ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/331175798

Complete Statistical Analysis to Weather Forecasting

Conference Paper · January 2019

CITATIONS
2

READS
2,325

3 authors:

Anisha Datta
Jalpaiguri Government Engineering College
6 PUBLICATIONS **15** CITATIONS

SEE PROFILE

Shukrity Si
International Institute of Information Technology, Hyderabad
6 PUBLICATIONS **15** CITATIONS

SEE PROFILE

Sanket Biswas
Autonomous University of Barcelona
11 PUBLICATIONS **41** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project Aggression Detection on Multilingual Social Media Text View project

Project Leaf Disease Detection View project

All content following this page was uploaded by Shukrity Si on 16 July 2019.

The user has requested enhancement of the downloaded file.

Complete Statistical Analysis To Weather Forecasting

Anisha Datta[1], Shukrity Si[1⋆], and Sanket Biswas[2]

[1] Department of Computer Science and Engineering, Jalpaiguri Government Engineering College, Jalpaiguri, India, Pin-735102
dattaanishadatta@gmail.com,sukriti.si98@gmail.com

[2] Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, Kolkata-700108, India
sanketbiswas1995@gmail.com

Abstract. The primary objective of the model applied in this work is to predict the weather of a city named Austin in Texas using supervised machine learning algorithms . In this case, artificial neural networks and gradient boosting classifier were implemented to build models to predict weather and comparisons between these two models are also made for this dataset. Here average temperature, average dew point, average pressure of sea-level, average percentage of humidity etc. are the parameters taken into consideration which influence the weather of the place. Using these parameters, the trained models performed a classification to predict whether the weather is rainy (thunderstorm or not), not rainy, snowy or foggy.

Keywords: Supervised learning, Neural Networks, Gradient boosting classifier, comparison

1 Introduction

Machine learning is a branch of data science that basically plays with the data in a statistical manner.If we have input and output data then it builds models from the data and the models are used for prediction or solving the given tasks.We can break it into two parts: Learner and Reasoner. With the help of the experience of the data and background knowledge,learner builds models and with the help of models,reasoner solves tasks.Here machine learning algorithms were applied to implement this work.
Weather forecasting is the work of predicting the condition of atmosphere on the basis of some given data like temperature,pressure,wind-speed,humidity etc at a specified area.Meteorological approach is very popular for predicting weather.Many meteorological instruments like thermometer,hydrometer,barometer,rain gauges etc are used for prediction.But,it is very complex to merge all the information together and predict the state of atmosphere.But,predicting the state of

[⋆] corresponding author

atmosphere is very important for our daily life especially for the farmers and for any outdoor work.If weather is predicted correctly farmers can know when to cultivate crops,surfers can know when large waves are expected,people can plan accordingly when and where to go on their leisure time and aircraft and shipping are also dependent on the state of atmosphere.The road accidents due to fog,alarm for thunderstorms,Tsunami these are the main reasons why correct weather prediction is so important. So,we have introduced machine learning models to predict weather.Here is our project on weather forecasting of Austin, the capital of the U.S. state of Texas.So far we are concerned about weather prediction and comparisons between the models used here.
Two models are used: artificial neural networks and gradient boosting classifier algorithm.A weather dataset for Austin, Texas was obtained and used to train this algorithm.The dataset contains some features of atmospheric condition as inputs and the outputs are generated based on the results whether the particular day will be rainy, snowy or something else and the efficiency of these two models are also compared in our study for this dataset.

