Sugarcane Data Analysis

Before going through our project, let us know what our objectives are.

Key Objectives:

- Analysing the data,
- Merging two or more data,
- Visualy analysing the data.

Tools used:

- Jupyter Notebook
- Tableau

Introduction \ Overview of the data:

The data is downloaded from http://data.icrisat.org/dld/src/crops.html in the form of csv. I have downloaded the data for yield, harvesting and irrigation specifically for Sugarcane crop.

All the three datas considered in this case are available here:
https://www.kaggle.com/datasets/suhanias/sugarcane-data?select=Crop+wise+irrigation.csv

Using Jupyter Notebook.

Data Preparation:

Let us import the data in jupyter notebook using pandas library.

```
import pandas as pd
sugarcane_yield = pd.read_csv(r'D:\Suhani\Projects\Sugarcane\Sugarcane_yield.csv')
sugarcane_yield.head()
```

	Dist Code	Year	State Code	State Name	Dist Name	SUGARCANE AREA (1000 ha)	SUGARCANE PRODUCTION (1000 tons)	SUGARCANE YIELD (Kg per ha)
0	65	1990	5	Karnataka	Kolar	4.94	46.00	9312.0
1	65	1991	5	Karnataka	Kolar	5.07	48.61	9588.0
2	65	1992	5	Karnataka	Kolar	5.04	35.40	7024.0
3	65	1993	5	Karnataka	Kolar	4.24	44.33	10455.0
4	65	1994	5	Karnataka	Kolar	4.61	37.65	8167.0

Let us drop State name, State code, Dist Code, Sugarcane production features as they are not important in this case.

```
data_1=sugarcane_yield.drop(['SUGARCANE PRODUCTION (1000 tons)', 'State Name', 'State Code', 'Dist Code'],axis=1)
data_1.head()
```

	Year	Dist Name	SUGARCANE AREA (1000 ha)	SUGARCANE YIELD (Kg per ha)
0	1990	Kolar	4.94	9312.0
1	1991	Kolar	5.07	9588.0
2	1992	Kolar	5.04	7024.0
3	1993	Kolar	4.24	10455.0
4	1994	Kolar	4.61	8167.0

```
data_1.tail()
```

	Year	Dist Name	SUGARCANE AREA (1000 ha)	SUGARCANE YIELD (Kg per ha)
725	2013	Yadagiri	1.21	8909.0
726	2014	Yadagiri	0.86	9140.0
727	2015	Yadagiri	0.79	8075.0
728	2016	Yadagiri	0.95	6935.0
729	2017	Yadagiri	1.22	9405.0

Thus, the year ranges from 1990 to 2017.

To know more about the data in python we use *data_name.info* comman.

```
data_1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 730 entries, 0 to 729
Data columns (total 4 columns):
 #   Column                       Non-Null Count  Dtype  
---  ------                       --------------  -----  
 0   Year                         730 non-null    int64  
 1   Dist Name                    730 non-null    object 
 2   SUGARCANE AREA (1000 ha)     730 non-null    float64
 3   SUGARCANE YIELD (Kg per ha)  730 non-null    float64
dtypes: float64(2), int64(1), object(1)
memory usage: 22.9+ KB
```

And when coming to availability of data with respect to district, we can see from the result below that the data is not equally distributed.

```
data_1['Dist Name'].value_counts()
```

```
Kolar               28
Bijapur             28
Bangalore(Urban)    28
Bangalore(Rural)    28
Kodagu              28
Uttara Kannada      28
Tumkur              28
Gulbarga            28
Raichur             28
Bidar               28
Dakshina Kannada    28
Belgaum             28
Bellary             28
Chitradurga         28
Chickmagalur        28
Shimoge             28
Hassan              28
Mandya              28
Mysore              28
Dharwad             28
Haveri              20
Udupi               20
Koppal              20
Bagalkote           20
Gadag               20
Davanagere          20
Chamaraja Nagar     20
Ramanagaram         11
Chikkaballapur      11
Yadagiri             8
Name: Dist Name, dtype: int64
```

Now let us import irrigation and harvest data and run all the codes required.

```
sugarcane_irrigation = pd.read_csv(r'D:\Suhani\Projects\Sugarcane\Crop wise irrigation.csv')
sugarcane_irrigation.head()
```

