Yield prediction

June 4, 2020

1 Machine learning for yield prediction in Ghana

1.1 Ismail Ougamane, Ismail.Ougamane@gmail.com

1.2 Abstract:

The project can be Summaries into 2 main parts, the first part is for us to show the use of machine learning algorithms for yield prediction, in this part, we have formulate the problematic of the project into 3 research questions, then we have present the use of 2 main keywords in this project yield prediction and machine learning, after that we have present the use of machine learning in yield prediction and in the end of this part, I have answers to the 3 research questions.The second part of this project is a case study for yield prediction in Ghana using machine learning algorithms, the process of the conduct of the project consist of four steps the first step is gathering data, the second step is pre-processing the data, the third step is exploring the data and the last is building and choosing the best machine learning model for yield prediction.

1.3 Introduction:

Although machine learning algorithms are frequently used for yield prediction there is no recent study available yet to the best of our knowledge that investigates and summarizes the utilization of machine learning in yield prediction. This study is performed to find answers to the following questions : 1. Research Question 1: How yield prediction is defined and measured? 2. Research Question 2: What are the factors that control yield prediction, and how these could be included in machine learning? 3. Research Question 3: How machine learning methods could be tailored for modelling yield prediction?

Yield prediction in precision farming is considered of high importance for the improvement of crop management, since variations in crop yield from year to year impact international trade, food supply and market prices, this prediction can be useful for policy purposes.

Crop yield measurements or agriculture productivity measurements are important indicators of the productivity and also provide a basis for assessing whether a landscape is supporting the livelihood of individuals who farm the land. Kilograms per hectare is commonly used crop yield measurement, often this standard weighs (kilograms) per area measure (hectare) crop yield is converted from a volumetric unit of measurement that is based on a commonly used container, this method of measuring involve weighing a complete harvest or relying on expert judgement, these two options are very expensive so we use the following two methods are more economical and provide a reasonably accurate assessment of crop yield:

```
Harvesting: Random sample of the crop in a particular field is cut and weighed. Then the
```

```
Framer estimation: We ask farmers for their estimation of the total crop harvested, this
```

This methods has been to be accurate in determining annual or seasonal crop yield, but is not effective for a continuous crop.

Machine learning gives the computer the ability to learn from experiences without being explicitly programmed , machine learning algorithms can be classified into 3 classes :

```
The first class, supervised learning is when the model learns from the labelled data with dire
The second class of machine learning algorithms is unsupervised learning is when we have no la
The third class is semi-supervised learning is used when we have a few labelled data and a lar
```

Before applying machine learning models, we should check 3 criteria' The first criterion is to ask if there is a pattern to be found in our data, even that is difficult to prove, and in all times we work without proving it. The second criterion: we can't pin down the pattern mathematically. The third criterion, we should have data that represent this pattern.

1.4 Machine learning for yield prediction:

Yield prediction is one of the most important topics in precision agriculture, classic yield prediction models use relationships between various factors like meteorological information and soil parameters – and the crop yields, the yield prediction models can be categorise into 3 categories:

1. The goal of Statistical analysis is to map crop productivity to the soil proprieties (cation exchange capacity (CEC), pH, organic matter,... .), to soil characteristics (texture, soil types,...) and with climatic information (rainfall, temperature, radiation from the sun,... .), the statistical analysis used linear and non linear regression analysis to achieve the goal, the statistical methods are generally regarded to be unrealistic for particle purposes.

2. Mechanistic Model simulates the process of carbon assimilation using physical environment factors (like pollution or proximity of toxic sites) and other various environmental factors such as climatic information, management practice and soil character issues. In mechanistic modelling we use the relationship between physical environments and crop productivity also the soil conditions are integrated to simulate crop growth and yield prediction, mechanistic models are advantageous in the interpretability of the results, special case of mechanistic model is the chlorophyll meters in yield prediction that correlated directly chlorophyll content of leaves and yield prediction using SPAD chlorophyll meters.