2 Related Work

We examined some papers on machine learning for weather prediction, linear regression and functional regression[1] were used in one paper and on the other hand, nave bayes and C4.5 decision tree[3], multiclass support vector machines[4],[6], ARIMA model[5] were used in the other papers and a good amount of literature is also witnessed based on artificial neural network[2],[6],[7]. In some papers, data mining technique[7],[8],[9] and normal equation methods[10] were also used. But these approaches didnt become successful to identify the abnormal pattern of the weather. Between the two regression models, linear regression model was unstable to outliers due to its high variance and low bias and functional regression model was also not relevant due to its low variance and high bias for that paper[1]. The problem could be solved by collection of more data. In other paper, the performance of C4.5 decision tree was very good but the performance of nave bias was very poor. With the increase of attributes in the dataset, the performance of nave bayes drastically affected but the decision tree handled the problem in suitable manner[3]. But, the time taken to build the models was less in case of nave bayes. In the case of multiclass support vector machines and neural network, due to complex data systems, categorical and continuous pattern of the weather, noisy and high dimensional data effective forecasting analysis could not be achieved[2][4][6]. So, effective models to predict weather are to be analyzed. In this paper, we have strove to build efficient models to predict weather. Neural networks and gradient boosting classifier are used in our study.

3 Preprocessing and Features Analysis

Here, average temperature,average dew point,average humidity,average pressure of sea-level,average visibility,average wind-speed and average wind-gust are used as features in the datasets to predict weather.The information is retrieved from mid 2014 to 2018 from the database of Austin airport.There are seven classes in this datasets: rainy, not rainy, snowy, foggy, thunderstorm, rainy with thunderstorm and fog.
Now we come to the point how the selected features affect our weather.From the knowledge of fluid mechanics we know that the air particles do not stay stable, rather they tend to move rapidly in free space.The more space it gets the more it spreads thus lowering the pressure and tries to cool off.As the earth gets heat during day time,the air above the ground becomes more hot than the upper part of the atmosphere due to radiation.The hot air having low pressure gradually goes up to the space by convection heat transfer and spreads rapidly as the height increases.The vacuum created on the ground gets filled by the cold air of surrounding.If the temperature goes too high, then the vacuum is created more rapidly and the cold air tries to occupy it with high speed giving rise to powerful wind gust or storm.Now if there is some water-body, the hot air carries moisture and rises up.Up there it gets cold and cold air can not carry that much moisture.The temperature and amount of moisture it gets saturated with (when it can not carry further) are called dew point and humidity respectively.So it then gives rise to rain if the water droplets are big, otherwise fog in tropical region and snow in polar region.The low the dew point and humidity, the more the weather remains cool.The visibility through the air ,depending on the wind speed or rain, can thus predict the weather also.
Here,we have calculated the central tendency of the datasets.1305 sample features are used in the datasets.So,it can be useful to represent the entire data set with a single value that describes the average value of the entire set.In statistics,this value is called the central tendency.There are three types of central tendency: Mean,Median and Mode.The values are shown in the table.we are also interested in ones that describe the spread or variability of the data values. So, standard deviation is calculated. The values are shown in table no 1.Here SD stands for standard deviation.

To understand the distribution of the data better Kernel Density Estimation is done. Kernel Density Estimation (KDE) is a non-parametric way of estimating the probability density function (pdf) of ANY distribution given a finite number of its samples. The pdf of a random variable X given finite samples, as per KDE formula, is given by:

$$\overline{f_h}(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{nh}\sum_{i=1}^{n} K(\frac{x - x_i}{h}) \qquad (1)$$

x_i=independent and identically distributed sample,K=kernel(a non-negative function that integrates to one),$h(> 0)$=smoothing parameter(Bandwidth)
Graphs of the histograms of the dataset are shown in figure 1.

Table 1. Statistics of the features

Features	**Mean**	**Median**	**Mode**	**Standard Deviation**
Average Temp	70.557854	73	84	14.009579
Average Dew Point	56.636782	61	71	14.862556
Average Humidity	66.662835	67	64	12.503302
Average Pressure	30.022835	30	29.95	0.171879
Average Visibility	9.162452	10	10	1.459463
Average Wind Speed	5.009195	5	4	2.081891
Average Wind Gust	21.383908	21	20	5.887797

