Exploratory analysis of rain fall data in India for agriculture

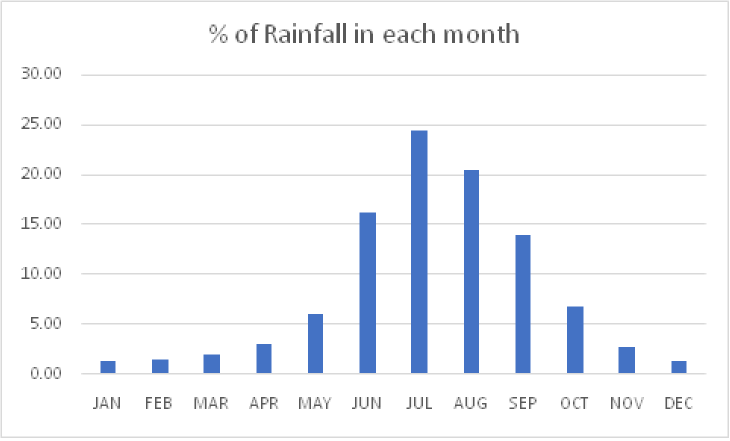
Introduction:

Rainfall has been a major concern these days. Weather conditions have been changing for time being. Rainfall forecasting is important otherwise, it may lead to many disasters. Irregular heavy rainfall may lead to the destruction of crops, heavy floods that can cause harm to human life. It is important to exactly determine the rainfall for effective use of water resources, crop productivity, and pre-planning of water structures.

India is an agricultural country and secondary agro based market will be steady with a good monsoon. The economic growth of each year depends on the amount of duration of monsoon rain, bad monsoon can lead to destruction of some crops, which may result in scarcity of some agricultural products which in turn can cause food inflation, insecurity and public unrest. In our analysis we are trying to understand the behaviour of rainfall in India over the years, by months and different subdivisions.

Overall description:

The below graph shows the percentage of rainfall each month receives when we consider India as a whole. The rainfall in the months of June, July, August and September together contribute to almost 80% of the annual rainfall.



Pre-Requisites:

Frequency analysis

It is a statistical prediction method of studying past events characteristics of a given process (hydrological or other) in order to define the future occurrence probabilities [47]. This prediction is based on the definition and implementation of a frequency model which is an equation describing the statistical behaviour of a process. These models describe the probability of a given value of event. Frequency analysis uses various statistical techniques and is a complex industry that should be treated very rigorously. Its various steps can be summarized very simply according to the diagram The validity of the results of a frequency analysis of the frequency depends on the choice model and especially its type Various avenues can help facilitate this choice but unfortunately there is no universal and infallible method. Several statistical laws such as laws Gumbel Pearson Frechet Goodrich Normal and Log-normal are used in frequency analysis [48]. Those that fit best with the values of the variable used would be retained. From a methodological standpoint this characteristic frequency study begins with the arrangement of the variable values in ascending order by giving each variable rank in the series then by calculating the frequency of return periods extraction and rainy days etc., weather.

The goal through this analysis is to try to categorize years of the study period according to their rainfall. Thus Figure shows the distribution of annual precipitation Bounkiling over the period 1975-2014 through the frequencies calculated by the ln-normal distribution. The analysis of this figure compared with the median frequency value leaves appear two years Class1 representing all wet years and 2 class symbolizing dry years. The analysis of class 1 compared to their average, gives two blocks of wet years: block a that brings together years (2009, 2006, 2012, 2008, 2003, 2011 et 2005) characterized by very strong rainfall

values and block b that brings together years (1982, 1987,1996, 1978 ,1994, 1981, 2004 , 2007 et 2010) characterized by heavy rainfall values. Similarly analysis of Class 2 in relation to their average, gives two blocks of dry years:

characterized by average rainfall values and bloc b that brings together years (1980, 1977, 1983 ,1995, 1992, 1991, 1985, 2014.1979) characterized by low rainfall values. Ultimately the rainfall of Bounkiling

is organized into four blocks.

Project objectives :

Government of India has made a lot of useful data publically available. This includes data for agriculture, infrastructure, technology and so on.

And so we can look at some of this data to see if we There appears to be a very strong corelation between food production for some of the cereals and grains

This corelation appears to be positive as well as negative implying that if we know that one crop had a bumper production, the correlated crop would have a bumper production as well (positive corelation) and if we know that a particular crop failed,then the corelated crop would fail (negative corelation) as well

There also appears to be a fairly strong dependence on rainfall for food productioncan glean something meaningful from them.

In this study here, we look at the agricultural food production for various grains, cereals and oilseeds from the year 2001 to 2017. We have also obtained data for annual rainfall in India for these years as a separate dataset and our goal is to see how strongly agricultural food production depends on annual rainfall.

Earth is surrounded by 70% water, but only 1% can be used for consumption. With the growing issue regarding global warming and climate change, water scarcity and drought is becoming a serious problem and it already exists in some parts of the world.

As the population continuous to grow, so is our demand for water which makes it all the more important to manage and use water wisely.

In some places, rainwater is sometimes the only supply of water for human consumption and other requirements

People have used hills for homes and urban areas for thousands of years. Many people have built their homes and villages on hills to avoid floods. The higher elevation also allows people to defend themselves

SPECIFIC REQUIREMENTS

- License: Free use and redistribution under the terms of the EULA for anaconda distribution.
- Operating system: Windows 8 or newer, 64-bit mac OS 10.13+, or Linux, including Ubuntu, Red Hat, CentOS 7+, and others.
- If your operating system is older than what is currently supported, you can find older versions of the Anaconda installers in our archive that might work for you. See Using Anaconda on older operating systems for version recommendations.