3. Machine learning models based on sensed data , in the past years, many types of sensors and satellite platforms have been used to gather data for yield prediction, the objective of using machine learning models is to match the data with the yield prediction to do that there are 2 methods:

   ```
   Method 1: Directly correlate the spectral information to crop yield using regression mode
   Method 2: Which estimate various crop parameters such as leaf area index and biomass from
   ```

   Computer based image interpretation is seen as the best methods for analysis and interpretation of remotely sensed images, machine learning algorithms are currently regarded as the key in the development of image interpretation. In general remote sensing system

are widely used in building decision systems or tools, however remote sensing based approaches require processing of the enormous amount of data from different sources using machine learning models, for the error measurement of the models usually depend on the nature and the goal of the project. #### Answers to Research Questions: The answers to research questions are as follows: Research Question 1: How yield prediction is defined and measured?

Answer: Yield prediction is one of the most important topics in precision agriculture, since the yield prediction can impact the international trade, food supply and market prices. For the crop yield measurement we use as unit Kilograms per Hectare and to compute the crop yield there are two economical methods:

```
Harvesting,
Farmer estimation.
```

Research Question 2: What are the factors that control yield prediction, and how these could be included in machine learning?

Answer: The factors that control yield prediction can be categories into 5 categories: 1. Soil proprieties factor: cation exchange caption(CEC), pH, Organic matter, ... 2. Soil characteristic factor: texture, soil type, ... 3. Climatic information: rainfall, temperature, radiation from the sun, ... 4. Physical environment factors: pollution or proximity of toxic sites. 5. Management practice.

All these factors can be included into machine learning models as variables.

Research Question 3: How machine learning methods could be tailored for modelling yield prediction?

Answer: Not only machine learning models can take the factors that control yield prediction as variable, but also can learn from sensed image and satellite images that is seen as the best methods for analysis and interpretation of the results of yield prediction.

1.5 Machine learning for yield prediction in Ghana:

The purpose of our project is to build a machine learning model for yield prediction in Ghana, the process of the project can be divide into 4 steps: 1. Gather the data, 2. Process the data, 3. Explore the data, 4. Build and choose the model the model.

```
In [1]: import numpy as np
        import pandas as pd
        import seaborn as sns
        %matplotlib inline
        import matplotlib.pyplot as plt
        from pandas import read_csv
        import seaborn as sns
        import glob
        import pickle
```

Gather the data: Obtain the data is the core of any machine learning project, in our case we have used the dataset provide by the "Food and Agriculture Organization of the United Nations (FAO)", the dataset contain crop statistics for 173 products in Africa, the Americas, Asia, Europe, and Oceania(for more details see the cells bellow).

```
In [2]: data = pd.read_csv('FAOSTAT_data_5-24-2020.csv')

In [3]: data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8014 entries, 0 to 8013
Data columns (total 14 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   Domain Code       8014 non-null   object
 1   Domain            8014 non-null   object
 2   Area Code         8014 non-null   int64
 3   Area              8014 non-null   object
 4   Element Code      8014 non-null   int64
 5   Element           8014 non-null   object
 6   Item Code         8014 non-null   int64
 7   Item              8014 non-null   object
 8   Year Code         8014 non-null   int64
 9   Year              8014 non-null   int64
 10  Unit              8014 non-null   object
 11  Value             6918 non-null   float64
 12  Flag              5936 non-null   object
 13  Flag Description  8014 non-null   object
dtypes: float64(1), int64(5), object(8)
memory usage: 876.7+ KB


In [4]: data=data[data['Value'].notna()]
        data.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 6918 entries, 0 to 8013
Data columns (total 14 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   Domain Code       6918 non-null   object
 1   Domain            6918 non-null   object
 2   Area Code         6918 non-null   int64
 3   Area              6918 non-null   object
 4   Element Code      6918 non-null   int64
 5   Element           6918 non-null   object
 6   Item Code         6918 non-null   int64
 7   Item              6918 non-null   object
 8   Year Code         6918 non-null   int64
 9   Year              6918 non-null   int64
 10  Unit              6918 non-null   object
 11  Value             6918 non-null   float64
 12  Flag              4840 non-null   object
 13  Flag Description  6918 non-null   object
```

```
dtypes: float64(1), int64(5), object(8)
memory usage: 810.7+ KB
```

```
In [5]: print(data.shape)
        display(data.head())
```

```
(6918, 14)
```