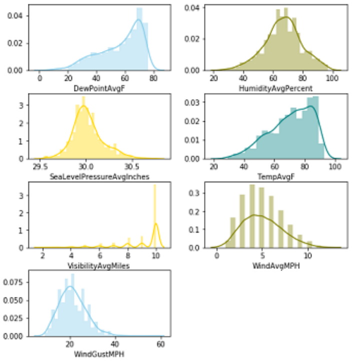

Fig. 1. Histogram Plot

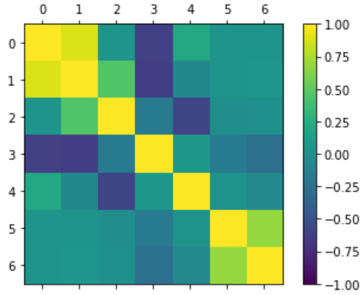

Fig. 2. Correlation Matrix

Now, we have found the correlation between the variables through correlation matrix. Correlation matrix is great way to explore new data. It can measure correlation between every combinations of the variables. It doesn't really matter if there is an outcome at this point or not, but it will compare everything against everything. Correlation coefficient is a measure of similarity between two vectors of numbers. the value can range between 1 and -1 where 1 is perfectly correlated, -1 is inversely correlated and 0 is not correlated. And our correlation matrix is given in figure 2.

Observing the distribution and correlation of the data,we have used standardization for preprocessing of the dataset. Standardization is the process of putting different variables on the same scale and a standardize variable is a variable that has been rescaled to have a mean of zero and a standard deviation of one. If,we start with a variable x and generate a variable x*, then the process is :

$$x^* = \frac{(x-m)}{sd} \qquad (2)$$

[where m is the mean of x and sd is the standard deviation of x]
Normalization is also used here for preprocessing of the dataset. Normalization scales all numeric variables in the range [0,1]. If,we start with a variable x and

generate a variable x ,and if, minimum value is called min and maximum value is called max, then the process is :

$$x' = \frac{(x - min)}{(max - min)} \qquad (3)$$

4 Proposed Methodology

Two algorithms are used for this prediction: Neural networks and Gradient Boosting Classifier.

4.1 Gradient Boosting -

Gradient Boosting is a greedy algorithm and it is a powerful model for prediction.It can be stated as Hypothesis Boosting Problem. This algorithm consists of loss function,weak learner and additive model. This algorithm works on the belief that a weak learner can be a better learner with the help of several regularization methods.
As base learner, Decision Trees of fixed size are used with Gradient Boosting algorithm.It is a method of ensemble learning that implements the sequential boosting algorithm.Basically the goal of GBM(Gradient Boosting Machine) is to reduce the expectation of the loss function.To achieve this, the residual of the initial model is calculated.Then the base(or, weak) learner is fitted to the residual by the gradient descent algorithm.Thus the model is updated by adding the weighted base learner to the previous model.Finally the target model is obtained by iteratively conducting the previous steps.

Now assuming that the no. of leaves for each tree is 'J', the space of the m^{th} tree can be divided into 'J' disjoint subspaces(or, leaves) such as R_{1m},R_{2m},...,R_{Jm} and the predicted value for subspace R_{jm} is the constant b_{jm}.The regression tree for input x_i can be expressed as-

$$g_m(x_i) = \sum_{j=1}^{J} b_{jm}.I(x_i \epsilon R_{Jm}) \qquad (4)$$

$$where \quad I(x_i \epsilon R_{Jm}) = 1, if x_i \epsilon R_{Jm}$$
$$otherwise \qquad = 0 \qquad (5)$$

To minimize the loss function, we use the steepest descent method.We take the approximate solution as-

On summation, $F(x_i) = \sum_{m=0}^{M} f_m(x_i)$ (6), M denotes the index of the tree, f_m is the learning model(or, function) for each input x_i

Here we can write,

$$f_m(x_i) = -\rho_m g_m(x_i) \qquad (7)$$

Where gradient $g_m(x_i)$(pseudo-residual for a given tree) is-

$$g_m(x_i) = [\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}] \qquad ,\text{with } f(x_i) = f_{m-1}(x_i) \qquad (8)$$

And the multiplier ρ_m(a constant for optimisation) is-

$\rho_m = argmin_p \sum_{i=1} nL(y_i, f_{m-1}(x_i) + \rho_m g_m(x_i))$ (9)

The updated model is expressed as:

$F_m(x_i) = F_{m-1}(x_i) + \rho_m g_m(x_i)$ (10)

—- this is the most important step (i.e. iteration) of this algorithm.