	Dist Code	Year	State Code	State Name	Dist Name	SUGARCANE IRRIGATED AREA (1000 ha)
0	64	1966	5	Karnataka	Bangalore	3.3
1	64	1967	5	Karnataka	Bangalore	2.5
2	64	1968	5	Karnataka	Bangalore	3.4
3	64	1969	5	Karnataka	Bangalore	3.6
4	64	1970	5	Karnataka	Bangalore	3.8

We will drop Dist Code, State Code and State Name.

```
data_2 = sugarcane_irrigation.drop(['Dist Code','State Code','State Name'],axis=1)
data_2.head()
```

	Year	Dist Name	SUGARCANE IRRIGATED AREA (1000 ha)
0	1966	Bangalore	3.3
1	1967	Bangalore	2.5
2	1968	Bangalore	3.4
3	1969	Bangalore	3.6
4	1970	Bangalore	3.8

```
data_2.tail()
```

	Year	Dist Name	SUGARCANE IRRIGATED AREA (1000 ha)
983	2013	Kodagu / Coorg	0.0
984	2014	Kodagu / Coorg	0.0
985	2015	Kodagu / Coorg	0.0
986	2016	Kodagu / Coorg	0.0
987	2017	Kodagu / Coorg	0.0

The year ranges from 1966 till 2017

```
data_2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 988 entries, 0 to 987
Data columns (total 3 columns):
 #   Column                             Non-Null Count  Dtype
---  ------                             --------------  -----
 0   Year                               988 non-null    int64
 1   Dist Name                          988 non-null    object
 2   SUGARCANE IRRIGATED AREA (1000 ha)  988 non-null    float64
dtypes: float64(1), int64(1), object(1)
memory usage: 23.3+ KB
```

Irrigation data has 988 entries in total.

```
data_2['Dist Name'].value_counts()
```

```
Bangalore                52
Dharwad                  52
Uttara Kannada           52
Dakshina Kannada         52
Gulbarga / Kalaburagi    52
Raichur                  52
Bidar                    52
Bijapur / Vijayapura     52
Belgaum                  52
Bellary                  52
Kolar                    52
Chitradurga              52
Chickmagalur             52
Shimoge                  52
Hassan                   52
Mandya                   52
Mysore                   52
Tumkur                   52
Kodagu / Coorg           52
Name: Dist Name, dtype: int64
```

The data is distributed equally. But the district count in less than the number of districts in yield dataset.

```
sugarcane_harvest = pd.read_csv(r'D:\Suhani\Projects\Sugarcane\Harvest price.csv')
sugarcane_harvest.head()
```

	Dist Code	Year	State Code	State Name	Dist Name	SUGARCANE GUR HARVEST PRICE (Rs per Quintal)
0	64	1966	5	Karnataka	Bangalore	-1.0
1	64	1967	5	Karnataka	Bangalore	-1.0
2	64	1968	5	Karnataka	Bangalore	-1.0
3	64	1969	5	Karnataka	Bangalore	-1.0
4	64	1970	5	Karnataka	Bangalore	-1.0

Let us drop Stae Code, State Name and Dist Code.

```
data_3 = sugarcane_harvest.drop(['Dist Code','State Code', 'State Name'],axis=1)
data_3
```

	Year	Dist Name	SUGARCANE GUR HARVEST PRICE (Rs per Quintal)
0	1966	Bangalore	-1.0
1	1967	Bangalore	-1.0
2	1968	Bangalore	-1.0
3	1969	Bangalore	-1.0
4	1970	Bangalore	-1.0
...	...	...	...
937	2012	Kodagu / Coorg	-1.0
938	2013	Kodagu / Coorg	-1.0
939	2014	Kodagu / Coorg	-1.0
940	2015	Kodagu / Coorg	-1.0
941	2016	Kodagu / Coorg	-1.0

942 rows × 3 columns

The year ranges from 1966 to 2016.

```
data_3.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 942 entries, 0 to 941
Data columns (total 3 columns):
 #   Column                                       Non-Null Count  Dtype
---  ------                                       --------------  -----
 0   Year                                         942 non-null    int64
 1   Dist Name                                    942 non-null    object
 2   SUGARCANE GUR HARVEST PRICE (Rs per Quintal)  942 non-null    float64
dtypes: float64(1), int64(1), object(1)
memory usage: 22.2+ KB
```