	Domain Code	Domain	Area Code	Area	Element Code	Element
0	QC	Crops	81	Ghana	5312	Area harvested
1	QC	Crops	81	Ghana	5312	Area harvested
2	QC	Crops	81	Ghana	5312	Area harvested
3	QC	Crops	81	Ghana	5312	Area harvested
4	QC	Crops	81	Ghana	5312	Area harvested

	Item Code	Item	Year Code	Year	Unit	Value	Flag	Flag Description
0	572	Avocados	1961	1961	ha	1000.0	F	FAO estimate
1	572	Avocados	1962	1962	ha	1000.0	F	FAO estimate
2	572	Avocados	1963	1963	ha	1000.0	F	FAO estimate
3	572	Avocados	1964	1964	ha	1000.0	F	FAO estimate
4	572	Avocados	1965	1965	ha	1000.0	NaN	Official data

Pre-processing data: After obtaining the data the next immediate thing to do is process the data. This process is for us to clean and filter the data, to do that we have deal with : 1. Missing values, 2. Categorical data.

```
In [6]: data.head()
```

Out[6]:

	Domain Code	Domain	Area Code	Area	Element Code	Element
0	QC	Crops	81	Ghana	5312	Area harvested
1	QC	Crops	81	Ghana	5312	Area harvested
2	QC	Crops	81	Ghana	5312	Area harvested
3	QC	Crops	81	Ghana	5312	Area harvested
4	QC	Crops	81	Ghana	5312	Area harvested

	Item Code	Item	Year Code	Year	Unit	Value	Flag	Flag Description
0	572	Avocados	1961	1961	ha	1000.0	F	FAO estimate
1	572	Avocados	1962	1962	ha	1000.0	F	FAO estimate
2	572	Avocados	1963	1963	ha	1000.0	F	FAO estimate
3	572	Avocados	1964	1964	ha	1000.0	F	FAO estimate
4	572	Avocados	1965	1965	ha	1000.0	NaN	Official data

```
In [7]: data=data.drop(['Year Code','Flag Description',"Flag"],axis=1)
        data.head()
```

```
Out[7]:   Domain Code Domain  Area Code   Area  Element Code         Element  \
        0          QC  Crops         81  Ghana          5312  Area harvested
        1          QC  Crops         81  Ghana          5312  Area harvested
        2          QC  Crops         81  Ghana          5312  Area harvested
        3          QC  Crops         81  Ghana          5312  Area harvested
        4          QC  Crops         81  Ghana          5312  Area harvested

           Item Code      Item  Year Unit   Value
        0        572  Avocados  1961   ha  1000.0
        1        572  Avocados  1962   ha  1000.0
        2        572  Avocados  1963   ha  1000.0
        3        572  Avocados  1964   ha  1000.0
        4        572  Avocados  1965   ha  1000.0

In [8]: data_new=pd.get_dummies(data)

In [9]: data_new.head()

Out[9]:    Area Code  Element Code  Item Code  Year   Value  Domain Code_QC  \
        0         81          5312        572  1961  1000.0               1
        1         81          5312        572  1962  1000.0               1
        2         81          5312        572  1963  1000.0               1
        3         81          5312        572  1964  1000.0               1
        4         81          5312        572  1965  1000.0               1

           Domain_Crops  Area_Ghana  Element_Area harvested  Element_Production  ...  \
        0             1           1                       1                   0  ...
        1             1           1                       1                   0  ...
        2             1           1                       1                   0  ...
        3             1           1                       1                   0  ...
        4             1           1                       1                   0  ...

           Item_Sugar cane  Item_Sweet potatoes  Item_Taro (cocoyam)  \
        0                0                    0                    0
        1                0                    0                    0
        2                0                    0                    0
        3                0                    0                    0
        4                0                    0                    0

           Item_Tobacco, unmanufactured  Item_Tomatoes  Item_Vegetables, fresh nes  \
        0                             0              0                           0
        1                             0              0                           0
        2                             0              0                           0
        3                             0              0                           0
        4                             0              0                           0

           Item_Yams  Unit_ha  Unit_hg/ha  Unit_tonnes
        0          0        1           0            0
```