4.2 Artificial Neural Networks -

In our brain, the neurons act as a unit cell which processes information through a connecting bridge called the synopses. These neurons communicate with other neurons through chemical signals and process the data. Like these neurons in biological system,the computer scientists have created a model named Artificial Neural networks(ANNs) to process data in a big data system in order to predict future outcomes by observing the previously recorded data pattern. This model is the foundation of AI(Artificial Intelligence) which is nothing but a machine that works much faster than a human brain. This learns from the human activity pattern upon training and gives the future analysis based on the learnt algorithms.The more collective data it gets, the more it performs well.

The ANN model has a activation function which calculates the weighted sum of the inputs i.e. given featured data. Here the weighted inputs create a multilayered network. By back propagation method(i.e. based on gradient descent algorithm for error calculation) we can learn the pattern and reduce the error with multiple iteration.

Here is the activation function(x,w in vector form)-

$A_j(\overline{x}, \overline{w}) = \sum_{i=0}^{n} x_i w_{ji}$, (11)

where w_{ji} is the weight connecting neuron j to neuron i(each neuron of previous layer) We get the actual output per neuron j as :

$O_j(\overline{x}, \overline{w}) = \frac{1}{1 + e^{A_j(\overline{x}, \overline{w})}}$ (12)

The error is the difference between the actual(O_j) and the desired output(d_j).We need to modify the weights in order to minimize the error. We can define the squared error function for the output of each neuron j as :

$E_j(\overline{x}, \overline{w}, d) = (O_j(\overline{x}, \overline{w}) - d_j)^2$ (13)

To adjust the weights of each neuron, we use back propagation algorithm which iteratively adds some random weight to the previous weight. The additional weight depends on the learning rate and gradient of error E w.r.t each given weight as-

$\Delta w_{ji} = -\eta \frac{\delta E}{\delta w_{ji}}$ (14),where η is the learning rate('+' constant)

Now, we calculate how much the error depends on the output :

$\frac{\delta E}{\delta O_j} = 2(O_j - d_j)$ (15)

The output can be expressed in terms of the inputs as-

$$\frac{\delta O_j}{\delta w_{ji}} = \frac{\delta O_j}{\delta A_j}.\frac{\delta A_j}{\delta w_{ji}} = O_j(1 - O_j)x_i \qquad (16)$$

Thus adjustment to each weight will be-

$$\Delta w_{ji} = -2\eta(O_j - d_j)O_j(1 - O_j)x_i \qquad (17)$$

So, the weights are modified as :

$$w_{ji}(t + 1) = w_{ji}(t) + \delta w_{ji} \qquad (18)$$

This modified weights(weight w_{ji} at time t+1) help in reducing the error generated in each layer thus help the model learn better from the training dataset and give better prediction.

5 RESULTS

In our study, we have observed accuracy, precision, recall and F1 score of the used model and made a comparison between these models based on the values. We got these values by preprocessing the dataset with two methods- standardization and normalization and one with no-preprocessing.We can see in each case the values are increased after preprocessing the dataset.It is also seen that the algorithm Gradient Boosting has worked well over ANN giving better results in most of the cases.Now we come to the confusion matrix from which we can calculate those things.

The confusion matrix is a way to summarize the performance of a classifier for classification tasks. The square matrix consists of columns and rows that list the number of instances relative or absolute True class vs. Predicted class ratios.

For the binary classification,the two outputs are positive and negative and its confusion matrix are shown in table no 2. Here,TP - True Positive(predicted true and actually true), FN - False Negative(predicted false but actually true) ,FP - False Positive(predicted true but actually false) , TN - True Negative(predicted false and actually false)

Table 2. Confusion Matrix of Binary Classification

Predicted(cols)/ True(rows)	**Positive**	**Negative**
Positive	TP	FN
Negative	FP	TN

But, in our study we have multiclass classification. For multiclass classification, if we have N classes then we will get N*N matrix. The heatmap representations of the confusion matrices of our study using gradient boosting classifier and artificial neural network are given in figures 3-8 below with no preprocessing,standardization and normalization respectively.

