Prediction of crop production using adaboost regression method

1Dr Shirin Bhanu Koduri, 2Loshma Gunisetti, 3Ch Raja Ramesh, 4K V Mutyalu and 5D. Ganesh

1,2,3,4,5CSE Department, Sri Vasavi Engineering College, Pedadarepalli, Tadepalligudem, India

Abstract: Territorial evaluations or forecast of yield creation is basic for some applications, for example, agrarian grounds administration, nourishment security cautioning framework, sustenance exchange strategy. Machine learning has risen with enormous information advancements and superior processing to make new open doors for information escalated science in the multi-disciplinary agricultural space. In this paper, we have applied and build a crop production prediction model using Decision Tree Classification and AdaBoost Regression Method. We have used the Indian Agriculture dataset. Performance analysis was done using R-squared Score.

Keywords— AdaBoost, agriculture, crop production, regression, R-squared score.

1. Introduction
Agriculture is the backbone of Indian economy. Agribusiness area utilizes more than 50 for each penny of the aggregate workforce in India and contributes around 17-18 percent to the nation’s Gross Domestic Product GDP. Artificial Intelligence is an area of computer science; it has the capability of machine to reproduce intellectual human behavior. Machine Learning is a subarea of Artificial Intelligence. In machine learning, we do not need to explicitly indicate the steps or conditions as in case of some programming applications. Regression is a technique to find the statistical relationship between two or more attributes associated with, and depends on, a change in one or more independent attributes.

Ensemble is the specialty of consolidating differing set of learners together to improve the balance and model prediction. Ensemble learning is a machine learning approach where numerous learners are prepared to take care of a same problem. Rather than customary machine learning approaches which attempt to take in one hypothesis from data used training, ensemble techniques endeavor to build an arrangement of theories and join them for use.

Boosting is one kind of ensemble procedure which endeavors to distinguish a solid classifier from an arrangement of classifiers which are weak. The different types of boosting algorithms are:

- AdaBoost (Adaptive Boosting)
- Gradient Boosting
- XGBoost
AdaBoost is a machine learning algorithm used for improving decision trees performance. Gradient boosting is a machine learning algorithm which can be utilized for regression and classification problems. It produces a prediction model consisting of ensemble of weak prediction models, typically decision trees. XGBoost is an efficient and optimal technique based on a distributed gradient boosting library.

2. Literature Review
There exist various methods for crop production estimation of satellite data. Dech Thammasiri, et al.[11] have designed ensemble based classification technique using decision tree, artificial neural network, and support vector machine models weighing classifier using AdaBoost in order to increase classification accuracy. Tang, et al. [10], modeled a effective learning algorithm Bagging-AdaBoost ensemble algorithm post optimization done with genetic algorithm. Anas Ahachad, et al. [1] have designed a simplified and efficient updation to Real AdaBoost emphasis procedure, which is based on combining the emphasis value of each sample with those of its K nearest neighbors in a convex manner.

Gaitang Wang et al. [12] used a dynamic AdaBoost ensemble technique, which was applied to function approximation problem and application of classification. Chouaib, et al. [2] designed a effective technique for simplified genetic algorithms (GAs) for selection of features. They used AdaBoost classifiers combination to evaluate a population sample.

We have applied Adaptive Boosting (AdaBoost) Algorithm. This system is a vital source for formulating crop related models and assessing their impacts.

3. Contribution of this paper
AdaBoost is boosting algorithm to improve the weak classifiers performance by reinforce training on incorrect classified samples. AdaBoost algorithm widely used to boost the weights of weak classifiers based on a classification error function [2]. Hence, classifier is denoted by

\[ h(x) = \begin{cases} 1 & \text{if} \sum_{i=1}^{M} c_i h_i \geq \text{threshold} \\ 0 & \text{otherwise} \end{cases} \]  

(1)

Where 1 indicates that the sample is belonging to the actual class. AdaBoost models [3] have the capability to achieve precision simply above arbitrary chance on a classification issue. Using Decision Trees with one level is the best choice with AdaBoost technique. These decision trees consist of one level of classification because of which they are referred as decision stumps. Every instance of training dataset has a weight.

Weight is initialized to:

\[ \text{weight} (x_i) = 1/n \]  

(2)

Where \( x_i \) is the \( i \)th training instance and \( n \) is the number of training instances.

Weak models are considered in a contiguous order and then training is done by training data which is already weighted. The technique then continues till a pre-defined count of weak learners has been created or there are no further updates to the training dataset. A collection of weak learners is available, each with a stage value after finishing point. Predictions are done by recalculating the weighted mean of the weak classifiers.

Every weak learner predicts either +1.0 values or -1.0 values, for every input instance. Each weak learner’s stage value is used to weigh the predicted value. The ensemble method prediction is considered to be a summation of weighted predictions. The first class is predicted, if the sum is positive and the second class is predicted, if negative.
4. Methodology

A. AdaBoost ensemble framework for Prediction of Crop Production

![AdaBoost ensemble framework for Prediction of Crop Production](image)

**Figure 1:** AdaBoost ensemble framework for Prediction of Crop Production

B. AdaBoost Algorithm:

Input: Initially set of n examples are considered as Training Set.

\[ I_{P} = \{ (x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \} \]

DecisionTreeRegressor denoted as Weak Learner

Integer \( ITR \) specifying total number of iterations.