The data has 942 entries in total.

```
data_3['Dist Name'].value_counts()
```

```
Bellary                 51
Dharwad                 51
Uttara Kannada          51
Dakshina Kannada        51
Gulbarga / Kalaburagi   51
Raichur                 51
Bidar                   51
Bijapur / Vijayapura    51
Belgaum                 51
Kodagu / Coorg          51
Kolar                   51
Chitradurga             51
Chickmagalur            51
Shimoge                 51
Hassan                  51
Mandya                  51
Mysore                  51
Tumkur                  51
Bangalore               24
Name: Dist Name, dtype: int64
```

The data is equaly distributed among the districts except Bangalore.

Now, let us merge the data. Before doing it, we will have to concentrate on some of the points mentioned below.

Whether the data is available for

- All the districts
- All the year

From the above, we get to know that the district count in irrigation data and harvest data is same. But the harvest data lacks 2017 data.

Where as the yield data has 30 districs and the year ranges from 1990 to 2017, which is very less when compaired. So let us combine the irrigation and harvest data, ignoring 2017 data.

```
data = data_3.merge(data_2, how='left', on = ['Year', 'Dist Name'])
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 942 entries, 0 to 941
Data columns (total 4 columns):
 #   Column                                     Non-Null Count  Dtype
---  ------                                     --------------  -----
 0   Year                                       942 non-null    int64
 1   Dist Name                                  942 non-null    object
 2   SUGARCANE GUR HARVEST PRICE (Rs per Quintal)  942 non-null    float64
 3   SUGARCANE IRRIGATED AREA (1000 ha)         942 non-null    float64
dtypes: float64(2), int64(1), object(1)
memory usage: 36.8+ KB
```

Export the data in the form of csv, so that we can use it further for data visualization using tableau.

Now we have two data, yield data and the merged data, for data visualization.

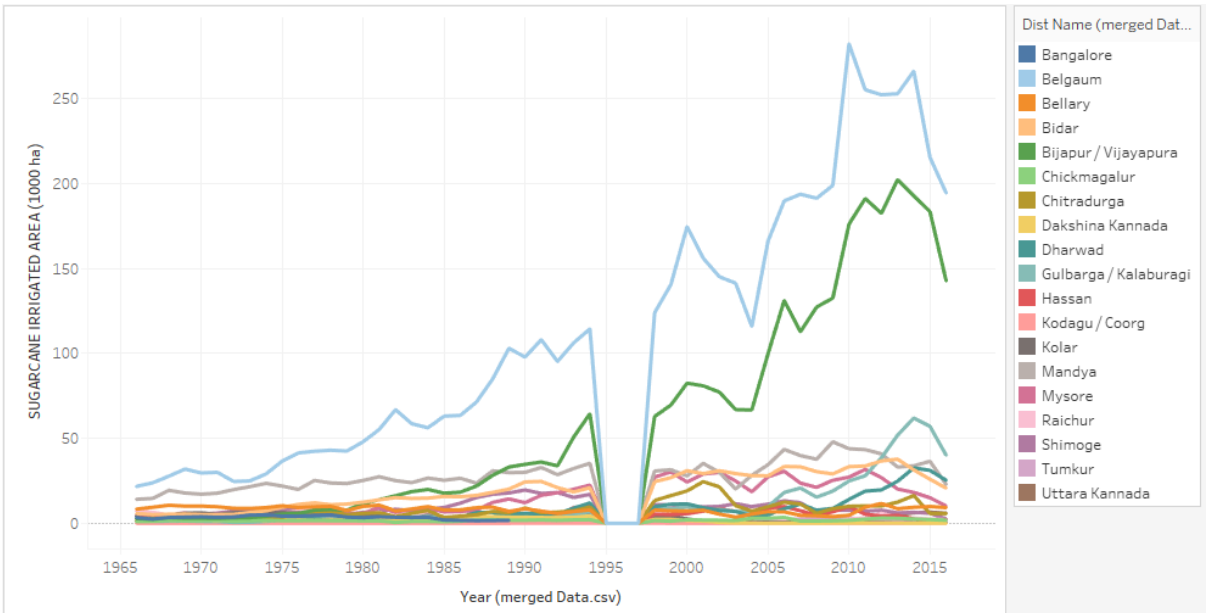
Using Tableau.