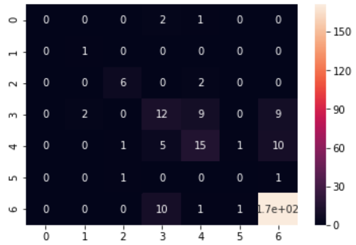

Fig. 3. GBM(no preprocessing)

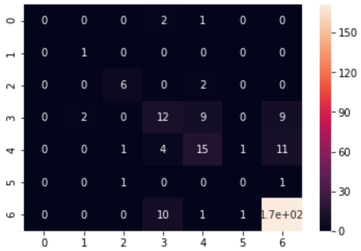

Fig. 4. GBM(standardised)

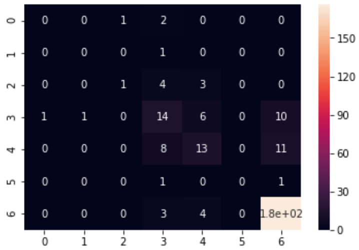

Fig. 5. GBM(normalised)

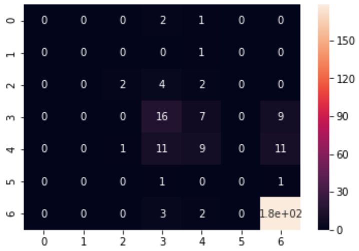

Fig. 6. ANN(no preprocessing)

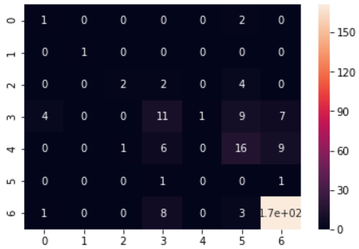

Fig. 7. ANN(standardised)

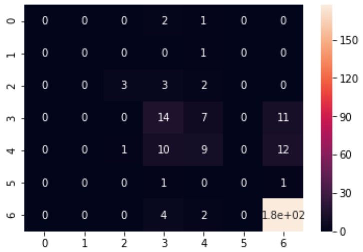

Fig. 8. ANN(normalised)

We have calculated accuracy, recall, precision, F score for multiclass classification by using this N*N matrix(M) shown in table no. 3 to 6 below.The formulae are also given.

Equations-

$$1.Accuracy = \frac{true\ \ prediction\ \ of' +' ve\ \ and\ \ ' -' ve\ \ classes}{Population\ \ of\ \ all\ \ classes} \tag{19}$$

$$2.Precision_i = \frac{M_{ii}}{\sum_j M_{ij}} \tag{20}$$

$$3.Recall_i = \frac{M_{ii}}{P} = \frac{M_{ii}}{\sum_j M_{ji}} \tag{21}$$

$$4.F1 - score = \frac{2}{\frac{1}{recall} + \frac{1}{precision}} \tag{22}$$

(i.e the harmonic mean of recall and precision)
M_{ii}=diagonal values of the matrix which represents true prediction(TP) , M_{ij}=Every Case predicted as true but actually may be true or false (TP+FP) , M_{ji}=total no. of true cases where predictions were sometimes true and sometimes not (TP+FN)
The more the value of diagonal elements with less value for the non-diagonal ones, the more correct predictions the model will give in distinguishing between the classes.