Initialize:

\[ W_{itr} = \frac{1}{n}, i.e. W = (w_1, w_2, \ldots, w_n) = \{1/n, 1/n, \ldots, 1/n\} \]

The ensemble \( P = \emptyset \)

For \( itr = 1, 2, \ldots, ITR \)

Step 1. Get an example \( R_{itr} \) from \( I_{P} \) using distribution \( W_{itr} \)

Step 2. \( P_{itr} \) is built using the training set \( R_{itr} \)

Step 3. Calculate \( E_{itr} = \sum_{i} I_{err}(x_i) \) and \( \alpha_{itr} = 0.5 \ln \left( \frac{1-E_{itr}}{E_{itr}} \right) \)

Step 4. Update the weight: \( W_{itr+1} = \text{normalize}(W_{itr} \cdot \exp(-\alpha_{itr} \cdot I_{err})) \)

Output: The ensemble \( P = \{ p_1, p_2, \ldots, p_{ITR} \} \) and \( \Lambda = \{ \alpha_1, \alpha_2, \ldots, \alpha_{ITR} \} \)

C. Data Set

We have used crop production data set available at [5]. This dataset is being used to investigate and analyze production contribution to particular land, Agro-climatic region wise production, and high yield production structure for crops, crop growing stages and heterogeneity. Table I shows selection of training-testing set distribution. Sample dataset is shown in Table II. Dataset used is a Comma Separated Values file.

| Training Set Size | Test Set Size |
|-------------------|--------------|
| 80%               | 20%          |
| 70%               | 30%          |
| 60%               | 40%          |
In data preprocessing, non-numerical labels (hashable and comparable) are converted into numerical labels. Target variable (Production) has null values, so we can either remove or replace with other values. Here we have removed null values as Target variable has only 2% null values.

D. Platform and Python Modules used

➢ Google Colab free K80 GPU platform for testing out our Machine Learning prototype. [6]
➢ Python Modules[7]:
  • pandas
  • numpy
  • sklearn.preprocessing [9]
  • sklearn.ensemble.AdaBoostRegressor
  • sklearn.tree. DecisionTreeRegressor
  • pylab[8]
  • sklearn.metrics
  • pandas
  pandas is an open-source Python library provides data structures and data analysis tools.
  • numpy
  numpy[7], Numerical Python library contains multidimensional array objects. It also has a collection of routines for array processing. We can perform mathematical and logical operations on arrays.
  • sklearn.preprocessing
  The sklearn.preprocessing package consists of several common utilities and transformer classes to convert raw feature vectors into a representation that is more suitable for the downstream estimators.
  • sklearn.ensemble.AdaBoostRegressor
  An AdaBoost regressor is a technique used for meta-estimation that starts with regression of the original dataset. Additional copies of the regression model are generated on the same dataset. Instances weights are refined based on error of the latest prediction. Further Regression concentrates on complicated cases.
  • sklearn.tree. DecisionTreeRegressor
  Decision tree builds regression or classification models in the form of a tree structure. Leaf node (e.g., Hours worked) represents a decision on the numerical target. The root node in a tree which corresponds to the best predictor.
  • pylab
  pylab is a programming environment, built on a set of unofficial python tools and libraries that turns Python into a high-performance scientific computing platform. The name pylab comes in part from the resemblance of the resulting environment to MATLAB.
  • sklearn.metrics
  $R^2$ (determination coefficient) regression score function is used. The optimum score is 1; if the value is negative then it is an unfit model. The value of $Y$ is always predicted by the constant models. $R^2$ score is 0, when the input features are ignored.

5. Experiment results and analysis

Data fit closure on regression line is measured statistically by using $R$-squared Score. $R$-squared Score is also termed as the coefficient of determination, or for multiple regression, it is coefficient of multiple determinations. It is the change of percentage of the target variable given by a linear model. $R$-squared score is in the range of 0 and 1(inclusive).

• The model explains none of the changes of the target data around its mean, if it indicates 0.
• The model explains all the changes of the target data around its mean, if it indicates 1.
Predominantly, the better model that fits data has higher $R$-squared score.

