the full extent of the variety of blocking structures. It also suggests that combining the labels from each method will result in a more generalised understanding of what constitutes a blocked pattern.

5.2 Algorithm Performance

The predicted labels from ten full runs of the algorithm were analysed in Section 4.2. The performance of the algorithm is judged by the 10-run average F1 score, $\bar{\mathcal{F}}_1'$, as well as the algorithm's ability to detect method-specific blocked days, i.e. looking at the $\bar{p}_{TM90}$, $\bar{p}_{DG83}$ and $\bar{p}_{S04}$ values in Table 7. The highest $\bar{p}_m$ value was found to be that corresponding to the DG83 method. This is an expected outcome since 70.8% of blocked labels in the GTD are the DG83 events, as discussed in Section 3.1. The reproduction of all DG83-blocked events is, however, not expected since our algorithm is an imperfect approximation of the DG83 method. The full DG83 reproduction is also not recommended as this method is known to over-predict blocking. Thus, the resultant $\bar{p}_{DG83} = 0.781$ is considered a good result. The CNN performance by definition falls short of the DG83 method's ability to identify blocking if only $\bar{p}_{DG83}$ is considered. The consideration of the $\bar{p}_{TM90}$ and $\bar{p}_{S04}$ values, however, indicates that the algorithm is also able to successfully detect other blocking patterns. The two values, 0.550 and 0.641 as shown in Section 4.2, despite being lower than the $\bar{p}_{DG83}$, are both larger than 0.5 and are therefore statistically significant. This gives us confidence that our CNN outperforms previously used blocking detection methods and is able to predict more than one type of blocking pattern. It does so by learning various structures from the three blocking detection methods and has the potential to improve if either the method's indices improve, or if more methods are considered for the CNN's training.

It is interesting to analyse the events that the CNN wrongly predicted to be blocked: the false negatives (FNs). Out of 224 FNs, 4 were found to come from the DG83 method, 5 from the TM90 method and 6 from the S04 method. This meant that the majority of FNs predicted by the algorithm did not match any of the method-specific blocked indices, shown in Figure 10. The FNs that matched method-specific blocked days belong to at most two distinct groups per method and the biggest event consist of four S04-blocked days (orange, dated between the 29th of July to the 1st of August 2011 in Figure 10). The biggest $\bar{p}_{FN\cap m}$ was found for $m =$ S04, however, it was too small to hypothesise that FNs agree more with the S04-blocking events (see Table 8). Note that every method-specific event is present at either the beginning or the end of a bigger GTD-labeled blocking event (indicated light-blue on Figure 10). Even though these method specific events were indicated to be blocked by the blocking methods, they were excluded from the GTD after manual labeling as they did not satisfy one or more of the blocking criteria, described in Section 3.1. This leads to the possibility that these 6 events describe the onset or decay stages of the blocking phenomena. Due to the poor understanding of blocking onset and decay processes, it is possible that these days were wrongly classified during the manual labeling. This can also explain some events that are shown in pink on Figure 10, currently referred to as FNs, as 14 out of 35 of these groups were also found to be at the beginning, middle, or end of a GTD-labeled blocking event. Out of the remaining 21 events, 6 groups consisted of less than 5 blocked days, which falls short on the persistence criterion for blocking. If our previous hypothesis - i.e. the CNN being able to detect the onset and decay stages of blocking events - is true, it is possible that these days were days of potential blocking onset that did not end up developing into proper blocking events. The other 15 groups of FN indices were of variable duration, the shortest being five days long to the longest lasting for fourteen days. The true classification of these groups can either be blocked or not blocked. In the scope of this project our definition of blocking does not go beyond the previously used blocking methods, i.e. the TM90, DG83 and S04. Therefore it is not possible to say whether the last 15 events are actually FN events, thus requiring us to make improvements on how our algorithm classifies blocking, or whether these events are the true blocking events that are too complicated to be picked up by any one of the established methods. The latter would

mean that the CNN understood the more general blocking picture and is the superior blocking classification method. In either case, further studies are required into the FN events detected by our CNN to distinguish between the two possibilities. It would also be interesting to see if the changes to the TM90 method computation proposed in Section 5.1 will enable the recognition of some of these FN events as true positives. The Z500* contours of the longest FN event are discussed further in Section 5.3.

