energies

MDPI

Article

Power Optimization for Wind Turbines Based on Stacking Model and Pitch Angle Adjustment

Zhikun Luo [1,*], Zhifeng Sun [1], Fengli Ma [1], Yihan Qin [1] and Shihao Ma [2]

[1] College of Electrical Engineering, Zhejiang University, Hangzhou 310027, China; eeszf@zju.edu.cn (Z.S.); fenglima190837@163.com (F.M.); qyh_1997@163.com (Y.Q.)

[2] Wuzhong Baita Wind Power Corporation Limited, Wuzhong 751100, China; ma_sh@ecidi.com

* Correspondence: eelzk@zju.edu.cn

Received: 6 July 2020; Accepted: 7 August 2020; Published: 12 August 2020

check for updates

Abstract: As we know, power optimization for wind turbines has great significance in the area of wind power generation, which means to make use of wind resources more efficiently. Especially nowadays, wind power generation has become more and more important. Generally speaking, many parameters could be optimized to enhance power output, including blade pitch angle, which is usually ignored. In this article, a stacking model composed of Random Forest (RF), Gradient Boosting Decision Tree (GBDT), Extreme Gradient Boosting (XGBOOST) and Light Gradient Boosting Machine (LGBM) is trained based on historical data exported from the Supervisory Control and Data Acquisition (SCADA) system for output power prediction. Then, we carry out power optimization through pitch angle adjustment based on the obtained prediction model. Our research results indicate that power output could be enhanced by adjusting pitch angle appropriately.

Keywords: power optimization; wind turbines; pitch angle adjustment; stacking

1. Introduction

As a clean and renewable energy, wind energy has been widely used by humankind in many areas, including electricity generation [1,2]. According to global wind report 2019 by the Global Wind Energy Council (GWEC) [3], 60.4 GW of wind energy capacity was installed in 2019, a 19 percent increase from installations in 2018 and the total capacity for wind energy globally is now over 651 GW, an increase of 10 percent compared to 2018. These statistical results imply that wind power generation is becoming more and more important.

To make full use of wind resources, power optimization for wind turbines is necessary. This has become a hot topic recently. To date, many researchers have contributed a lot to this area. For example, in [4], large-eddy simulations with extremum-seeking control for individual wind turbine were performed for power optimization. Their study was focused on how rated wind speeds with turbines are controlled by generator torque gain. In [5], the authors proposed a control framework to maximize wind turbine power generation through dynamic optimization of drive train gear ratio. Furthermore, a kind of lazy, greedy algorithm for power optimization of wind turbine positioning on complex terrain was put forward in [6]. Other papers, like [7–12], also presented many optimization strategies to enhance power output of wind turbines.

However, we notice that these methods have not involved any Machine Learning (ML) technology to date. In fact, ML has been applied in many areas like Image Recognition, Natural Language Processing (NLP) and Data Mining [13]. It could also be used for our topic. One major thing we can do with ML technology is to obtain a power prediction model. This is important to do before we can carry out optimization. In fact, many papers related to power prediction for wind turbines have been published, like [14], where researchers created a turbine regression tree model to predict power output. In [15], authors used deep neural networks with discrete target classes to do probabilistic short-term wind power forecasts. Besides, BP and RBF neural networks were applied to realize wind power short-term prediction in [16]. These contributions are of great significance. Motivated by their approaches, in this article, we chose a stacking method for prediction modeling. The stacking method is one kind of Ensemble Learning. The main idea of stacking is to obtain a strong learner by combining many weak learners together. We will introduce this in detail later.

Once we get the prediction model, we get the objective function for optimization problem so that the power optimization process could be started. In our work, blade pitch angle is the main parameter to be optimized. A completed algorithm flow showing how to optimize pitch angle will be presented in this article.

The rest of this paper is organized as follows: In Section 2, a brief introduction to three typical types of abnormal data in original datasets is given to indicate the necessity of data preprocessing. Section 3 shows the entire structure of our optimization strategy. In Section 4, we present the power prediction modeling process. The optimization process is shown in Section 5. We conclude our work in Section 6.

2. Background of Abnormal Data

As we know, a wind turbine system is complicated, composed of blades, drive system, yaw system, hydraulic system, braking system, control system, etc. Therefore, power output could be affected by many factors like wind speed, yawing error, temperature inside and outside cabin, pitch angle, etc. Among these factors, wind speed is the main factor. We used to use the wind power curve as a reference to describe general power output under different wind speeds. Figure 1 shows a wind power scatter diagram based on some original data. Three common types of abnormal data are shown in this figure as well.

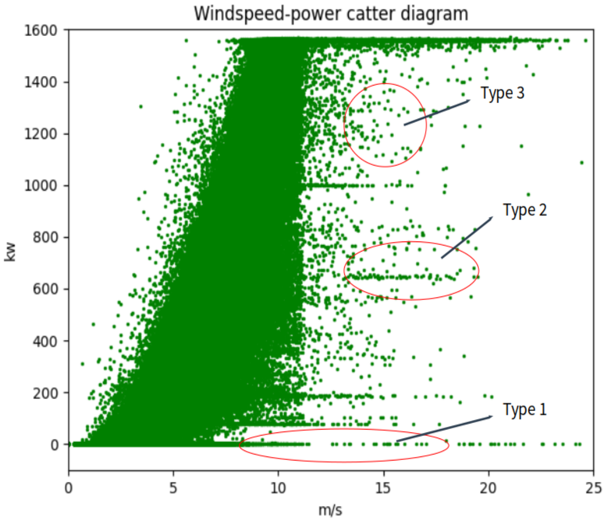


Figure 1. Three common types of abnormal data in wind power scatter diagram.

Obviously, we can recognize a general wind power curve from this scatter diagram. This is useful for us to identify those abnormal data:

- The lower stacked abnormal data which are typically recognized no matter the wind speed, with a power output of zero, are as shown in Type 1. Some reasons for such abnormal data include damage to power measuring instrument, wind speed measuring instrument, abnormal communication equipment or failure of wind turbine, etc.;

- The wind curtailment data are a horizontally dense data cluster, as shown in Type 2. Such abnormal data show the output power of wind turbine does not change as wind speed changes. One main reason is the forced wind abandonment of wind farm;
- The decentralized abnormal data are randomly distributed around wind power curve, as shown in Type 3. Such a distribution of these abnormal data is irregular, and usually caused by a decline in sensor accuracy, instrument failure or signal propagation noise, etc.

