

Multilayered Ensemble Learning for short-term forecasting in agro-climatology

Jade Eva Guisiano
ISEP
Paris, France
jade-guisiano@isep.fr

Julien Orensanz
Cap2020
Gradinian, France
jorensanz@cap2020.fr

Raja Chiky
ISEP
Paris, France
raja.chiky@isep.fr

Shoreh Ahvar
ISEP
Paris, France
shohreh.ahvar@isep.fr

ABSTRACT

Agriculture is one of the areas whose activities depend heavily on weather forecasts. Indeed, in order to optimize their production, farmers must be able to anticipate climate conditions favorable or not to their activities by deploying the appropriate action plans. For this purpose, they consult the data daily from various suppliers of weather forecasts. However, the reliability of the forecasts of each supplier is variable according to the period, the climate or the geographical area. Farmers, therefore, have to arbitrate between suppliers daily. This paper proposes a new set of learning architecture that significantly improves the accuracy of weather short-term forecasts for the next 1-12h in order to assist farmers in decision-making.

CCS CONCEPTS

• **Computing methodologies** → **Ensemble methods.**

KEYWORDS

Ensemble Learning, short-term forecast, agro-climatology

1 INTRODUCTION

The reliability of weather forecasts is necessary for many areas for which activities depend on weather conditions. Agriculture is one of this areas that can be particularly impacted by weather events such as extreme temperatures, wind, storm, rain, etc. These events can cause significant damage to harvests and the result can be the total or partial loss of production. In addition to the damage, certain climatic conditions can also affect cultural operations such as the limited possibility of treatment in windy conditions, but also the difficulty of access to the soil with agricultural machinery in case of rain.

The challenge for farmers is to learn early enough about future climate risks in order to put in place action plans to minimize potential damage. For this, they usually consult the weather forecasts

several times a day. The forecasting providers have limited reliability, thus farmers consult 3, 4, sometimes more, sources of weather forecasts and arbitrate between these sources in a subjective way. Indeed, these sources do not necessarily always perform in the same way. Some providers of weather forecasts will sometimes over-perform by providing forecasts close to the values really observed, and others under-perform with forecasts further away from the values actually observed. The accuracy of the weather forecasts of each supplier may vary according to the period, the type of climate or the geographical area. Given the variability in the reliability of weather forecasts and the multiple prediction providers available to farmers, their decision-making is not facilitated and therefore often remains unclear. One of the possibilities for improving farmer's decision-making would be to provide them with a single source of forecasts that outperforms those they have. The latter should allow farmers to consult only one source of forecasts and thus no longer have to arbitrate between the various providers of usual forecasts. The ideal range of forecasts allowing farmers to prepare for certain climate events is 1 to 12 hours.

In this paper, the goal is to provide more reliable temperature and humidity forecasts than weather forecast providers for the next 1 to 12 hours for 2 agricultural sites a few kilometers apart. First, we establish in the 2 section a state of the art short-term forecasting methods. Secondly, we will present in the section 3 a new set of methods consisting of 4 distinct layers to make predictions. Third, we will detail the context and characteristics of the dataset. And finally, we will present in section 5 the results and we will show that the method is effective for realizing reliable forecasts in the short term.

2 STATE-OF-THE-ART

Meteorology is an interdisciplinary science which relies mainly on the fields of physics, chemistry, and mathematics. The evolution over time of climatic variables such as temperature, wind, pressure, etc. is mainly predicted via the numerical resolution of the Navier-Stokes equations (which describe the movement of gases and liquids) and of mass continuity, but also with the laws of thermodynamics, especially that of perfect gases.

The Numerical Weather Prediction (NWP) [8] is a method that can predict the probabilities of meteorological variables by taking into account the equations and laws previously stated. These are

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ICCTA 2020, April 14–16, 2020, Antalya, Turkey
2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-7749-2/20/04...\$15.00

<https://doi.org/10.1145/3397125.3397130>

integrated into various models representing as closely as possible the atmospheric dynamics, each one is based on different initial conditions or uses different model configurations and/or parameters. These models are then powered by current weather observations and provide forecasts. This forecast set allows to establish forecast probabilities for the coming hours and months. The NWP provides reliable forecasts while estimating their error. However, this method requires a great computational power and this continues to grow with the improvements made to NWP systems.