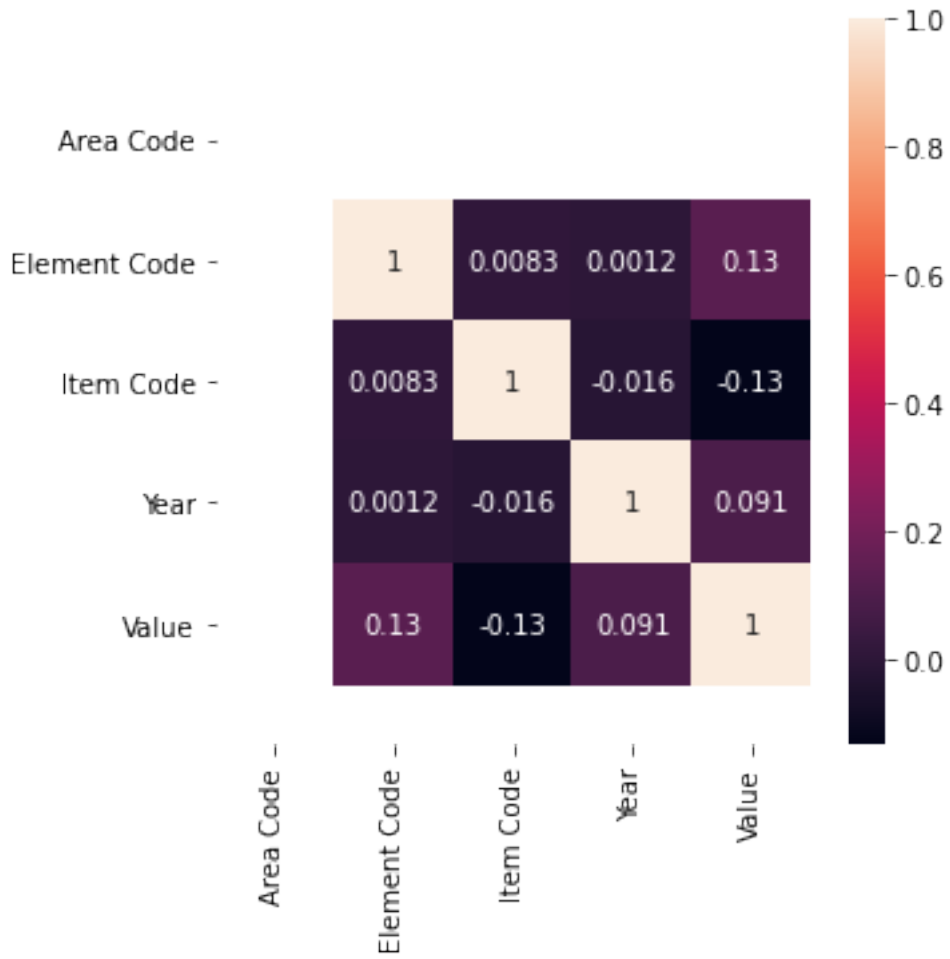
```
1          0        1          0          0
2          0        1          0          0
3          0        1          0          0
4          0        1          0          0

[5 rows x 68 columns]
```

Explore the data: Before jumping into building the model, we will examine the data, first of all we will inspect the data and its properties, then we will use the data visualization to help us to identify significant patterns and trends in our dataset, the scatter-plot below shows a strong positive, non- linear association between the yield estimation and years for items Avocados, banana and beans whether for the dry beans or for the green beans, and for the other items shows a positive linear association between yield estimation and years.