These abnormal data should be eliminated since they could not represent a normal working situation of wind turbine. A model trained under these data will face a great prediction error. Due to these reasons, a data preprocessing process is very necessary.

3. Framework of Power Optimization Approach

Note that in our work, output power is only target value to be optimized. To solve this optimization problem, we need the objective function first. Our objective function should be able to describe mapping between features and output power. In other words, it is a power prediction model which predicts power output based on the features given. Therefore, our approach would be divided into two parts. In the first part, we need to build power prediction model based on historical data. In the second part, we use this model for optimization. To make our method reasonable, prediction modeling and optimization will be performed on each wind turbine individually. Figure 2 shows a brief overview of the entire framework.

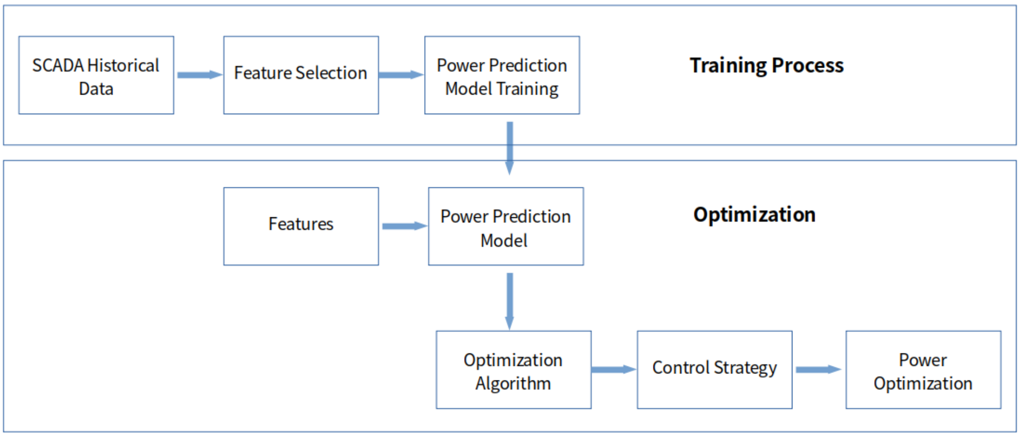


Figure 2. Overview of entire framework.

3.1. Data Preprocessing

As discussed in Section 2, a data preprocessing process which eliminates three typical types of abnormal data is needed first to ensure that our model can learn better. We notice that historical data contain a “state” field which represents the current working state of wind turbine. This is useful information. We first eliminate those data with abnormal “state” field. Those data with power output equal or below 0 are also removed, because it is meaningless to perform prediction or optimization of these data.

For the rest of the abnormal data, we use the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) method to remove them. DBSCAN is a typical clustering algorithm based on density of data. Different from partition method and hierarchical clustering, it defines clusters as the largest set of points connected by density which can divide regions with high enough density into clusters and find clusters of any shape in noise space [17].

It is practicable and reasonable to apply a DBSCAN clustering algorithm. As shown in Figure 1, normal data generally gather around, which forms a region with shape like a wind power curve. This region has high density. On the contrary, abnormal data are randomly distributed around the wind power curve, which forms some regions with low density. Therefore, the DBSCAN algorithm could be able to eliminate these abnormal data.

3.2. Feature Selection

The main purpose of feature selection is to reduce dimensions of feature vectors to avoid over-fitting, improve model generalization ability and make the model learn faster. There are three main types of feature selection algorithms: Filter, Wrapper and Embedding. The filter method scores each feature according to divergence or correlation, sets a threshold, then selects them. The wrapper method circularly chooses several features or excludes them referring to the objective function until one best subset is selected. The embedding method uses original data to train a learning model and obtain the weight coefficients of each feature, then selects features from large to small according to coefficients [18]. In our research, all of them will be tried once. We make a comprehensive consideration after that to obtain the final feature selection results.

3.3. Stacking Model

In the Machine Learning area, Ensemble Learning has been popular in recent years. To date, there have been three main forms of ensemble learning method: Bagging, Boosting and Stacking. Bagging was first proposed by Leo Bierman in 1996. One classical bagging model is known as RF. Boosting was informed by Michael Kearns. There are many popular boosting models like GBDT, XGBOOST or LGBT. Stacking was recently proposed as one popular ensemble learning method which has been proved to be efficient and powerful in Kaggle competition, sentiment classification and many other areas. In recent years, many researchers chose the stacking method to finish their prediction work, like [19–22]. Bagging is a good way to reduce variance in the training process, since it uses repeated sampling to ensure that each subnet of bagging can cover training sample space well. On the other hand, Boosting mainly reduces the bias of the training process through iterative learning. As for the Stacking method, due to its specific structure which combines different learning models together reasonably, it can not only reduce variance in the training process, but also bias. Therefore, in this article, we select the stacking method to construct a learning model. Figure 3 shows a two-layer stacking model constructed by us.

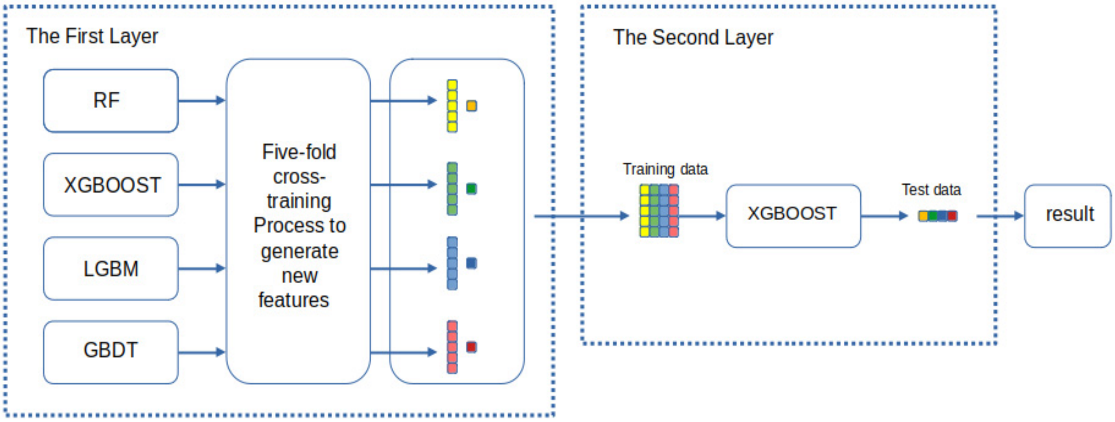


Figure 3. Structure of a two-layer stacking model.