Nevertheless, other prediction methods that are less demanding in terms of power computation can give relatively reliable forecasts. The Deep Hybrid Model for Weather Prediction [4] uses both a network of deep beliefs (consisting of several layers of Stacked Restricted Boltzmann Machine) for weather forecasting from 6 to 24 hours and also taking into account thermodynamic functions and the dynamic influence of atmospheric laws on meteorological phenomena. There is also some works in the literature that use machine learning techniques for weather forecasting. The study [5] compares the results of various machine learning methods to predict the temperature from 1 to 24 hours in Nashville, Tennessee in USA. The historical weather data of many areas surrounding the city of Nashville and the current meteorological conditions of the considered city are used as inputs for many models such as Regression models, Support Vector, Multilayer Perceptron, Extra-Tree Regression, Random Forest, and finally Ridge Regression. These models prove a certain predictive efficiency by presenting between 1°C and 4°C difference between the predicted temperatures and the observed temperatures.

Other machine learning techniques such as recurrent neural networks [11] and Bayesian networks [2] are used for short-term forecasting of climatic variables. It is also possible to use clustering techniques such as an Enhanced K-nearest neighbor[10]. However, 1 degree difference between the predicted temperatures and the observed temperatures can be decisive for decision making. As a result, our method, via an innovative architecture, proposes to minimize the difference between the predicted temperatures and the observed temperatures.

3 MODEL PRESENTATION

3.1 Global approach

The approach presented in this paper is based on machine learning methods for the short-term forecasting of climate variables. Thus, we avoid the complex and power consuming models that are based on the laws of thermodynamics. In this section, we present the architecture and the overall operations of the Multilayer Ensemble method, then each component of the architecture and their parameters will be detailed. In this section, the architecture and the global operation of the Multilayer Ensemble method will be presented then each component of the architecture and their parameters will be detailed. These techniques are useful for regression and classification problems. However, in this paper we will focus mainly on the regression problem which allows the computation of forecasts.

3.1.1 Introduction to Ensemble Learning. Ensemble Learning consists of training a set of models and then combining them in order to obtain more reliable forecasts than those obtained with a single

model [3]. As a result, each model of the set is called a basic learner or a weak learner. The ensemble of the weak learners improve the results coming from each one in a so-called strong learning model. The Ensemble Learning is broken down into two main sub-parts [12]:

Homogeneous weak learners. consist in using the same learning algorithm for each weak learner of the set. However, each algorithm of the homogeneous learning method takes as input different dataset. Several techniques exist that allow to determine with extract of the dataset to give as input for each learner:

The Bagging technique [1] consists of using several samples of the training data using the bootstrap, i.e a sample with replacement. Each of these samples is used by only one weak learner. The results of the models (independent of each other and executed in parallel) are then combined using a weighted average to create a single result. This technique is used to reduce the instability of models against changes in the learning set and thus reduce the variance of the estimator.

On the other hand, the boosting [9] also offers predictions by performing the weighted average of the results of several models. Unlike Bagging, each model of the set is executed in a sequential way. Each model is an adaptive version of the precedent giving more weight, during the following prediction, to the badly predicted observations. Boosting, aims at reducing the error of prediction, and focuses on the most difficult observations to predict by trying to better model them over the iterations. The Boosting technique helps to reduce bias.

Heterogeneous weak learners. In this approach, we use various learning algorithms based on the same training data. The Stacking technique involving heterogeneous models is as follows (represented in the Figure 1):

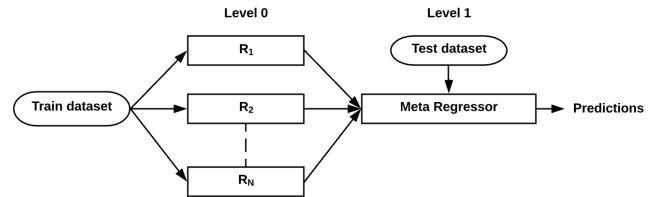


Figure 1: Representation of Stacking.

