Finding Trends of Food Waste in Countries in Future Years

Sabbir Ahmed

Introduction:

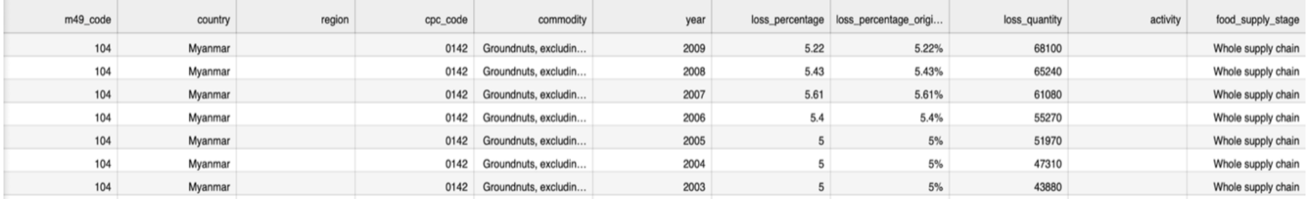
Food waste is now a global problem. Food waste problem considered as world dumbest problem. I want to know about how much food waste happened in past years in country and region. What will be the food waste trends in future years. I want to understand about country and the region about food waste. I will try to implement some machine learning model to find out results about food waste. Before implementing the model, I need to go through OSEMN process. OSEMN process consist of five steps Obtain, Scrub, Explore, Model, iNterpret. The OSEMN process provide a structured approach to data analysis project that helps us obtain meaningful results from analysis.

Data-driven question:

My project data driven question is “finding trends of food waste in countries in future years”. How much food is wasted globally each year and what are the percentages. Food waste impact on environmental, social, and economic. Implement multiple machine learning algorithm to predict future food waste. Implementing various model find out which model gives us the best result. In the results applying different kinds of visualization to clearly understand the results.

Obtain Data:

The first step of OSEMN process is get the right data. Dataset can be found in online and offline. There are several different sources available in online such as Kaggle, UCI Machine Learning Repository, Google Dataset Search, Data.gov and Social media APIs. I found my dataset in Food and Agriculture Organization of the United Nations. [1] In the data set there are more than 29 thousand datapoints. I store the data in .xlsx file.



m49_code	country	region	cpc_code	commodity	year	loss_percentage	loss_percentage_origi...	loss_quantity	activity	food_supply_stage
104	Myanmar		0142	Groundnuts, excludin...	2009	5.22	5.22%	68100		Whole supply chain
104	Myanmar		0142	Groundnuts, excludin...	2008	5.43	5.43%	65240		Whole supply chain
104	Myanmar		0142	Groundnuts, excludin...	2007	5.61	5.61%	61080		Whole supply chain
104	Myanmar		0142	Groundnuts, excludin...	2006	5.4	5.4%	55270		Whole supply chain
104	Myanmar		0142	Groundnuts, excludin...	2005	5	5%	51970		Whole supply chain
104	Myanmar		0142	Groundnuts, excludin...	2004	5	5%	47310		Whole supply chain
104	Myanmar		0142	Groundnuts, excludin...	2003	5	5%	43880		Whole supply chain

Fig: Food waste dataset.

All the data type in this dataset contains intiger, float and string. There are 17 columns in this dataset which are cosiderd as attributes. The attributes are m49_code country region cpc_code commodity year loss_percentage loss_percentage_original loss_quantity “,“activity”, “food_suply_stage “,“treatment”, “cause_of_loss”, “sample_size “,“method_data_collection”, “reference”, “url”.

Example:

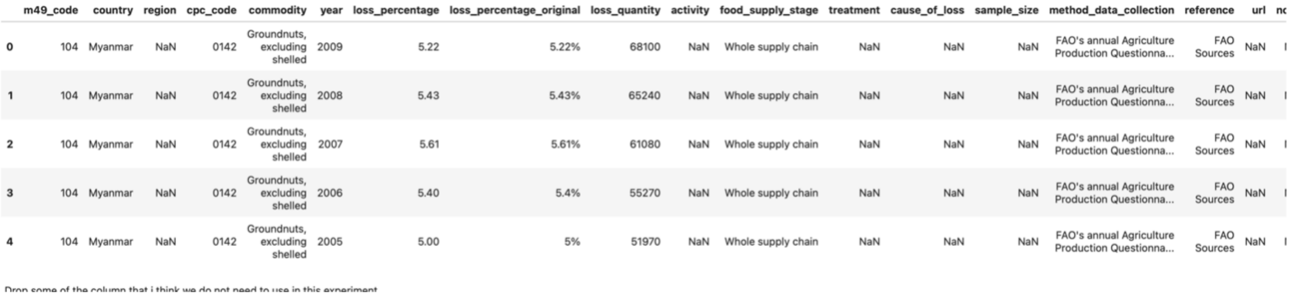
Attribute Name	Value	Data Type	Maximum	Minimum
country	country	String	N/A	N/A
cpc_code	113	integer	N/A	N/A
year	2017	Date	2000	2021
loss_percentage	4.55	Float	7.5	1.0

Table: Example of Data Type in Data Set.