Mathematical Representation of $R$-squared score is:

$$R^2 = 1 - \frac{SSE}{SSR}$$  \hspace{1cm} (4)
\[
SSE = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2
\]

(5)

SST is the sum of squared errors of baseline model.

\[
SST = \sum_{i=1}^{n}(y_i - \bar{y})^2
\]

(6)

TABLE II. Sample Crop Production Dataset [5]

| State_Name                  | District_Name | Crop_Year | Season  | Crop                | Area  | Production |
|-----------------------------|---------------|-----------|---------|---------------------|-------|------------|
| Andaman and Nicobar Islands | Nicobars      | 2000      | Kharif  | Arecanut            | 1254  | 2000       |
| Andhra Pradesh              | Anantapur     | 1997      | Kharif  | Moong(Green Gram)   | 1300  | 500        |
| Andhra Pradesh              | Anantapur     | 1997      | Kharif  | Ragi               | 6700  | 11800      |
| Andhra Pradesh              | Anantapur     | 1997      | Kharif  | Rice               | 35600 | 75400      |
| Karnataka                   | Gadag         | 2005      | Rabi    | Onion              | 266   | 2757       |
| Karnataka                   | Gadag         | 2005      | Rabi    | Other Rabi pulses  | 423   | 57         |
| Karnataka                   | Gadag         | 2005      | Rabi    | Rapeseed &Mustard  | 295   | 78         |
| Tamil Nadu                  | Dharmapuri    | 1997      | Whole   | Potato             | 144   | 3720       |
| Tamil Nadu                  | Dharmapuri    | 1997      | Whole   | Ragi               | 54240 | 106670     |
| West Bengal                 | Purulia       | 2009      | Winter  | Sesamum            | 238   | 87         |
| West Bengal                 | Purulia       | 2010      | Autumn  | Groundnut          | 982   | 1019       |

Regression Models are generated and plotted for random combinations of train and test datasets in following figures Figure.2. to Figure.9.

![AdaBoostRegression Train:80%](image)

Figure. 2: AdaBoost Regression Plot for Training with 80% samples
Figure 3: AdaBoost Regression Plot for Testing with 20% samples

Figure 4: AdaBoost Regression Plot for Training with 70% samples
Figure 5: AdaBoost Regression Plot for Testing with 30% samples

Figure 6: AdaBoost Regression Plot for Training with 60% samples
Figure 7: AdaBoost Regression Plot for Testing with 40% samples

Figure 8: AdaBoost Regression Plot for Training with 50% samples
Figure. 9: AdaBoost Regression Plot for Testing with 50% samples

6. Conclusion
Inspiration driving the examination has been to explore possibility of open and private mists for organizations with light operational burdens. It is viewed as that distributed computing is still seen as another figuring standard anyway it is less mechanically decided sensation yet rather another model to deal with, enhance, offer, buy, and obtainment of data innovation. Comfortable moment mists are in early stages and development stage and there are very conflicting and different conjectures on job that mists will play in changing the fate of business exchanges and communications. Regardless, it has all the earmarks of being very conceivable that the variety in endeavor business is immediate and formative than genuinely radical. Administration providers should be able to help apportionment of cloud to customers by showing how does the customer's entirety cost of data innovation proprietorship lessens or how distributed computing engages new capacities, for instance speedier chance to market of administrations. It was found that each unmistakable cloud organization should have different organizations. The organizations' structures are mediums to decode mists to customer worth.

REFERENCES
[1] G. Eason, B. Noble, and I.N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529-551, April 1955.
[2] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.
[3] I.S. Jacobs and C.P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G.T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271-350.
[4] K. Elissa, “Title of paper if known,” unpublished.
[5] R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.
[6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740-741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].
[7] M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989. P.S. Barreto, H. Y. Kim, B. Lynn, and M. Scott, “Efficient algorithms for pairing-based cryptosystems,” in Proc. Crypto, vol. 2442, Lecture Notes in Computer Science, 2002, pages 354–368. R. Bhatti, J. Joshi,
E. Bertino, and A. Ghafoor, “X-GTRBAC Admin: A decentralized administration model for enterprise-wide access control,” in Proc. ACM SACMAT, 2004, pages 78–86.

[8] M. Blaze, J. Feigenbaum, and A. D. Keromytis, “KeyNote: Trust management for public-key infrastructures,” in Proc. Security Protocols Int. Workshop, 1998, pages 59–63.

[9] M. Blaze, J. Feigenbaum, and J. Lacy, “Decentralized trust management,” in Proc. IEEE Symp. Security Privacy, May 1996, pages 164–173.

[10] D. Boneh and M. K. Franklin, “Identity-based encryption from the Weil pairing,” in Proc. CRYPTO, vol. 2139, LNCS, 2001, pages 213–229.

[11] D. Boneh, C. Gentry, B. Lynn, and H. Shacham, “Aggregate and verifiably encrypted signatures from bilinear maps,” in Proc. Eurocrypt, 2003, pages 416–432.

[12] D. Boneh, C. Gentry, B. Lynn, and H. Shacham, “A survey of two signature aggregation techniques,” CryptoBytes, vol. 6, no. 2, pages 1–9, 2003.

[13] D. Boneh, B. Lynn, and H. Shacham, “Short signatures from the Weil pairing,” in Proc. Asiacrypt, vol. 2248, LNCS, 2001, pages 5 14–532.

[14] D. Clarke, J.-E. Elien, C. Ellison, M. Fredette, A. Morcos, and R. L. Rivest, “Certificate chain discovery in SPKI/SDSI,” J. Comput. Security, vol. 9, no. 4, pages 285–322, Jan. 2001.