The stacking aims at combining several regressors by using a set of N different learning algorithms noted L_k where $k = 1, \dots, N$, on the same set of training data S composed of subsets, $S_i = \{X_i, Y_i\}$, with the pairs of exogeneous variables $X = (x_1, \dots, x_W)$ where W represents the number of values, and endogenous variables $Y = (x_{W+1}, \dots, x_{W+h})$ where h is the number of values to predict. As we can see in the figure 1, the stacking consists of two main levels. At level 0, a set R_k of N regressors with $k = 1, \dots, N$, is generated, where $R_i = L_i(S)$. At level 1, a learning algorithm called meta-regressor, is responsible for taking a new set of data formed by the results of N regressors of level 0 and then combining them. In other words, the N regressors in the set R_k generate the forecasts for S_j :

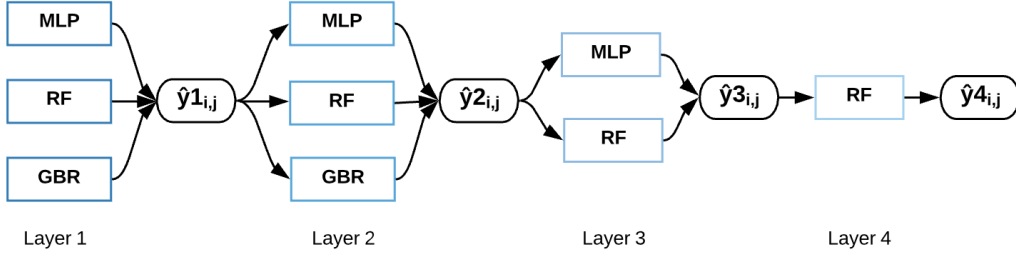


Figure 2: Architecture of the Multilayered Ensemble Learning model.

$$R_k^i(x_i) = \hat{y}_{i,j}^k \quad (1)$$

All the predictions are then combined by the f_j function induced by a learning algorithm called meta-regressor. The regressor $R_{k,j}$ where $j = 1, \dots, h$ takes as input X to predict \hat{y}_{W+h} which is obtained by the function:

$$\hat{y}_{W+h} = f_j(R_{1,j}, \dots, R_{N,j}) \quad (2)$$

In practice, the f_j function of the meta-regressor is a model induced by a learning algorithm. Regarding the dataset, a subset of S_i data where $S_i \in S$ is kept as test data for the Level 1, so the input training data of N regressors of level 0, $\forall i = 1, \dots, n$ and $k = 1, \dots, N$, correspond to:

$$R_{k,i} = L_k(S - S_i) \quad (3)$$

3.1.2 Multilayered Ensemble Learning. Our model, represented by the figure 2, is mainly inspired by the stacking technique. Indeed, each layer of the model is composed of a set of heterogeneous weak learners R_k , where $k = 1, \dots, 3$. We find in each layer a L_k set of different supervised learning algorithms such as Multilayer Perceptron (MLP), Random Forest (RF) and Gradient Boosting Regressor (GBR). These different algorithms allow the use of regressions and the purpose is to compute predictions as indicated by the equation (1).

Definition of the input data format. For a dataset S , a sub-sample S_i is kept as a test data, so each algorithm of each layer takes as input the complementary¹ S_i^c representing the training data. The data set S is organized as a matrix $A = (a_{i,j})_{1 \leq i \leq m, 1 \leq j \leq n}$ of size $(n \times m)$ where the sets I and J are finite and are respectively sets of integers $\{1, \dots, m\}$ and $\{1, \dots, n\}$. Matrix A is composed of exogenous variables $X_{1 \leq i \leq m, 1 \leq j \leq n-1}$ and an endogenous variable $Y_{1 \leq i \leq m, j=n}$.

Layer 1. The three² supervised learning algorithms of the Layer 1 take as input the same training data S_i^c , and each one learns in parallel a regression function based on the exogenous variables $X_{1 \leq i \leq m, 1 \leq j \leq n-1}$ to correctly predict the endogenous variable $Y_{1 \leq i \leq m, j=n}$. In order to test the prediction efficiency of the learned

regression functions, the algorithms will take as input the exogenous variables of the test data S_i and apply the learned regression functions. Each of these three regressors will produce \hat{y}_i forecasts and these will be compared to the endogenous variables of the S_i test data, thus allowing the performance of each algorithm to be reported. As shown in the equation (1), the predictions from the set of R_k regressors are denoted $\hat{y}_{i,j}^k$ where in our case $k = 1, \dots, 3$ and form the matrix $\hat{y}_{1,j}$ where $j = k$.