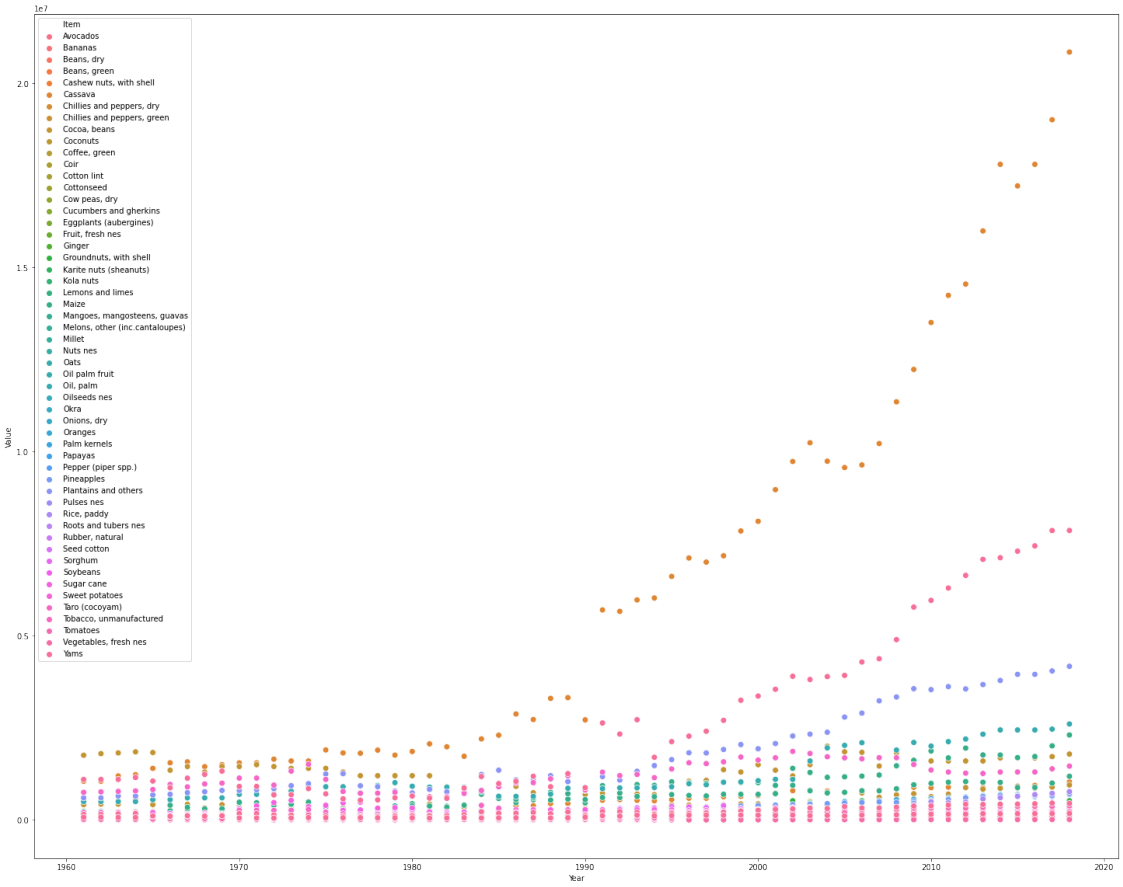
```
In [10]: # Analysis
         f,ax = plt.subplots(figsize = (5,5))
         sns.heatmap(data.corr(), annot=True)

         ## this part is optional. I had to do it because the
         ## plot had been disproportionate.
         bottom, top = ax.get_ylim()
         ax.set_ylim(bottom + 0.5, top - 0.5)
         plt.show()
```