Figure 10: The averaged predicted labels, $\bar{\mathcal{X}}'_{CNN}$, decomposed into $\mathcal{X}'_{GTD}$ (blue), $\mathcal{X}'_{FN\cap TM90}$ (cyan), $\mathcal{X}'_{FN\cap DG83}$ (purple), $\mathcal{X}'_{FN\cap S04}$ (orange) and $\mathcal{X}'_{FN}$ (pink), are shown. The event dated just before the 29/07/2014 shows an overlap between a $\mathcal{X}'_{FN\cap TM90}$ and $\mathcal{X}'_{FN\cap DG83}$ event, giving a slightly different colour.

5.3 Dependency of CNN Performance on the Input Data Format

Changing the Domain Size

As described in Section 4.3 the domain size of the Z500* contours was increased. The resulting $\bar{d}^{\,ext}_{CNN,tot}$, $\bar{p}^{\,ext}_{m}$, $\bar{\mathcal{F}}'^{\,ext}_{1,ext}$ and $\bar{d}^{\,ext}_{FN}$ were all found to decrease compared to the results from the original domain. The 34 FN events in $\bar{\mathcal{X}}'^{\,ext}_{CNN}$ were found to match the 34 out of 35 FN events in $\bar{\mathcal{X}}'_{CNN}$. This indicates that the decrease in average quantities was due to the loss of the true positive events rather than the increase of the number of FN events. Evidently, the extended domain has introduced a confusion into the blocking detection process, indicated by the loss of true positives and thus lower F1 scores. This is accounted for by the fact that the GTD-labelling is based off the original domain, whereas the extended domain has an added 31.3% of information. Even if the extra structures in the extended domain are recognised as blocked, they are penalised by the GTD labelling, resulting in a compromised output. In the future, if the training domain is to be expanded, a new GTD has to be created over the new domain to maintain the validity of the results. Regardless of the domain size, the CNN outputs the predicted labels, $\bar{\mathcal{X}}'^{\,ext}_{CNN}$, that have manifestations of the same FN events. This reinforces our hypothesis that the FN events found in both $\bar{\mathcal{X}}'_{CNN}$ and $\bar{\mathcal{X}}'^{\,ext}_{CNN}$ have the potential to be

true blocking structures that were too complicated to be detected by the established blocking methods.

Figure 11: The averaged predicted labels, $\bar{\mathcal{X}}'^{\,ext}_{CNN}$, decomposed into $\mathcal{X}'_{GTD}$ (blue), $\mathcal{X}'^{\,ext}_{FN\cap TM90}$ (cyan), $\mathcal{X}'^{\,ext}_{FN\cap DG83}$ (purple), $\mathcal{X}'^{\,ext}_{FN\cap S04}$ (orange) and $\mathcal{X}'^{\,ext}_{FN}$ (pink), are shown. The event dated just before the 29/07/2014 shows an overlap between a $\mathcal{X}'^{\,ext}_{FN\cap TM90}$ and $\mathcal{X}'^{\,ext}_{FN\cap DG83}$ event, giving a slightly different colour.

The total number of FNs in the $\bar{\mathcal{X}}'^{\,ext}_{CNN}$ was found to only slightly decrease compared to $\bar{\mathcal{X}}'_{CNN}$, from 216 to 224. However, looking at the number of FNs that agreed with one of the methods, they increased from 5,4 and 6 days in $\bar{\mathcal{X}}'_{CNN}$ to 8, 6 and 6 days for the TM90, DG83 and S04 methods respectively. Comparing those events between Figures 10 and 11, we can see that those extra FN were found to be the additional days in the same blocking events, thus increasing their duration (indicated by cyan shading for TM90, purple for DG83 and orange for S04). This could be explained by the ability of the CNN to follow a blocking event for a longer number of days as it propagates slowly through the extended domain. The same events will be present on the extended domain for longer periods of time than on the original domain. However, this directly contradicts the fact that the total number of blocked days (both true positives and FNs) detected has decreased with the increased domain size. It is possible that the true identity of the less significant blocking events in the original domain became more evident in the larger domain and therefore their duration is cut short compared to the original domain. By this same argument, the identity of the more significant events is reinforced by the use of the extended domain, thus, their duration is increased. This hypothesis takes after a well-documented effect of slow blocking pattern migration from west to east with the westerly winds [24]. A stable blocking event moves in quasi-stationary fashion and can remain visible in a bigger domain for longer. The predicted duration of an unstable block is also influenced by the additional structures in the bigger domain, providing more information about the surrounding flow. If the surrounding flow penetrates into the block too much it can be seen to dissipate faster. Regardless of the validity of our claim, further investigation is required into the FN events detected by the CNN.