Layers 2 to 4. Unlike the layer 1, the algorithms of the following layers are no longer trained on the S_i^c training data from the initial S_i dataset. For example, the algorithms of layer 2 are trained on the matrix $\hat{y}_{1,j}$ which represents the exogenous variables and are given as input to each algorithm of the layer 2. However, the endogenous variables from the test data S_j of the layer 1 are kept for the training phase of the algorithms of the layer 2, but also for the test phase. Thus, from the layer 2, each algorithm of each layer takes as input the $\hat{y}_{k,i,j}$ forecast matrix from the previous layer. These algorithms also retain the endogenous variable $Y_{1 \leq i \leq m, j=n}$ of the test data, used in the previous layer, for their training and testing phases.

The output data format. Knowing that the fourth layer has a single $k = 1$ algorithm and that $j = k$ then the final $\hat{y}_{4,i,j}$ forecast from the fourth layer is a column vector. Our method was designed to obtain the most reliable forecasts possible in the short term, which is why it is inspired by the operation of stacking, which is known to improve forecasts. The main interest of Multilayer set Learning is the gradual improvement of forecast accuracy as layers progress. Indeed, at the end of each layer, the difference between the forecasts and the values actually observed tends to decrease. For this, we selected 3 methods with different benefits for improving forecasts.

3.2 Learning methods

The weak learners that we can use in the set approach for predictions are varied [7], we can cite, for example, Support Vectors regression models, Random Forests, and Neural Networks. Our multi-layer set method uses three different weak learners: the Multilayer Perceptron (MLP), the Random Forests (RF) and the Gradient Boosting Regressor (GBR). These three algorithms have as main advantage to not being sensitive to scale changes (there is no need to normalize data and therefore the risk of losing information) and they do not require any particular statistical hypotheses. We chose these three algorithms because they present good accuracy

¹ Given a reference set S , the complement of the subset S_i of S (implied in relation to S) is the set of elements of S that do not belong to S_i . It's noted $\bar{S}_i : S_i^c = \{x \in S \mid x \notin S_i\}$.

² The number of algorithms for each layer and the number of layers itself are relative to our application data (see section 3.2) and can vary for a different application

for predictions. We also compared them against other algorithms such as Support Vector Regression (SVR) and Long Short-Term Memory (LSTM) and we did not retain SVR and LSTM because of their poor prediction performance in our context.

Multilayer Perceptron (MLP). is part of the family of neural networks, this algorithm is able to provide accurate predictions if its hyperparameters are correctly adjusted according to the data. Different values for the hyperparameters were tested, those allowing the neural networks to provide optimal predictions were retained: 3 layers of respectively 40,15,1 neurons, Regularization of weights and bias (0.03), Iterations (7000), Mini-Batch (10), Linear activation functions, Algorithm optimization (Adagrad);

Random Forest (RF). this algorithm is a special case of bagging, indeed, it trains each model independently and then averages the forecasts of all models. It has the advantage of reducing the variance and thus reducing the sensitivity to small fluctuations in the learning sample. In our case, we use the default hyperparameter (but which is also optimal) where the number of trees is 10.

Gradient Boosting Regressor (GBR). is similar to Boosting. It builds models one by one, each new model helps to correct errors made by previous models. This principle has the advantage of reducing the bias and therefore the forecast error. We use the GridSearchCV³ automatic search algorithm from the Scikit-Learn library to select hyperparameters such as the pseudo-regularization of the loss function and the maximum depth of the trees.

4 EXPERIMENTAL STUDY

4.1 Presentation of the dataset

Our dataset represent the period from March 1st 2019 until August 2019 and focuses on a small area located in France, where we have 2 weather stations distant several kilometers one from the other. These 2 stations collect 1 to 4 per hour of really observed data about humidity, precipitation, and temperature. We also have data from several weather forecast providers that have forecasts for each station ranging from 1 to 3 hours depending on the supplier. Forecasts include cloud cover, humidity, pressure, temperature, wind intensity direction. The whole of this dataset presents some constraints, firstly, the meteorological stations can present failures, also the dataset which they provide is more or less complete according to the considered station. Secondly, it is also possible that some providers do not have one or more predictor variables. Third, the period covered by the data represents only 3 seasons of the current year, so the volume of the dataset is relatively small. After cleaning the dataset, the experimental framework was defined as follows. X is a matrix composed of the following exogenous variables : cloud cover (%), wind direction (degrees), humidity (%), pressure (hPa), temperature (°C) and wind (m/s) Concerning the vector column Y , it is composed of the endogenous variable humidity(%) or temperature (°C). For each of the 2 stations (Station_1 and Station_2) 3 meteorological providers are selected (Api-Agro⁴, Weatherbit⁵, and

Dark Sky⁶) all having a forecast step equal to 1 hour are retained. The X matrix is only composed of supplier predictor variables and the Y column vector consists of observations of temperature or humidity depending on what we are trying to predict. Regarding the volume of the data, if we take the example of the Station_1, the X matrix at the input of the first layer is of size (2131×15) and (2131×1) for Y . Each line of X and Y is associated with a date as an index with a granularity hour by hour. The input training data selection percentage of the layers 1 to 3 of the model is 75% and the remaining 25% represents the test data. As for the last layer, the training data selection percentage is 80% and 20% for the test data.