This stacking model is constructed of two layers. The first layer combines four weak learners: RF, XGBOOST, LGBM and GBDT. Each learner needs to go through a five-fold-cross-training process individually to generate one new training datum and one new test datum. We combine these data into column vectors in the second layer, respectively, as new training data and test data, which will be used to train and test another weak learner, XGBOOST. Figure 4 shows the five-fold-cross-training process in detail.

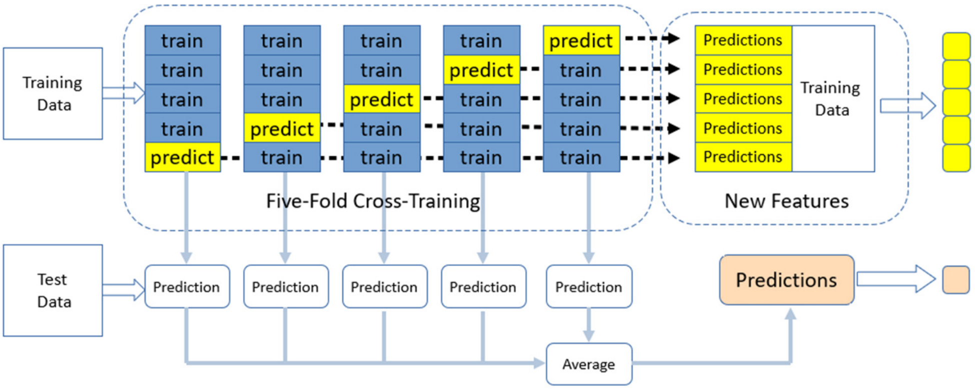


Figure 4. Five-fold cross-training process [23].

The five-fold cross-training process is an effective way to avoid over-fitting. In this process, training data will be divided into five folds. Each time, we use four folds of them to train a model and use this model to generate two predictions based on rest fold and test data. We repeat this process five time to obtain five predictions from five folds, which make up one training datum. We obtain five predictions from the same test data as well, taking their average to obtain one new test datum. We use these data to train and test the meta model in the second layer. Note that each basic model in the first layer should be retrained once under the whole training data as a final basic model.

Stacking provides a reasonable way to mix different learning models together. Given the new features, each basic model in the first layer gives out its predictions individually first. The meta model combines these predictions, comprehensively considers them, and gives out final predictions. Through this way, prediction error and variance could decrease, and the model could obtain better generalization ability. We will prove these advantages by research results later.

3.4. Power Optimization

3.4.1. Problem Formulation

Suppose our prediction model is $f(L)$, where L represents all features. Then, we can give out a basic optimization problem, which is formulated as

$$max_x f(L_r, x), x \in \Phi \tag{1}$$

where x represents features to be optimized, L_r represents rest features. Clearly, we have $L_r \cup x = L$. Φ is set composed of all possible values of x. Problem (1) represents a seek for optimal x to get maximum $f(L_r, x)$.

If such a kind of optimization is performed on dataset D, then (1) should be rewritten as

$$max_x \sum_D f(L_r, x), x \in \Phi \tag{2}$$

where $\sum_D f(L_r, x)$ represents a power accumulation on dataset D. Problem (2) represents a seek for optimal x to get maximum $\sum_D f(L_r, x)$ in dataset D.

3.4.2. Pitch Angle Control Strategy

Before introducing a pitch angle control strategy, we want to introduce a general control strategy first. Suppose the regulation period is T, x is a parameter to be optimized. Then, our general control strategy could be described as follows. At the beginning of each period, T_{cur}, we solve the optimization problem (2) on a historical dataset D_{last} of last period T_{last} to get optimal the parameter x_{best} for the current period T_{cur}, then, we set $x_{cur} = x_{best}$. Note that there should not be abnormal data in D_{last}.

In short, our strategy is a track for maximum power output. Below is the completed form of our algorithm (see Algorithm 1).

Algorithm 1: Power Optimization (Maximum Power Output Tracking)

Input: A trained model $f(L)$, historical dataset D_{last} in last period T_{last}, a set composed of all possible values of feature x to be optimized $C = \{x_1, x_2, \ldots, x_n\}$.

```
initialize: set maximum power accumulation as $S_{max} = \sum_{D_{last}} f(L_r, x_1)$, set optimal $x_{best}$ as
$x_{best} = x_1$.
for $x = x_1, x_2, \ldots, x_n$ do
    calculate $S_i = \sum_{D_{last}} f(L_r, x_i)$
    if $S_i > S_{max}$ then
        $S_{max} = S_i, x_{best} = x_i$
end for
set $x_{cur} = x_{best}$ for current period $T_{cur}$.
```

The regulation period T is an important parameter which decides whether we can tack maximum power output in real time or not. Generally speaking, a long period T will decrease the effectiveness of tacking and the optimization result may get worse. On the other hand, if T is small enough, the optimal parameter x_{best} in the last period T_{last} will be optimal in the next period. In this way, we could get the best power optimization result.

If parameter x is pitch angle, then we get the pitch angle control strategy according to general strategy. This strategy could be described by a flowchart, as shown in Figure 5, where p represents pitch angle.

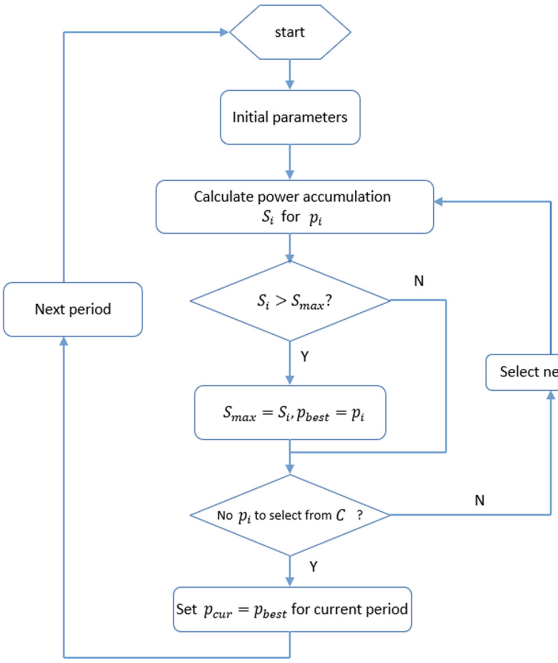

start
Initial parameters
Calculate power accumulation S_i for p_i
$S_i > S_{max}$?
N
Y
$S_{max} = S_i, p_{best} = p_i$
Next period
Select next p_i
No p_i to select from C ?
N
Y
Set $p_{cur} = p_{best}$ for current period

Figure 5. Flowchart of power optimization process.