One of the FN events detected in both the original and the extended domains is shown in

Figure 12. This event was found to be the longest FN event in both $\bar{\mathcal{X}}'_{CNN}$ and $\bar{\mathcal{X}}'^{\,ext}_{CNN}$, lasting 13

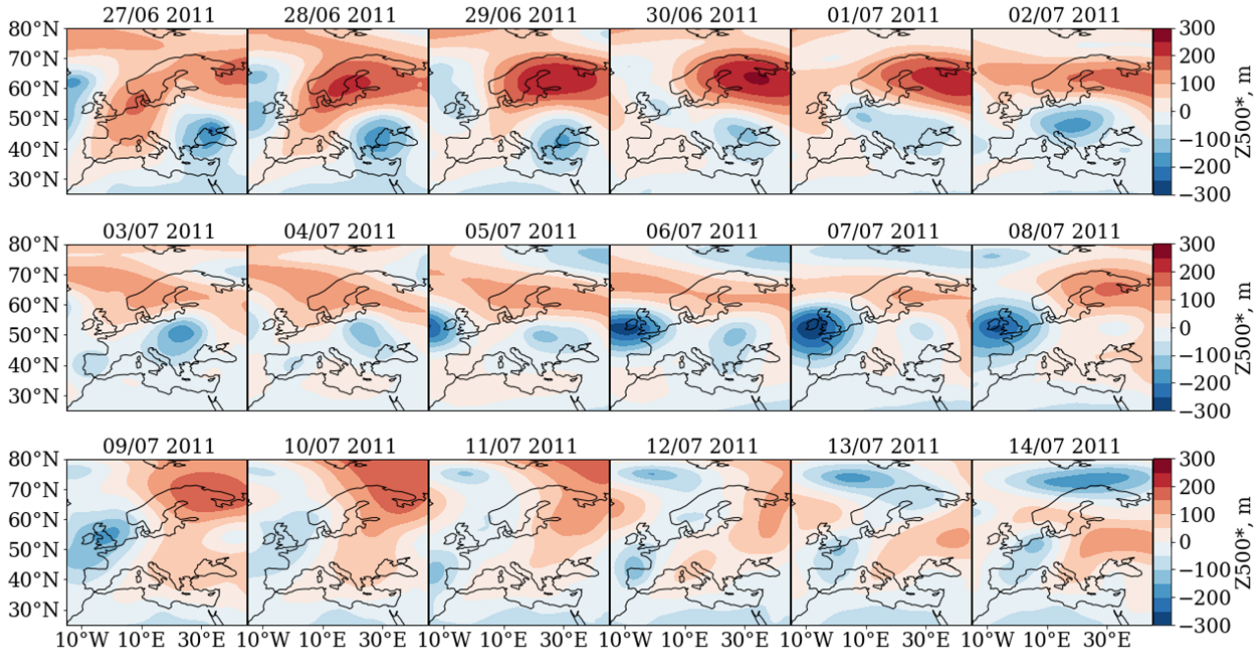

Figure 12: The Z500* contours (colour) of the 27/06-14/07 2011 dated, FN event are plotted over the European coast-line (solid black line). For consistency the range of values was taken to be between $-300 < Z500* < 300$ m for every image.

and 18 days in the respective domains. Its duration has increased with the use of the extended domain, making it an example of a significant blocking event. It is clear by looking at the contours that the persistence and overlap criteria are met, at least for the first six days, which would classify this event as blocked according to the manual labelling protocol (described in Section 3.1). However, none of the blocking methods identified this event as blocked. Figure 14 in Appendix B shows this event after the application of the Z500* threshold, see Section 2.2. This event was abolished after the application of the persistence and contour overlap criteria, which are employed as part of the modifications introduced to the DG83 method [15]. Since all methods depend on some type of threshold that distinguishes between a blocked and a non-blocked pattern, it is evident that this particular event has fallen short of such thresholds. One example of how the TM90 method calculation could be changed to adjust the method's ability to detect blocking is described in Section 5.1. Other modifications can include adjusting the DG83 and S04 threshold values or simply considering more blocking methods for the GTD computation. It is clear, however, that the FN events detected by our CNN certainly present an interesting starting point for further studies of atmospheric blocking and how the established methodologies should be modified.