4.2 Objectives and Metrics Evaluation

In this study, we focus primarily on improving the temperature and humidity forecasts for each of the 2 stations. The purpose of the study will be to provide more accurate forecasts than those of public weather providers. To determine the predictive accuracy of the providers for each station, it is sufficient to compare the value of their forecasts with the values of the observations of the studied station. Since our dataset has about 4 months of observations in the past, we can use these observations to evaluate a provider's performance across the last 3 seasons. To calculate the difference between forecasts and observations, we use the Root Mean Square Error measure:

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (\hat{y}_i - y_i)^2}{n}} \quad (4)$$

The equation (4) allows to obtain the differences between the n forecasts \hat{y}_i and the n observations y_i that have the same unit as the prediction and observation values. The 3 providers selected in our experimental framework are those with the smallest quadratic difference and therefore those with forecasts closer to the observed values. Each of these providers uses a numerical model that simulates the evolution of the atmosphere. The model divides the area of interest into a set of grids of three dimensions, using meshes of a few kilometers of side. It is therefore for each of these meshes that the forecasts are established. However, it is possible that between two points distant from a hundred meters only climatic data such as temperature and humidity are significantly different. These local variations are phenomena called sub-mesh, for which providers aim, through the use of complex algorithms, to predict local phenomena. The observed data for each station that we have in our dataset makes it possible to take local phenomena into account. Our model will learn to predict the temperature and humidity at the exact location of each station in order to eliminate the problem of generalizing forecasts over several kilometers and thus provide more reliable forecasts in the short term.

³https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html

⁴<https://api-agro.eu/>

⁵<https://www.weatherbit.io/api>

⁶<https://darksky.net/dev>

5 RESULTS AND DISCUSSION

Table 1: RMSE Comparison between the 4 layer’s algorithms for temperature forecast on Station_1.

	Layer 1	Layer 2	Layer 3	Layer 4
MLP	1.16	1.19	1.84	1.45
RF	1.13	1.08	1.45	0.29
GBR	1.11	1.13	2.07	0.42

We base our experimental study on the one from [6] about the impact of an Extra Layer on the Stacking Algorithm for classification which affirms that an extra layer improves classification task. The table 1 present the RMSE (Root mean Square Error) obtained for each algorithm of each layer of our prediction method. We can see in our case that a real improvement takes place in the final layer. In the third layer, we decide to delete GBR forecasts because they present a too high RMSE value. In the final layer, we decide to only conserve RF forecasts because it presents the lowest RMSE value.

Table 2: RMSE Comparison between forecasters and Multi-layered Ensemblist Learning Model.

	Api-Agro	Dark Sky	Weatherbit	Forecast
Temperature Station_1	0.80	0.73	1.32	0.29
Temperature Station_2	1.19	1.80	2.13	0.79
Humidity Station_1	9.63	4.39	5.28	4.02
Humidity Station_2	4.98	4.74	7.12	4.12

The table 2 shows the RMSE of the forecasts (temperature and humidity) from 28 August 12 hours up to 28 August 18 hours from the 3 suppliers (Api-Agro, Dark Sky and Weatherbit) and predictions made with our model for Station_1 and Station_2. The RMSE values for temperatures are interpreted in degrees and in percentage for humidity. For better representativeness of the results, we run 10 times all the algorithms of our method and averaged the forecasts for each hour. The RMSE is used here to compute the average difference between predicted values and observed values over the 12 to 18 hours time range. For example, for the Station_1, the temperatures predicted by our algorithm (Forecast), the 28 August between 12 and 18 hours differ by an average of 0.50 degrees from the actual temperatures observed. Regarding the humidity, for Station_2, according to the table 2, the forecasts from our model differ on average from 4, 02 of the humidity actually observed.

Table 3: Comparison between our model’s forecast and singles algorithms.