4. Power Prediction Modeling

4.1. Data Description

Our data are provided by Wuzhong Baita Wind Power Corporation Limited. They cooperate with us and select two types of wind turbines in the wind farm as research objects. Each type contains four turbines, namely type A (#5, #6, #12, #13) and type B (#70, #71, #89, #91). Therefore, our entire data are from these eight wind turbines. Original data are at 1-min interval from 1 January 2019 to 15 May 2020. Some wind turbine parameters between type A and type B are different. Table 1 shows some of them.

Table 1. Specification of two types of wind turbines.

Wind Turbine Parameters	**Type**	
	A	**B**
Cut in, rated wind speed	3 m/s,11 m/s	2.5 m/s,10 m/s
Rotor diameter	88 m	93 m
Blade length	43 m	45.3 m
Hub height	80 m	90 m
Swept area	6079 m^2	6789 m^2

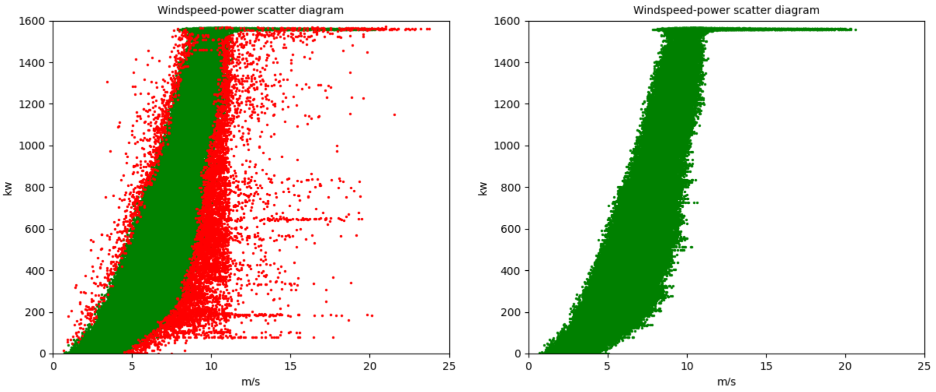
In our work, we chose two wind turbines from each type as research objects: #6, #12 from type A, #70, #89 from type B. For each turbine, we need one dataset for prediction modeling and one dataset for optimization. Table 2 describes these datasets in detail.

Table 2. Description of prediction modeling and optimization datasets.

Type	**Turbines Considered**	**Dataset**	**Timestamp**
A	#6	Prediction modeling	1st January 2019–15th May 2020
		Optimization	1st January 2019–31st December 2019
	#12	Prediction modeling	1st January 2019–15th May 2020
		Optimization	1st January 2019–31st December 2019
B	#70	Prediction modeling	1st January 2019–15th May 2020
		Optimization	1st January 2019–31st December 2019
	#89	Prediction modeling	1st January 2019–15th May 2020
		Optimization	1st January 2019–31st December 2019

4.2. Data Preprocessing

Data preprocessing on the original dataset is needed before modeling. As mentioned, this process contains two steps. In step 1, we eliminate those data with abnormal "state" field or a power output no larger than 0. In step 2, we use DBSCAN to remove the rest of the abnormal data. Here, we take #6 as example and show the performance of DBSCAN in Figure 6.



(**a**) (**b**)

Figure 6. Performance of density-based spatial clustering of applications with noise (DBSCAN): (**a**) result of DBSCAN, (**b**) final dataset.

As we can see from subgraph a, DBSCAN algorithm removes almost all abnormal data which distribute randomly around the wind power curve. The rest of the data will be used as final dataset for modeling.

4.3. Feature Selection

The original datasets contain 23 features and we only chose 12 of them. Three feature selection algorithms, Filter, Wrapper and Embedding, give out their selection, shown in Table 3. Note that in the Embedding method, we needed to choose one learning model to obtain the weight coefficient of each feature after training it. There were many models for us to choose like RF, LGBM and XGBOOST, but the training duration of each model is different. Considering that the XGBOOST model can be trained well within a short time, we selected it as the training model.

Table 3. Result of three feature selection algorithms.

NO.	Filter	Wrapper	Embedding (XGBOOST)
1	Wind Speed (30 s)	Blade 1 Pitch Angle Feedback A	Wind Speed (30 s)
2	Wind Speed (3 s)	Blade 2 Pitch Angle Feedback A	Yawing Error (3 s)
3	Wind Speed (5 m)	Gearbox Bear Temperature B	Cabin Location
4	Real Wind Speed	Gearbox Bear Temperature F	Gearbox Bear Temperature F
5	Wind Speed 1	Generator Stator Temperature L1	Wind Direction (3 s)
6	Generator Stator Temperature L3	Generator Stator Temperature L2	Blade 1 Pitch Angle Feedback A
7	Generator Stator Temperature L1	Generator Stator Temperature L3	Extra Temperature
8	Gearbox Bear Temperature F	Grid Power Factor	Wind Speed (5 m)
9	Generator Stator Temperature L2	Wind Speed (30 s)	Blade 3 Pitch Angle Feedback A
10	Gearbox Oil Temperature	Wind Speed (3 s)	Grid Power Factor
11	Blade 1 Pitch Angle Feedback A	Yawing Error 2	Gearbox Bear Temperature B
12	Cabin Temperature	Wind Speed (5 m)	Gearbox Bear Temperature B

Based on these selections, we need to make a comprehensive consideration. First, the Filter method chooses features based on their correlation with power out. Therefore, this selection has great reference value. Five parameters related to wind speed are considered in the Filter method: Wind Speed (30 s), Wind Speed (3 s), Wind Speed (5 m), Real Wind Speed and Wind Speed 1. They are duplicated to a certain extent, so we just chose two of them: Wind Speed (30 s) and Real Wind Speed. Three parameters related to generator stator temperature were considered: Generator Stator Temperature L3, Generator Stator Temperature L1 and Generator Stator Temperature L2. We only chose one of them: Generator Stator Temperature L1. As for rest of the parameters, Gearbox Bear Temperature F, Gearbox Oil Temperature, Blade 1 Pitch Angle Feedback A and Cabin Temperature, we selected all of them. Therefore, we obtained seven features. The rest of the five features were chosen from selections of the Wrapper and Embedding method. Specifically, we selected Gearbox Bear Temperature B, Grid Power Factor consulting with selection of Wrapper and chose Wind Direction (3 s), Cabin Location, Yawing Error (3 s) considering the selection of Embedding. These parameters have a high ranking in selections.