Table 3. Accuracy prediction by these two algorithms using no preprocessing one time and two preprocessing methods standardization and normalization

	No preprocessing	**Standardization**	**Normalization**
Gradient Boosting	78.16	78.92	78.54
Artificial Network	78.16	78.54	77.78

Table 4. Precision prediction by these two algorithms using no preprocessing one time and two preprocessing methods standardization and normalization

	No preprocessing	**Standardization**	**Normalization**
Gradient Boosting	76.37	77.023	75.641
Artificial Network	75.134	77.291	74.270

Here, we have made a comparison between Gradient Boosting and Neural Network algorithm on the basis of accuracy, precision, recall and F1 score. The results are pretty much close but Gradient Boosting Classifier with standardization model is better than the other models in this case. For this dataset,

Table 5. Recall prediction by these two algorithms using no preprocessing one time and two preprocessing methods standardization and normalization

	No preprocessing	**Standardi-zation**	**Normali-zation**
Gradient Boosting	78.16	78.93	78.54
Artificial Network	78.16	78.544	77.778

Table 6. :F1 score prediction by these two algorithms using no preprocessing one time and two preprocessing methods standardization and normalization

	No preprocessing	**Standardi-zation**	**Normali-zation**
Gradient Boosting	77.05	77.80	76.56
Artificial Network	76.056	77.557	75.55

preprocessing has played a key role and standardization gives better result. But, the result can be made better for Neural Network with both standardization and normalization if the study is extended.

6 CONCLUSION

In our study, we use machine learning algorithms for weather prediction and these models yield good results and can be considered as an alternative to traditional metrological approaches. The study explains the effectiveness of machine learning algorithms for predicting various weather phenomena like rain, thunderstorm, snow, fog etc. Here, we use gradient boosting classifier and artificial neural network algorithms and after observing the comparison of the results between two models, we can say that they are well suited models for this kind of application. It also concludes that the Back Propagation Algorithm is also capable of predicting weather and can also be applied to this kind of weather forecasting data. And further improvement can be made to the results of these models by doing proper preprocessing to the dataset at early stage.

References

1. Mark Holmstrom, Dylan Liu, Christopher Vo, Machine Learning Applied to Weather Forecasting. Stanford University (Dated: December 15, 2016)
2. Sanjay D. Sawaitul, Prof. K. P. Wagh, Dr. P. N. Chatur, Classification and Prediction of Future Weather by using Back Propagation Algorithm-An Approach. International Journal of Emerging Technology and Advanced Engineering (ISSN 2250-2459, Volume 2, Issue 1, January 2012
3. Fahad Sheikh, S. Karthick, D. Malathi, J. S. Sudarsan and C. Arun, Analysis of Data Mining Techniques for Weather Prediction. Indian Journal of Science and Technology, Vol 9(38), DOI: 10.17485/ijst/2016/v9i38/101962, October 2016.

4. Wenying Zhang, Huaguang Zhang, Fellow, IEEE, Jinhai Liu, Kai Li, Dongsheng Yang, and Hui Tian, Weather Prediction With Multiclass Support Vector Machines in the Fault Detection of Photovoltaic System. IEEE/CAA JOURNAL OF AUTOMATICA SINIC, VOL. 4, NO. 3, JULY 2017
5. G.Vamsi Krishna, An Integrated Approach for Weather Forecasting based on Data Mining and Forecasting Analysis. International Journal of Computer Applications (0975 8887), Volume 120 No.11, June 2015
6. Janani.B, Priyanka Sebastian, ANALYSIS ON THE WEATHER FORECASTING AND TECHNIQUES. International Journal of Advanced Research in Computer Engineering and Technology (IJARCET), Volume 3, Issue 1, January 2014
7. R. Samya, R. Rathipriya, Predictive Analysis for Weather Prediction using Data Mining with ANN: A Study. International Journal of Computational Intelligence and Informatics, Vol. 6: No. 2, September 2016
8. Divya Chauhan, Jawahar Thakur, Data Mining Techniques for Weather Prediction: A Review. Data Mining Techniques for Weather Prediction: A Review, Volume: 2 Issue: 8
9. Prashant Biradar, Sarfraz Ansari, Yashavant Paradkar, Savita Lohiya, Weather Prediction Using Data Mining. 2017 IJEDR — Volume 5, Issue 2 — ISSN: 2321-9939
10. Sanyam Gupta, Indumathy K, Govind Singhal, Weather Prediction Using Normal Equation Method and Linear regression Techniques. Sanyam Gupta et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 7 (3) , 2016, 1490-1493 www.ijcsit.com 1490

View publication stats