	MLP	RF	GBR	MLEL
Temperature Station_1	0.80	1.04	0.93	0.29
Temperature Station_2	1.51	0.97	1.09	0.79
Humidity Station_1	4.79	6.88	6.03	4.02
Humidity Station_2	8.43	7.53	8.51	4.12

We have also compared our method’s RMSE with RMSE obtained with the use of single algorithms in table 3. This comparison permits to validate the hypothesis that a stacking model gives better forecasts than a single algorithm in our application case. We can see that our model is able to produce better forecast for temperature and humidity independently of the site.

6 CONCLUSION AND FUTURE WORK

The Multilayer Ensemble Learning method presented in this paper is capable of providing hour-by-hour temperature and humidity forecasts, which are more accurate than those of our best meteorological providers. And on average, over a total range of 7 hours, our method has a better forecast accuracy than the 3 providers for both studied stations. Improved forecasting can thus help farmer’s decision-making. Predictions from our model will be used to predict the occurrence of various crop diseases, but also to predict the occurrence of extreme events such as freezing or heavy rains. One possible improvement of our approach is to make it generic and applicable to a broad range of applications. Indeed, the choice of our model and more precisely the number of layers, algorithms and finally their hyperparameters depend on our data. The number of layers depends, for example, on the number of rows in our dataset, but also on the percentage of training and test data chosen within each layer. The objective would be to implement a method to optimize forecasts by automatically selecting optimal parameters such as training and test percentages, number of layers, algorithms and their hyperparameters.

7 ACKNOWLEDGMENTS

This work has been supported by FEDER-IDEAC project, and funded by "Conseil Regional de l’Ile de France".

REFERENCES

- [1] Leo Breiman. 1996. Bagging Predictors. *Mach. Learn.* 24, 2 (Aug. 1996), 123–140. <https://doi.org/10.1023/A:1018054314350>
- [2] Antonio S. Cofino, R. Cano, C. Sordo, and José Manuel Gutiérrez. 2002. Bayesian Networks for Probabilistic Weather Prediction. In *ECAI*.
- [3] Saso Džeroski and Bernard Ženko. 2004. Is Combining Classifiers with Stacking Better than Selecting the Best One? *Machine Learning* 54, 3 (01 Mar 2004), 255–273. <https://doi.org/10.1023/B:MACH.0000015881.36452.6e>
- [4] Aditya Grover, Ashish Kapoor, and Eric Horvitz. 2015. A Deep Hybrid Model for Weather Forecasting. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD ’15)*. ACM, New York, NY, USA, 379–386. <https://doi.org/10.1145/2783258.2783275>
- [5] A H M Jakaria, Md Mosharaf Hossain, and Mohammad Rahman. 2018. Smart Weather Forecasting Using Machine Learning: A Case Study in Tennessee.
- [6] Rodolfo Lorbieski and Silvia Nassar. 2018. Impact of an Extra Layer on the Stacking Algorithm for Classification Problems. *Journal of Computer Science* 14 (05 2018), 613–622. <https://doi.org/10.3844/jcssp.2018.613.622>
- [7] Zhiyuan Ma, Ping Wang, Zehui Gao, Ruobing Wang, and Koroush Khalighi. 2018. Ensemble of machine learning algorithms using the stacked generalization approach to estimate the warfarin dose. In *PloS one*.
- [8] A. Thorpe P. Bauer and G. Brunet. 2015. The quiet revolution of numerical weather prediction. *Nature* 525 (2015), 47–55. <https://doi.org/10.1038/nature14956>
- [9] Robert E. Schapire and Yoav Freund. 2012. *Boosting: Foundations and Algorithms*. The MIT Press.
- [10] Mohammed Sharif and Donald Burn. 2007. Improved K -Nearest Neighbor Weather Generating Model. *Journal of Hydrologic Engineering - J HYDROL ENG* 12 (01 2007). [https://doi.org/10.1061/\(ASCE\)1084-0699\(2007\)12:1\(42\)](https://doi.org/10.1061/(ASCE)1084-0699(2007)12:1(42))
- [11] Mohamed Akram Zaytar and Chaker El Amrani. 2016. Sequence to Sequence Weather Forecasting with Long Short-Term Memory Recurrent Neural Networks.
- [12] Zhi-Hua Zhou. 2012. *Ensemble Methods: Foundations and Algorithms* (1st ed.). Chapman & Hall/CRC.