In the end, 12 features were selected as follows: Wind Speed (30 s), Real Wind Speed, Generator Stator Temperature L1, Gearbox Bear Temperature F, Gearbox Oil Temperature, Blade 1 Pitch Angle

Feedback A, Gearbox Bear Temperature B, Grid Power Factor, Wind Direction (3 s), Cabin Location, Yawing Error (3 s) and Cabin Temperature.

4.4. Model Construction

After data preprocessing on the original dataset, we made a division on it to obtain training data and test data. Specifically, we selected one datapoint from every 10 datapoints as a test point. These test points were combined as final test data. The rest of the datapoints were used as training data. We used training data to train models and evaluated them based on test data. Here, we chose three common evaluation indexes for evaluation: Mean Absolute Error (MAE), Root Mean Square Error (RMSE) and Coefficient of Determination (R^2) formulated as

$$MAE = \frac{1}{N}\sum_{i=1}^{n}\left|\hat{y}_i - y_i\right| \tag{3}$$

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

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y_i})^2} \tag{5}$$

In our work, six different models were trained in total: Neural Networks (NN), GBDT, LGBM, XGBOOST, RF and STACKING. Table 4 shows their performances on the test data of wind turbines #6 and #70. Their performances on other wind turbines are similar so we omitted them.

Table 4. Performances of different models on test data.

Wind Turbine		**#6**			**#70**	
Type		**A**			**B**	
Model	**MAE**	**RMSE**	**R^2**	**MAE**	**RMSE**	**R^2**
NN	36.91	61.39	0.973	37.13	59.90	0.971
GBDT	38.96	65.35	0.970	35.29	58.10	0.973
LGBM	36.07	60.42	0.974	32.61	53.93	0.976
XGBOOST	35.74	60.35	0.974	32.32	53.56	0.977
RF	34.03	58.61	0.976	31.05	52.76	0.977
STACKING	33.96	58.22	0.976	30.86	52.16	0.978

It is no doubt that the stacking model shows the best quality in MAE, RMSE and R^2 compared with the other models. Therefore, we chose the stacking model as the final prediction model.

4.5. Power Prediction

In this section, we performed a power prediction process based on test data to evaluate whether the stacking model could predict power output accurately or not. First, we used three indexes informed to describe performances of the stacking model on test data of four wind turbines in Table 5.

Table 5. Performances of stacking model on test data.

Type	**Turbine**	**MAE**	**RMSE**	**R^2**
A	#6	33.96	58.22	0.976
A	#12	37.45	63.56	0.972
B	#70	30.86	52.16	0.978
B	#89	34.04	59.50	0.974

Note that the coefficient of determination (R^2) of each stacking model is generally around 0.97, which represents a good fitting between actual power output and predicted power output. Therefore, these models could be used as power prediction model. Taking wind turbines #6 and 70 as examples, Figures 7 and 8 describe actual power, predicted power and absolute error intuitively.

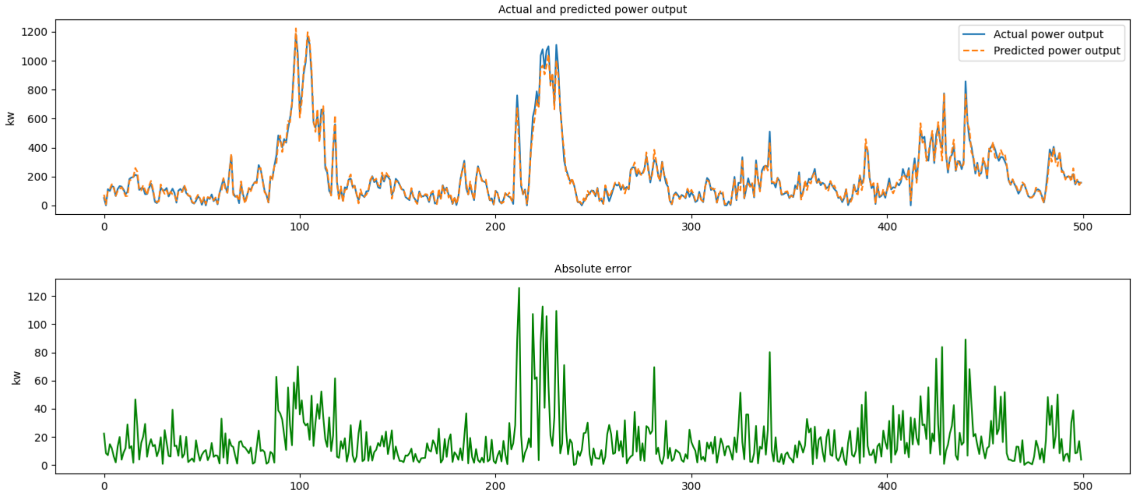


Figure 7. Power prediction on test data (#6, top 500 test points).

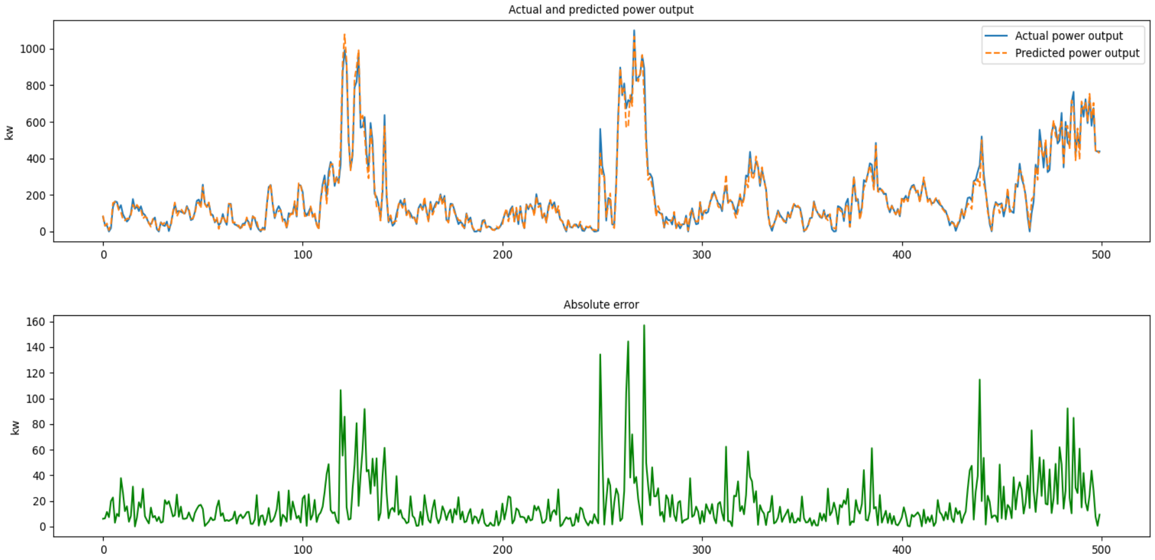


Figure 8. Power prediction on test data (#70, top 500 test points).