In [11]:
```
# Analysis
f,ax = plt.subplots(figsize = (25,20))
sns.scatterplot(data['Year'],data['Value'],legend='full',data=data,hue=data["Item"],s

## this part is optional. I had to do it because the
## plot had been disproportionate.
# bottom, top = ax.get_ylim()
# ax.set_ylim(bottom + 0.5, top - 0.5)
plt.savefig('foo.png')
plt.show()
```



Build the model: Regression models are one of the most powerful models used to find relations within a dataset, with the key focus being on relationships between the independent variables (predictors) and a dependent variable (outcome). In this phase, we will apply the following models: 1. Linear Regression, 2. Polynomial Regression, 3. Decision Tree Regression.

To validate our models we split the data set into 2 parties train data (70%) and test data (30%), to assess the performance of the models we use R2 metrics: 1. Linear Regression score 0.266, 2. Polynomial Regression score 0.748, 3. Decision Tree Regression with Max Depth =2 score 0.525 4. Decision Tree Regression with Max Depth =5 score 0.938

```
In [12]: from sklearn.model_selection import train_test_split
```

```
In [13]: Y=data_new['Value']
         X=data_new.drop(['Value'],axis=1)
         print(X.shape,Y.shape)
```

```
(6918, 67) (6918,)
```

```
In [14]: x_train, x_test, y_train, y_test = train_test_split(X, Y, test_size=.20, random_state
```

```
In [15]: from sklearn.linear_model import LinearRegression
         from sklearn.model_selection import train_test_split
         from sklearn.preprocessing import PolynomialFeatures
         from sklearn.metrics import r2_score,mean_squared_error
         from sklearn.ensemble import RandomForestRegressor

In [16]: x_train,x_test,y_train,y_test = train_test_split(X,Y, random_state = 0)
         lr = LinearRegression().fit(x_train,y_train)

         y_train_pred = lr.predict(x_train)
         y_test_pred = lr.predict(x_test)

         print(lr.score(x_test,y_test))
```

```
0.26614351105404055
```

```
In [17]: quad = PolynomialFeatures (degree = 2)
         x_quad = quad.fit_transform(X)

         X_train,X_test,Y_train,Y_test = train_test_split(x_quad,Y, random_state = 0)

         plr = LinearRegression().fit(X_train,Y_train)

         Y_train_pred = plr.predict(X_train)
         Y_test_pred = plr.predict(X_test)

         print(plr.score(X_test,Y_test))
```

```
0.748432497967417
```

```
In [18]: import numpy as np
         from sklearn.tree import DecisionTreeRegressor
         import matplotlib.pyplot as plt




         # Fit regression model
         regr_1 = DecisionTreeRegressor(max_depth=2)
         regr_2 = DecisionTreeRegressor(max_depth=5)
         regr_1.fit(x_train, y_train)
         regr_2.fit(x_train, y_train)

         # Predict

         y_1 = regr_1.predict(x_test)
         y_2 = regr_1.predict(x_test)
```

```
y_train_pred = regr_1.predict(x_train)
y_test_pred = regr_1.predict(x_test)

print(regr_1.score(x_test,y_test))



y_1 = regr_2.predict(x_test)
y_2 = regr_2.predict(x_test)


y_train_pred = regr_2.predict(x_train)
y_test_pred = regr_2.predict(x_test)

print(regr_2.score(x_test,y_test))
```

```
0.4380351126851457
0.9361254708944124
```

1.6 Conlusion:

In important issues for precision agriculture purposes is the accurate yield prediction, machine learning models are an essential approach for achieving practical and effective solutions for this problem.

So in our work we show the use of machine learning models in yield prediction, then for more clarification, we have implemented machine learning algorithms for yield prediction in Ghana.