- System architecture: Windows- 64-bit x86, 32-bit x86; Mac OS- 64-bit x86 & M1; Linux- 64-bit x86, 64-bit aarch64 (AWS Graviton2), 64-bit Power8/Power9, s390x (Linux on IBM Z & Linux ONE).
- Minimum 5 GB disk space to download and install.

ANACONDA NAVIGATOR:

Anaconda Navigator is a desktop graphical user interface (GUI) included in Anaconda® distribution that allows you to launch applications and easily manage the packages, environments, and channels without using command-line commands. Navigator can search for packages on Anaconda.org or in a local Anaconda Repository. It is available for Windows, mac OS, and Linux.

ANACONDA JUPYTERLAB

The jupyterlab notebook application allows you to create and edit documents of a python or R language script. Once saved, you can share these files with others.

If Anaconda (conda) and Jupyter Notebook (Jupyter Lab) are set up the right way the combination of them can become the perfect team, where you are able to easily switch between Deep Learning anaconda environments.

Anaconda is a nice package containing a lot of Python packages already and allows for an easy start into the world of Python. Additionally, it allows creating environments in python, which contain different versions of your Python packages. E.g. if a program only runs with Python 2.7 or older versions of Matplotlib, you can create an own workspace for this program and switch back to Python 3 with a click of a button. Furthermore switching between Tensorflow 2.0 and Tensorflow 1.15 becomes easy as well, finally allowing you to switch between versions easily (which can be quite a headache otherwise).

ANACONDA QT CONSOLE

The Qt console is a very lightweight application that largely feels like a terminal, but provides a number of enhancements only possible in a GUI, such as inline figures, proper multi-line editing with syntax highlighting, graphical call tips, and much more. The Qt console can use any Jupyter kernel. The Qt console frontend has hand-coded emacs-style bindings for text navigation.

The Qt Console has the ability to save your current session, as either HTML or XHTML. Your inline figures will be PNG in HTML, or inlined as SVG in XHTML. PNG images have the option to be either in an external folder, as in many browsers' "Webpage, Complete" option, or inlined as well, for a larger, but more portable file.

With the two-process ZMQ model, the frontend does not block input during execution. This means that actions can be taken by the frontend while the Kernel is executing, or even after it crashes. The most basic such command is via 'Ctrl-.', which restarts the kernel. This can be done in the middle of a blocking execution. The frontend can also know, via a heartbeat mechanism, that the kernel has died. This means that the frontend can safely restart the kernel..

Import Necessary Libraries

Pandas is an open source Python package that is most widely used for data science/data analysis and machine learning tasks. It is built on top of another package named Numpy, which provides support for multi-dimensional arrays. As one of the most popular data wrangling packages, Pandas works well with many other data science modules inside the Python ecosystem, and is typically included in every Python distribution, from those that come with your operating system to commercial vendor distributions like ActiveState's ActivePython.

One of the best options for working with tabular data in Python is to use the Python Data Analysis Library (a.k.a. Pandas). The Pandas library provides data structures, produces high quality plots with matplotlib and integrates nicely with other libraries that use NumPy (which is another Python library) arrays.

Remember we can load a library with the import statement and give it a nickname. For example we can import the pandas library using the common nickname pd:

```
import pandas as pd
```

Exploring Our rainfall data

Again, we can use the type function to see what kind of thing rainfall_df is:

```
type(rainfall_df)

<class pandas.core.frame.DataFrame>
```

As expected, it’s a DataFrame (or, to use the full name that Python uses to refer to it internally, a pandas.core.frame.DataFrame).

```
In [82]: rainfall_df.dtypes
Out[82]:
ID                   int64
UT                   int64
year                 int64
month                int64
day                  int64
time                object
raingauges_id        int64
name                object
ward_id              int64
region              object
data               float64
dtype: object
```

Calculating Statistics From Data In A Pandas DataFrame

```
Look at the column names

rainfall_df.columns
```

which **returns**:

```
Index(['ID', 'UT', 'year', 'month', 'day', 'time', 'raingauges_id', 'name',
       'ward_id', 'region', 'data'],
      dtype='object')
```

We can access individual columns of `rainfall_df` by giving the column name in `[]`, e.g. `rainfall_df['year']` only shows the content of column `year`.

Let’s get a list of all the rain gauges in our dataset. The `pd.unique` function tells us all of the unique values in the `name` column.

```
pd.unique(rainfall_df['name'])
```

which **returns**:

```
array(['BLUFF RES NO.3', 'CHATSWORTH RES NO.1', 'CHATSWORTH RES NO.4',
       "CITY ENGINEER'S DEPT", 'CRABTREE S-P-S', 'DUNKELD RES',
       'FIRWOOD RES', 'ISLAND VIEW S-P-S', 'KENNEDY ROAD S-P-S',
       'RIDGE END RES', 'RIDGE VIEW RES', 'SAND PUMP HOPPER',
       'SHERWOOD RES NO.3', 'ST THOMAS RES', 'WENTWORTH RES',
       'WOODLAND RES NO.2', 'DBN NORTH HL RES', 'NEWLANDS RES NO.3',
       'PHOENIX RES NO.1', 'PHOENIX RES NO.4', 'BALLITO', 'HAZELMERE DAM',
       'UMHWWTW', 'CRAWFORD', 'UMH NTH', 'BUFFELS', 'CATO RIDGE',
       'ALVERSTN', 'INANDA DAM', 'WATERFALL', 'HILLCREST', 'SHONGWENI DAM',
       'PINETOWN', 'NAGLE DAM', 'RIVERLEA', 'KLOOF', 'UMLAZI',
       'UMBUMBULU RES', 'ISIPINGO RES', 'UMKDEPOT', 'AMANZIMTOTI',
       'not there'], dtype=object)
```