As shown in Figures 7 and 8, the fitting between actual power and predicted power is good. A common phenomenon is that the larger the value tabof actual power, the larger the value of absolute error. However, relative error is generally around 10%.

5. Power Optimization

After we obtained prediction models, the optimization process could be started. Here, we chose feature "Blade 1 Pitch Angle Feedback A", a parameter which represents the pitch angle of blade 1 as a parameter to be optimized; we introduced the pitch angle control strategy clearly in Section 3.4.2. The original data of each turbine contain a series of possible pitch angle values; we checked these values and collected them into an optimization set *C*. Table 6 describe these sets in detail.

Table 6. Description of optimization parameter and optimization set.

Type	Wind Turbine	Optimization Parameter	Optimization Set
A	#6	Blade 1 Pitch Angle Feedback A	$C = \{0°, -0.5°, -1°\}$
A	#12	Blade 1 Pitch Angle Feedback A	$C = \{0°, -0.5°, -1°\}$
B	#70	Blade 1 Pitch Angle Feedback A	$C = \{0.5°, 0°, -0.5°\}$
B	#89	Blade 1 Pitch Angle Feedback A	$C = \{0°, -0.5°, -1°\}$

The regulation period is optional. Minimum period is 1 min since original data are at 1-min intervals. As informed before, abnormal data should be avoided to participate in power optimization process. For the abnormal data in each period, the predicted power output under any pitch angle will be set as equal to actual power output.

5.1. Power Optimization for #6

In this section, we carry out power optimization for wind turbine #6 with the timestamp from 1 January 2019 to 31 December 2019. Figure 9 shows partial optimization results with regulation period T = 1 min.

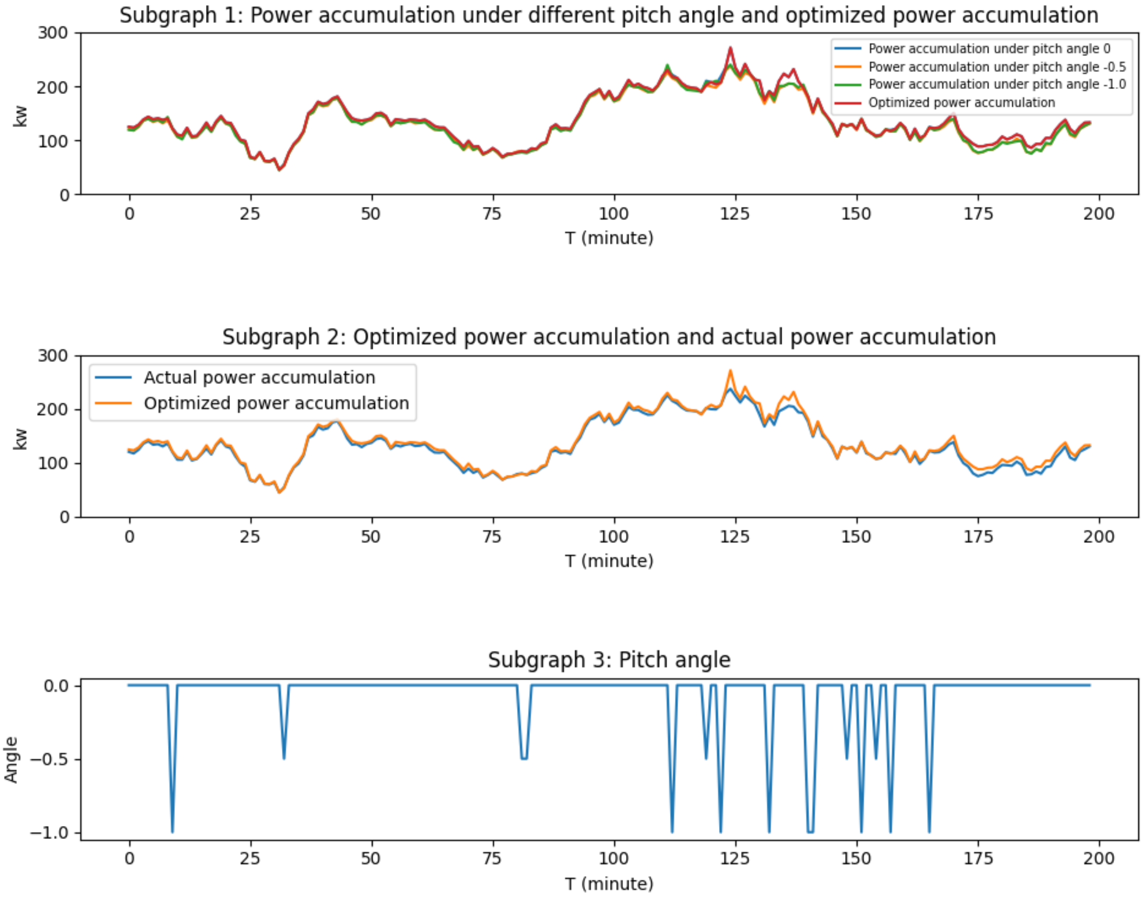


Figure 9. Partial power optimization results on #6 with T = 1 min.

Subgraph 1 shows optimized power accumulation and power accumulation under different pitch angles in each period. It proves the effectiveness of our pitch angle control strategy in maximum power output tracking. Since optimized power accumulation is at a maximum in most cases. We can find that power accumulation gets enhanced after the optimization process, as shown in Subgraph 2. Subgraph 3 shows the value of pitch angle in each period after optimization. Note that the regulation period could be changed, and a different period will influence the optimization results. Therefore, we tried different periods and give statistical results in Table 7.