Using the IPV Maps as Input Data

Against our expectations, the use of 300 K IPV maps did not result in a better performance of the algorithm. As shown in Table 10 in Section 5.3 all average quantities are found to decreased drastically, in many cases by more than a few σ, compared to the average quantities for the Z500* contours. Out of the 39 FN events found in $\bar{\mathcal{X}}'^{\,pv}_{CNN}$, only 10 were found to match the FN events in both $\bar{\mathcal{X}}'_{CNN}$ and $\bar{\mathcal{X}}'^{\,ext}_{CNN}$. The number of events that coincided with the GTD-labelled events has also decreased from 11 to 7 and there were 8 new FN events that did not match

any event detected in $\bar{\mathcal{X}}'_{CNN}$ or $\bar{\mathcal{X}}'^{ext}_{CNN}$. The number of total FN indices in $\bar{\mathcal{X}}'^{pv}_{CNN}$ was found to be 210, which is similar to the 224 and 216 in $\bar{\mathcal{X}}'_{CNN}$ and $\bar{\mathcal{X}}'^{ext}_{CNN}$ respectively. This indicates

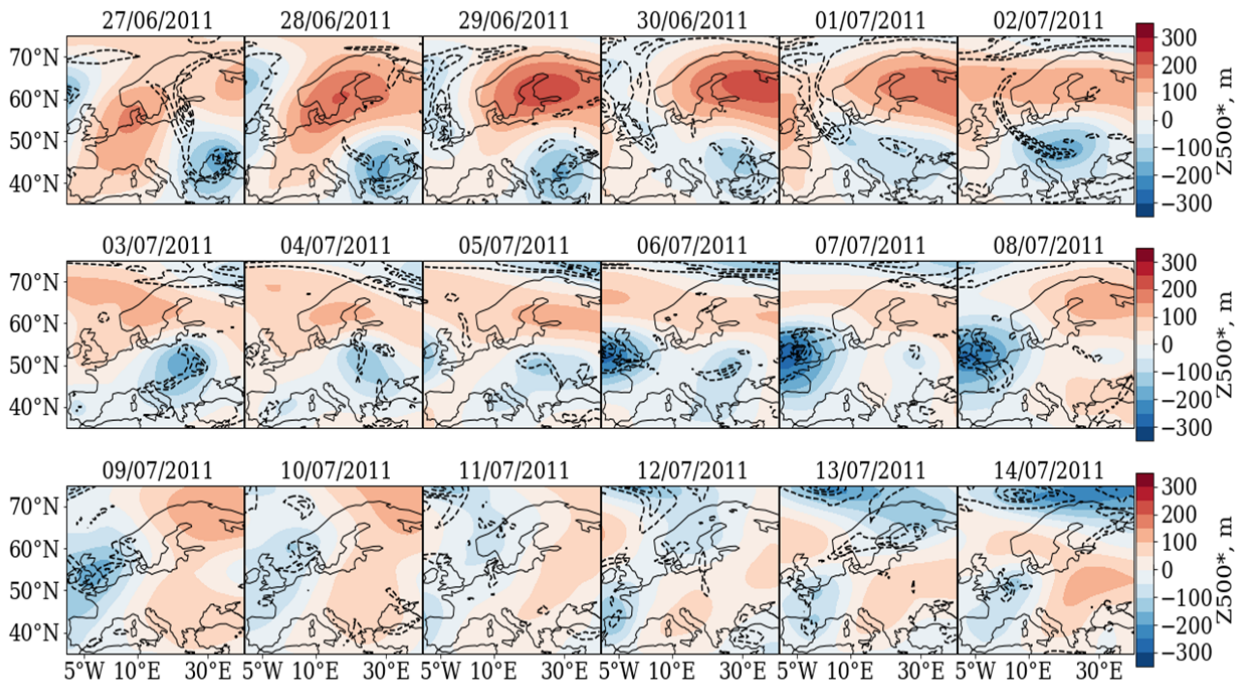

Figure 13: The Z500* contours (colour) overlaid with IPV contours (dashed) for FN event detected between 27/06 and 14/07 2011 are plotted over the European coast-line (solid back). For simplicity only $PV \in \{-1.5, 1.5, 4.5, 7.5, 10.5\}$ PVU contours are shown.