Data Visualization:

Import yield data and merge it with the merged data with respect to district name. So that we can use both the data for visualization.

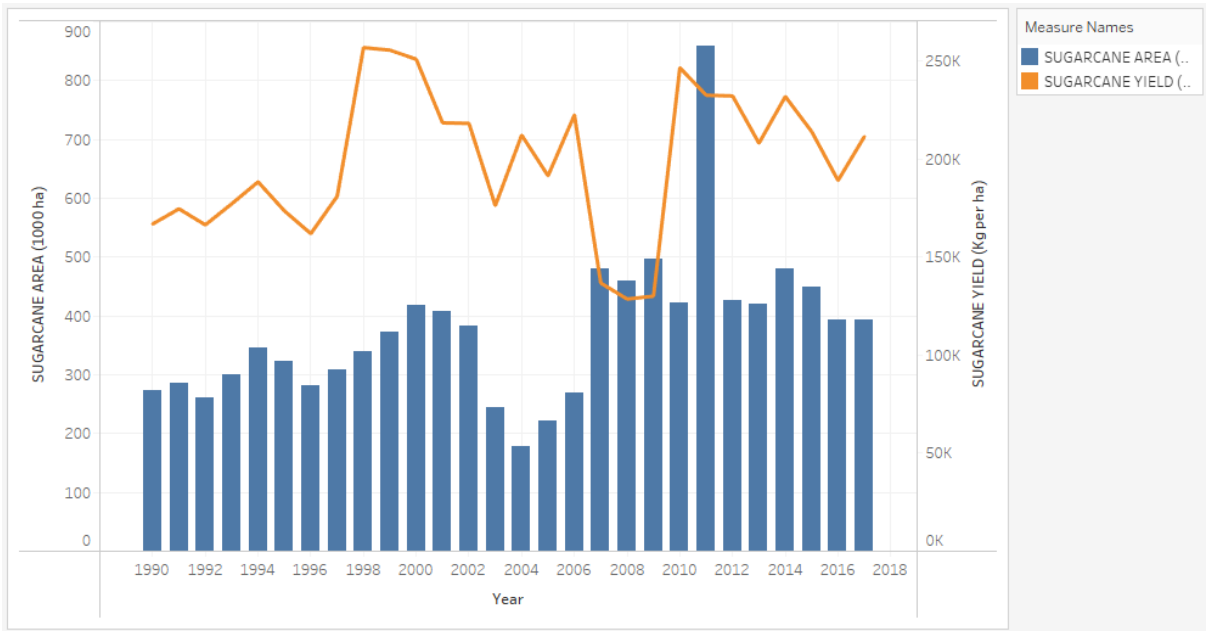
Let’s begin the Visualization.

I have used year and irrigatted area data from merged data.