Table 7. Power optimization results on #6 under different period.

T (minute)	Total Actual Power Accumulation (10^8 kw)	Total Optimized Power Accumulation (10^8 kw)	Increase Percentage
1	1.4161	1.4367	1.45%
50	1.4161	1.4367	1.45%
100	1.4161	1.4364	1.43%
500	1.4161	1.4358	1.39%
1000	1.4161	1.4358	1.39%

We can see that the increase percentage decreases as the period gets longer. It supports our opinion that a short regulation period generally brings better optimization results.

5.2. Power Optimization for #70

Like the optimization process for #6, first we set the regulation period T = 1 min. Figure 10 shows partial optimization results.

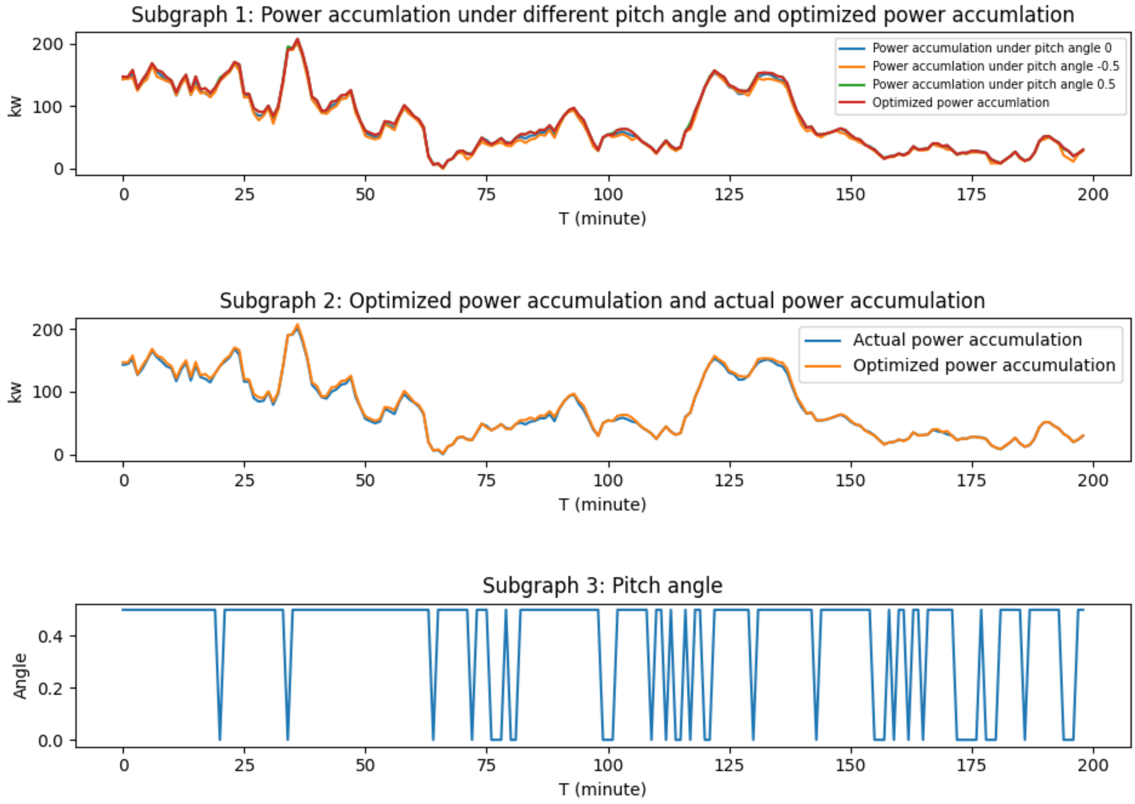


Figure 10. Partial power optimization results for #70 with T = 1 min.

Our pitch angle control strategy is proved to be effective in maximum power output tracking, as shown in Subgraph 1. Subgraph 2 represents power accumulation becoming enhanced after the optimization process compared with actual power accumulation. We use Table 8 to show optimization results under a different regulation period.

Table 8. Power optimization on #70 under different period.

T (minute)	Total actual Power Accumulation (10^8 kw)	Total Optimized Power Accumulation (10^8 kw)	Increase Percentage
1	1.4056	1.4162	0.75%
50	1.4056	1.4164	0.76%
100	1.4056	1.4160	0.73%
500	1.4056	1.4152	0.66%
1000	1.4056	1.4150	0.66%

Similar to Table 7, the increase percentage decreases as the period gets longer. Therefore, generally a short period would be better.

5.3. Power Optimization for Other Turbines

We also performed optimization for wind turbines #12, #89. Table 9 shows statistical results.

Table 9. Power optimization on #12, #89 under a different period.

Wind Turbine	Type	Period (minute)	Total Actual Power Accumulation (10^8 kw)	Total Optimized Power Accumulation (10^8 kw)	Increase Percentage
#12	A	1	1.3799	1.3930	0.94%
		50	1.3799	1.3907	0.78%
		100	1.3799	1.3892	0.67%
		500	1.3799	1.3857	0.58%
		1000	1.3799	1.3852	0.38%
#89	B	1	1.3735	1.3984	1.81%
		50	1.3735	1.3980	1.78%
		100	1.3735	1.3970	1.71%
		500	1.3735	1.3950	1.56%
		1000	1.3735	1.3965	1.67%

6. Conclusions

In this article, we proposed a new power optimization strategy to enhance the power output of wind turbines based on a stacking model and pitch angle adjustment. Our strategy includes two steps. In step one, we train a stacking model based on historical data to obtain the power prediction model. In step two, we carry out optimization based on prediction model and pitch angle adjustment. To prove the effectiveness of our strategy, we perform prediction modeling and the optimization process on four wind turbines, respectively.

Our strategy is proven to be reasonable and practical by the research results. First, we successfully obtained the power prediction model for each wind turbine by training a stacking model under original datasets. Three indexes MAE, RMSE and R^2, are considered to evaluate the trained model and they represent a good quality of prediction model, as shown in Section 4. We also accomplished the optimization process successfully. Specifically, based on the prediction model, we performed the optimization process on four wind turbines by adjusting their pitch angles. From research results in Section 5, we can see that power output is enhanced, which proves that our pitch angle control strategy is effective. According to the optimization results of wind turbine #6, #12, #70 and #89, the general power increase percentage is around 1% when the regulation period is set as 1 min, and the increase percentage generally decreases as the period gets longer. Our improvement could be enhanced if the regulation period is smaller because a short regulation period generally brings better optimization results, as we have emphasized. In the end, even though our optimization process is performed on historical data of the year 2019, it could be applied into real application as well.