that, similar to the extension of the domain, the use of IPV contours has negatively affected the algorithm's ability to detect true blocking events as defined by the GTD. Not only was there a smaller total number of days detected as blocked by the algorithm, but the detected blocked days were also more randomly distributed. The bigger events were found to split into smaller groups and the average duration of the blocking events decreased. The number of cases where a blocking event consisted of just one blocked day was also found to increase. Since none of the indices are based on the raw IPV data, it is possible that the IPV contours do not carry the structural information that is required for the CNN to detect the TM90, DG83 and S04 specific events. Stating that, the highest $\bar{p}^{pv}_{m}$ was found to be for the S04 method, unlike previously, where it was highest for the DG83 method. The value of $\bar{p}^{pv}_{S04}$, however, was too low to guarantee that the S04 events will always be detected by the algorithm if the IPV contours are used for training. The use of Z500* showed a better detection for each method and for multiple algorithm runs. Remarkably, 10 FN events that were also previously found in $\bar{\mathcal{X}}'_{CNN}$ and $\bar{\mathcal{X}}'^{ext}_{CNN}$ were also detected in $\bar{\mathcal{X}}'^{pv}_{CNN}$. Furthermore, the event shown on Figure 12 was also detected on the IPV maps. This event's IPV contours were plotted on top of the Z500* contours and are shown on Figure 13. The dashed lines are seen to flow around the positive Z500*, indicating the movement of air around the pressure surfaces. It is evident that the IPV maps alone do not provide enough information for the algorithm to detect TM90 and DG83 blocking structures. They have a high noise to signal ratio making it difficult to differentiate between the important features and the less valuable information, even for a human eye. However, IPV maps are known to replicate the blocking features found on Z500* maps [13], as shown on Figure 13. If the IPV maps were utilised in such a way that reduces the noise and still conserves the flow properties, it is possible that IPV maps would be better suited for the CNN's training. The S04 method utilises column-averaged PV over non-isentropic surfaces, thus loosing the information

about the flow properties. We propose that for future studies, both the Z500* and the IPV contours are combined to yield a successful recognition of a broad variety of blocking patterns, an example of which is the event shown on Figure 13.

6 Conclusion

In this project established blocking detection methods were analysed. The blocking indices from these methods were combined to produce a novel GTD of historic blocking events. It included the classification of the pressure patterns for 40 European summers between 1979 and 2018, inclusive. A deep learning classification algorithm, CNN, has been created and trained on the GTD. Two types of input data were used for training: Z500* and IPV maps. The regional domain size was also varied. The performance of the CNN was decided by comparing its outputs with the GTD labels using the F1 metric. The CNN was found to outperform any individual blocking method as it was able to successfully recognise a variety of blocking structures. It was found to identify $(55.0 \pm 18.8)\%$, $(78.1 \pm 10.4)\%$ and $(64.1 \pm 15.5)\%$ of the TM90, DG83 and S04 specific events respectively. Thus, the majority of the predicted blocking events were found to be similar to DG83-events. This was anticipated as this method contributed 70.8% of blocked labels to the GTD. The CNN was able to reproduce $77.9 \pm 10.4\%$ of GTD-labelled events as well as highlight some events that were not identified as blocked by any of the blocking methods, i.e. the FNs. The performance metric for the Z500* input data, $\bar{\mathcal{F}}_1' = 0.723 \pm 0.052$, was found to be 6σ greater than $\bar{\mathcal{F}}_1'$ for the Z500* input with the extended domain and 11σ above the $\bar{\mathcal{F}}_1'$ for the IPV input data. In the extended domain case, the low 10-run average F1 score was caused by the biased GTD-labelling towards the original domain. In the latter case, it was due to the IPV contours having a high noise to signal ratio, thus not providing enough relevant information to the algorithm for blocking classification. In both cases these were instances where the detected blocking events did not match the GTD labelling and in both cases these events were found to match almost fully the FN events detected with the original Z500* input data. The longest FN event, as detected by the CNN on the Z500*, the extended Z500* and the IPV input data, was investigated. It was found to have a persistent positive Z500*, as defined by the DG83 method, as well as appearing to visually meet the persistence and overlap criteria. Even though this event has fallen short of the set thresholds in the established blocking methods, it was clearly significant enough to be detected three times by our classification algorithm.