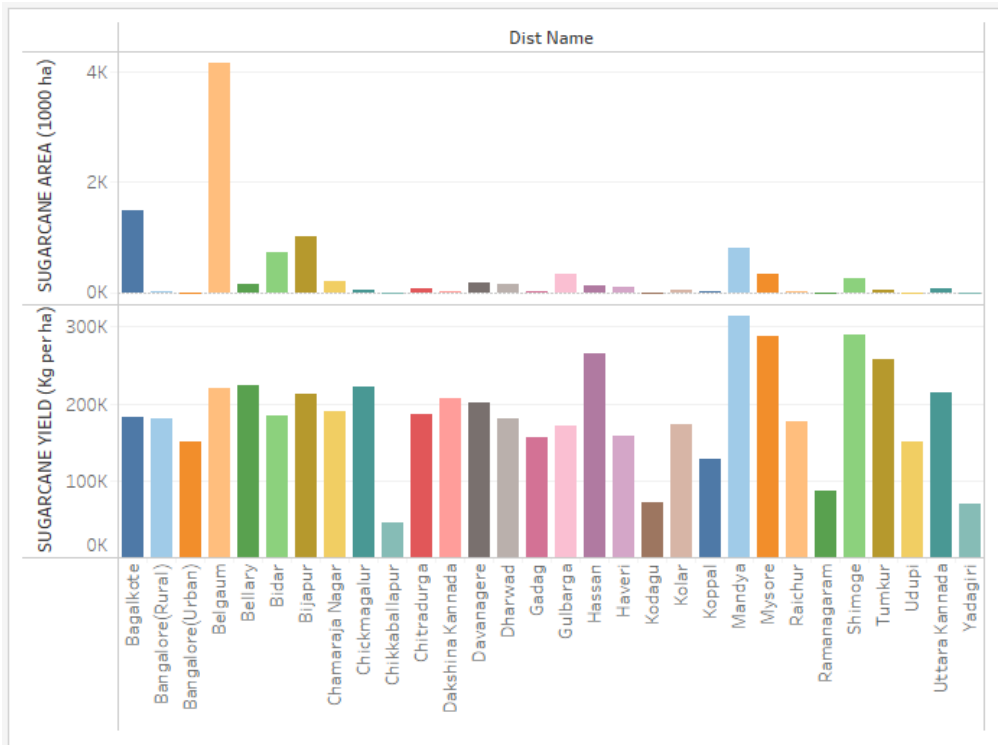
From the above, we get to know that there is a trend at Belgaum and Bijapur. That means, the irrigated area is increasing along the year specifiacally in Belgaum and Bijapur.

The below graph shows the relationship between yield and area used.



We can not give a better conclusion from the above as the complete data is included.

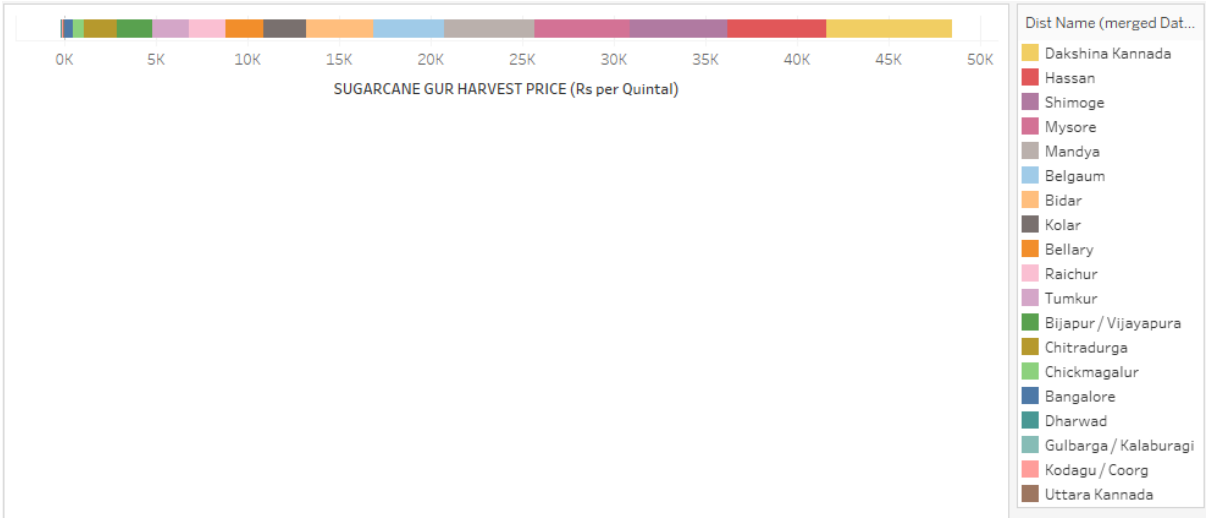
So let us concentrate districs wise.



We can see that, it is Belgaum which is in top when coming to Area but has lesser yield when compaired.

When coming to other districts, though they have lesser area, they are giving greater yield.

When coming to harvest price,



Dakshina Kannada spends more and Bangalore spends the least when compaired to other districts.

So, these are the informations that I could get from the data available.

Conclusion:

Analysing the data and preparing the data as required (Data cleaing is included) would be the prefered task before going for vizualization.