In fact, power optimization for wind turbines has been researched a lot in recent years. However, many traditional methods have not involved any Machine Learning (ML) technology. In this article, Machine Learning is first applied to build a prediction model which plays an important role in optimization. Pitch angle adjustment is seldom considered in traditional design. In other word, pitch angle generally remains unchanged in these approaches. In our work, pitch angle adjustment is first applied for optimization, and our research indicates that by adjusting pitch angle appropriately, power output could be enhanced. Note that if the pitch angle changes too frequently, mechanical stress increases and the lifetime of mechanical system may be reduced. Therefore, an appropriate regulation period is important in real application.

Author Contributions: Formal analysis, Writing, Z.L.; Supervision, Z.S.; Resources, S.M.; Validation, F.M.; Visualization, Y.Q. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported Commonweal Technology Research Project of Zhejiang Province under Grant LGG18F030005.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. John, K.K.; Zafirakis, D. The wind energy (r)evolution: A short review of a long history. *Renew. Energy* **2011**, *36*, 1887–1901.
2. Dai, K.S.; Bergot, A.; Liang, C.; Xiang, W.N.; Huang, Z.H. Environmental issues associated with wind energy—A review. *Renew. Energy* **2015**, *75*, 911–921. [CrossRef]
3. GWEC. Global Wind Report. 2019. Available online: https://gwec.net/global-wind-report-2019/ (accessed on 31 March 2020).
4. Ciri, U.; Rotea, M.; Santoni, C.; Leonardi, S. Large-eddy simulations with extremum-seeking control for individual wind turbine power optimization. *Wind Energy* **2017**, *20*, 1617–1634. [CrossRef]
5. Hall, J.F.; Chen, D.M. Dynamic Optimization of DrivetrainGear Ratio to Maximize Wind Turbine Power Generation—Part 1: System Model and Control Framework. *J. Dyn. Syst. Meas. Control.* **2013**, *135*, 11016–11026. [CrossRef]
6. Song, M.X.; Chen, K.; Zhang, X.; Wang, J. The lazy greedy algorithm for power optimization of wind turbine positioning on complex terrain. *Energy* **2015**, *80*, 567–574. [CrossRef]
7. Mérida, J.; Aguilar, L.T.; Dávila, J. Analysis and synthesis of sliding mode control for large scale variable speed wind turbine for power optimization. *Renew. Energy* **2014**, *71*, 715–728. [CrossRef]
8. Xiong, L.Y.; Li, P.H.; Wu, F.; Ma, M.L.; Khan, M.W.; Wang, J. A coordinated high-order sliding mode control of DFIG for power and grid synchronization. *Int. J. Electr. Power Energy Syst.* **2019**, *105*, 679–689. [CrossRef]
9. Bektache, A.; Boukhezzar, B. Nonlinear predictive control of a DFIG-based for power capture optimization. *Int. J. Electr. Power Energy Syst.* **2018**, *101*, 92–102. [CrossRef]
10. Talebi, J.; Ganjefar, S. Fractional order sliding mode controller design for large scale variable speed for power optimization. *Environ. Prog. Sustain. Energy* **2018**, *37*, 2124–2131. [CrossRef]
11. Rajendran, S.; Jena, D. Backstepping sliding mode control of a variable speed wind turbine for power optimization. *J. Mod. Power Syst. Clean Energy* **2015**, *3*, 402–410. [CrossRef]
12. Fan, Z.X.; Zhu, C.H. The optimization and the application for the wind turbine power-wind speed curve. *Renew. Energy* **2019**, *140*, 52–61. [CrossRef]
13. Zhou, L.N.; Pan, S.M.; Wang, J.W.; Vasilakos, A.V. Machine Learning on big data: Opportunities and challenges. *Neurocomputing* **2017**, *237*, 350–361. [CrossRef]
14. Clifton, A.; Kilcher, L.; Lundquist, J.K.; Fleming, P. Using machine learning to predict wind turbine power output. *Environ. Res. Lett.* **2013**, *8*, 24009–24017. [CrossRef]
15. Felder, M.; Sehnke, F.; Ohnmeiß, K.; Schröder, L.; Junk, C.; Kaifei, A. Probabilistic short term wind power forecasts using deep neural networks with discrete target classes. *Adv. Geosci.* **2018**, *45*, 13–17. [CrossRef]
16. Li, W.H.; Xiao, Q.; Liu, J.L.; Liu, H.Q. Application and contrast analysis of BP and RBF neural network in short-term prediction. *Appl. Mech. Mater.* **2014**, *492*, 544–549. [CrossRef]

17. Schubert, E.; Sander, J.; Ester, M.; Kriegei, H.; Xu, X. DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN. *ACM Trans. Database Syst.* **2017**, *42*, 1–21. [CrossRef]
18. Li, J.D.; Cheng, K.W.; Wang, S.H.; Morstatter, F.; Trevino, R.; Tang, J.L.; Liu, H. Feature Selection: A Data Perspective. *ACM Comput. Surv.* **2018**, *50*, 1–45. [CrossRef]
19. Divina, F.; Gilson, A.; Goméz-Vela, F.; Torres, M.G.; Torres, J. Stacking ensemble learning for short-term electricity consumption forecasting. *Energies* **2018**, *11*, 949. [CrossRef]
20. Kwon, H.; Park, J.; Lee, Y. Stacking Ensemble Technique for Classifying Breast Cancer. *Healthc. Inform. Res.* **2019**, *25*, 283–288. [CrossRef]
21. Shi, F.; Liu, Y.H.; Liu, Z.; Li, E. Prediction of pipe performance with stacking ensemble learning based approaches. *J. Intell. Fuzzy Syst.* **2018**, *34*, 3845–3855. [CrossRef]
22. Demir, N.; Dalkilic, G. Modified stacking ensemble approach to detect network intrusion. *Turk. J. Electr. Eng. Comput. Sci.* **2018**, *26*, 418–433. [CrossRef]
23. Yuan, T.K.; Sun, Z.F.; Ma, S.H. Gearbox Fault Prediction of Wind Turbines Based on a Stacking Model and Change-Point Detection. *Energies* **2019**, *12*, 4224. [CrossRef]