Some of the FN events were also found to be present at the beginning, middle or end of a GTD-labelled event. Therefore, such events pose a good starting point for a further investigation into the blocking classification methods as well as for future studies of blocking onset, maintenance and decay stages. Regardless of the true classification of those events, the ML tools used for this project have highlighted statistically significant patterns that can be the turning point in the study of blocking patterns. Moreover, using the deep learning model has proven to indicate where the currently adopted methods have failed in describing the blocking phenomena. It is also possible to adapt our model for the prediction of future blocking events from the projected Z500* and IPV contours produced by the climate models.

Appendices

A Hyper-Parameter Tuning for the Learning Rate

Table 11: Hyper-Parameter Tuning for the Learning Rate

Learning rate	$\bar{f}_{1v}$	σ_v	$\bar{f}_{1t}$	σ_t
10^{-1}	0.449	0.317	0.432	0.305
10^{-2}	0.534	0.090	0.531	0.109
10^{-3}	0.604	0.068	0.547	0.123
10^{-4}	0.735	0.039	0.735	0.021
10^{-5}	0.669	0.005	0.663	0.021
10^{-6}	0.593	0.114	0.553	0.134
10^{-7}	0.456	0.307	0.432	0.305
10^{-8}	0.671	0.003	0.653	0.008

B An Example of a Masked False Negative Event Detected by the CNN

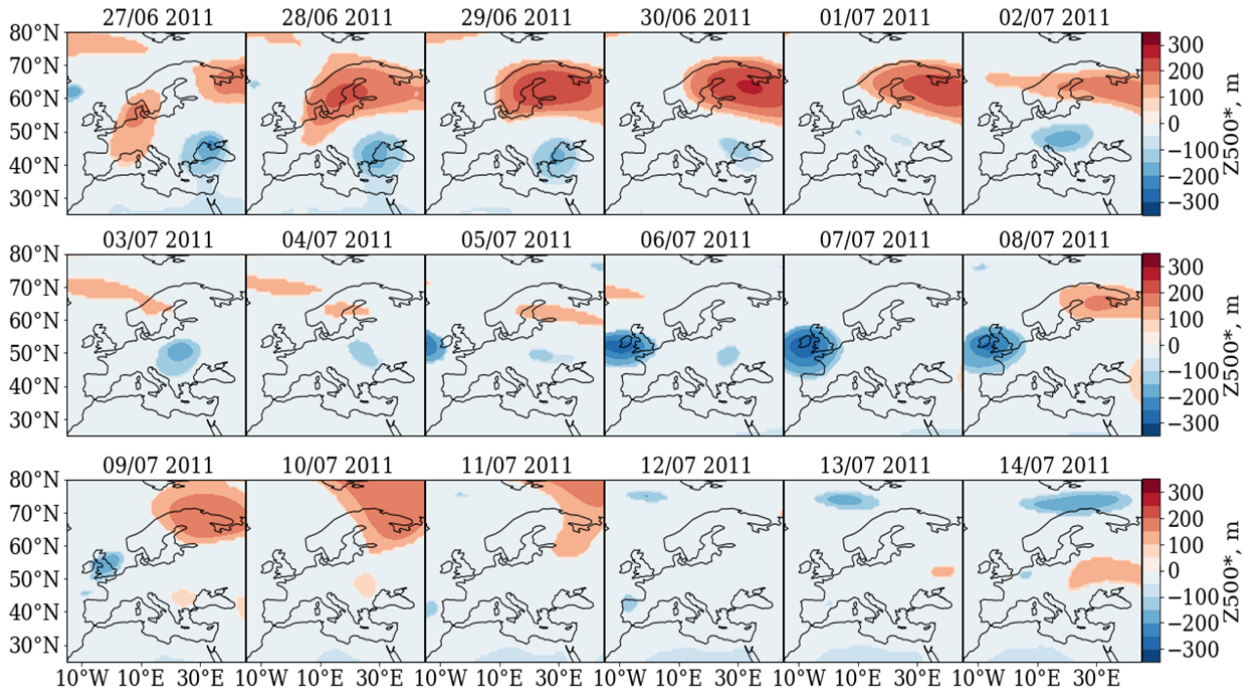

Figure 14: The Z500* contours (colour), that satisfied the anomaly threshold, for the FN event detected between 27/06 and 14/07 2011 are plotted over the European coast-line (solid back).

References

[1] Barnes EA, Slingo J, Woollings T. A methodology for the comparison of blocking climatologies across indices, models and climate scenarios. Climate dynamics. 2012;38(11-12):2467–2481.

[2] Pfahl S, Schwierz C, Croci-Maspoli M, Grams CM, Wernli H. Importance of latent heat release in ascending air streams for atmospheric blocking. Nature Geoscience. 2015;8(8):610–614.

[3] Rex DF. Blocking action in the middle troposphere and its effect upon regional climate. Tellus. 1950;2(4):275–301.

[4] Woollings T, Barriopedro D, Methven J, Son SW, Martius O, Harvey B, et al. Blocking and its response to climate change. Current climate change reports. 2018;4(3):287–300.

[5] Stefanon M, D'Andrea F, Drobinski P. Heatwave classification over Europe and the Mediterranean region. Environmental Research Letters. 2012;7(1):014023.

[6] Hersbach H, Bell W, Berrisford P, Horányi A, J MS, Nicolas J, et al. Global reanalysis: goodbye ERA-Interim, hello ERA5. 2019 04;p. 17–24. Available from: `https://www.ecmwf.int/node/19027`.

[7] Verdecchia M, Visconti G, D'Andrea F, Tibaldi S. A neural network approach for blocking recognition. Geophysical research letters. 1996;23(16):2081–2084.

[8] Glickman TS, Zenk W. Glossary of meteorology. American Meteorological Society; 2000.

[9] Tibaldi S, Molteni F. On the operational predictability of blocking. Tellus A. 1990;42(3):343–365.

[10] Dole RM, Gordon ND. Persistent anomalies of the extratropical Northern Hemisphere wintertime circulation: Geographical distribution and regional persistence characteristics. Monthly Weather Review. 1983;111(8):1567–1586.

[11] Schwierz C, Croci-Maspoli M, Davies H. Perspicacious indicators of atmospheric blocking. Geophysical research letters. 2004;31(6).

[12] Holton JR. An introduction to dynamic meteorology. American Journal of Physics. 1973;41(5):752–754.

[13] Hoskins BJ, McIntyre ME, Robertson AW. On the use and significance of isentropic potential vorticity maps. Quarterly Journal of the Royal Meteorological Society. 1985;111(470):877–946.

[14] Grotjahn R, Zhang R. Synoptic analysis of cold air outbreaks over the California Central Valley. Journal of Climate. 2017;30(23):9417–9433.

[15] Pinheiro MC, Ullrich PA, Grotjahn R. Atmospheric blocking and intercomparison of objective detection methods: flow field characteristics. Climate Dynamics. 2019 Oct;53(7):4189–4216. Available from: `https://doi.org/10.1007/s00382-019-04782-5`.

[16] Kuhn M, Johnson K. Applied predictive modeling. vol. 26. Springer; 2013.

[17] Ghojogh B, Crowley M. The theory behind overfitting, cross validation, regularization, bagging, and boosting: tutorial. arXiv preprint arXiv:190512787. 2019;.

[18] He H, Ma Y. Imbalanced learning: foundations, algorithms, and applications. John Wiley & Sons; 2013.

[19] LeCun Y, Bengio Y, Hinton G. Deep learning. nature. 2015;521(7553):436–444.

[20] Bengio Y, Courville A, Goodfellow IJ. Deep learning: adaptive computation and machine learning. Bengio A Courville. 2016;.

[21] Traore BB, Kamsu-Foguem B, Tangara F. Deep convolution neural network for image recognition. Ecological Informatics. 2018;48:257–268.

[22] Han S, Meng Z, Li Z, O'Reilly J, Cai J, Wang X, et al. Optimizing filter size in convolutional neural networks for facial action unit recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2018. p. 5070–5078.

[23] Arlot S, Celisse A, et al. A survey of cross-validation procedures for model selection. Statistics surveys. 2010;4:40–79.

[24] Treidl R, Birch E, Sajecki P. Blocking action in the Northern Hemisphere: A climatological study. Atmosphere-ocean. 1981;19(1):1–23.