There are mainly two types of variable Qualitive Data and Quantitate Data. In quantitative data there are two types of data present in this data set Discrete and continuous. Continuous data is loss percentage. The example of qualitative data is country name. There are some discrete data too like year. This data is collected from around the world in long time span. The whole data set is one of the largest data set about food waste. There are some missing data presents in the dataset. The dataset version is the march 2020.

Scrub the dataset:

In this OSEMN process we would like to scrub the data. Scrubbing the means cleaning the data from missing value, handling outliers and transform the data in a organize format that we can analyze. Sometimes we need to remove whole attribute column. If we do not need those columns, we can remove it and only work with what we need. I did not find any outliers in my data. The initial data set given below



	m49_code	country	region	cpc_code	commodity	year	loss_percentage	loss_percentage_original	loss_quantity	activity	food_supply_stage	treatment	cause_of_loss	sample_size	method_data_collection	reference	url	nc
0	104	Myanmar	NaN	0142	Groundnuts, excluding shelled	2009	5.22	5.22%	68100	NaN	Whole supply chain	NaN	NaN	NaN	FAO's annual Agriculture Production Questionna...	FAO Sources	NaN	
1	104	Myanmar	NaN	0142	Groundnuts, excluding shelled	2008	5.43	5.43%	65240	NaN	Whole supply chain	NaN	NaN	NaN	FAO's annual Agriculture Production Questionna...	FAO Sources	NaN	
2	104	Myanmar	NaN	0142	Groundnuts, excluding shelled	2007	5.61	5.61%	61080	NaN	Whole supply chain	NaN	NaN	NaN	FAO's annual Agriculture Production Questionna...	FAO Sources	NaN	
3	104	Myanmar	NaN	0142	Groundnuts, excluding shelled	2006	5.40	5.4%	55270	NaN	Whole supply chain	NaN	NaN	NaN	FAO's annual Agriculture Production Questionna...	FAO Sources	NaN	
4	104	Myanmar	NaN	0142	Groundnuts, excluding shelled	2005	5.00	5%	51970	NaN	Whole supply chain	NaN	NaN	NaN	FAO's annual Agriculture Production Questionna...	FAO Sources	NaN	

Drop some of the column that i think we do not need to use in this experiment

Fig: Initial data set.

We read the data set by

df = pd.read_csv('Data-3.csv')

df.head()

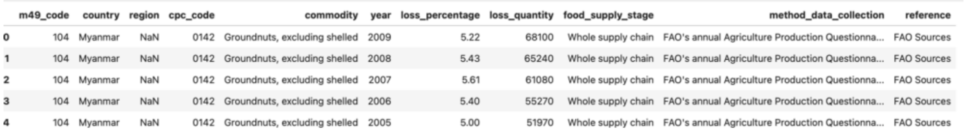
The "df" parameter represent the whole data set. We need to remove some attribute column. Because I think all the attribute is not necessary for analyzing the data set.

We drop some column by using this line of code:

d2=df.drop(['loss_percentage_original','activity','treatment','cause_of_loss','sample_size','url','notes'],axis='columns')

d2.head()

Now the dataset represents by "d2". In new dataset we have less attribute than first dataset.

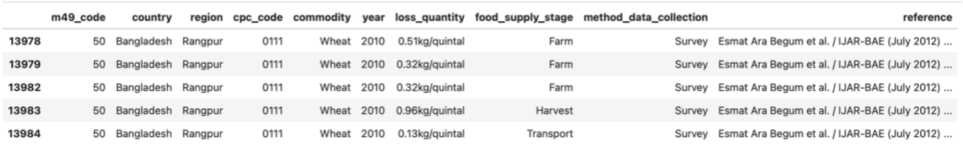


	m49_code	country	region	cpc_code	commodity	year	loss_percentage	loss_quantity	food_supply_stage	method_data_collection	reference
0	104	Myanmar	NaN	0142	Groundnuts, excluding shelled	2009	5.22	68100	Whole supply chain	FAO's annual Agriculture Production Questionna...	FAO Sources
1	104	Myanmar	NaN	0142	Groundnuts, excluding shelled	2008	5.43	65240	Whole supply chain	FAO's annual Agriculture Production Questionna...	FAO Sources
2	104	Myanmar	NaN	0142	Groundnuts, excluding shelled	2007	5.61	61080	Whole supply chain	FAO's annual Agriculture Production Questionna...	FAO Sources
3	104	Myanmar	NaN	0142	Groundnuts, excluding shelled	2006	5.40	55270	Whole supply chain	FAO's annual Agriculture Production Questionna...	FAO Sources
4	104	Myanmar	NaN	0142	Groundnuts, excluding shelled	2005	5.00	51970	Whole supply chain	FAO's annual Agriculture Production Questionna...	FAO Sources

Fig: After dropping some attribute.

We can drop more attribute we want. Now we will drop the loss_percentage column.

X = df.drop(['loss_percentage'],axis='columns')



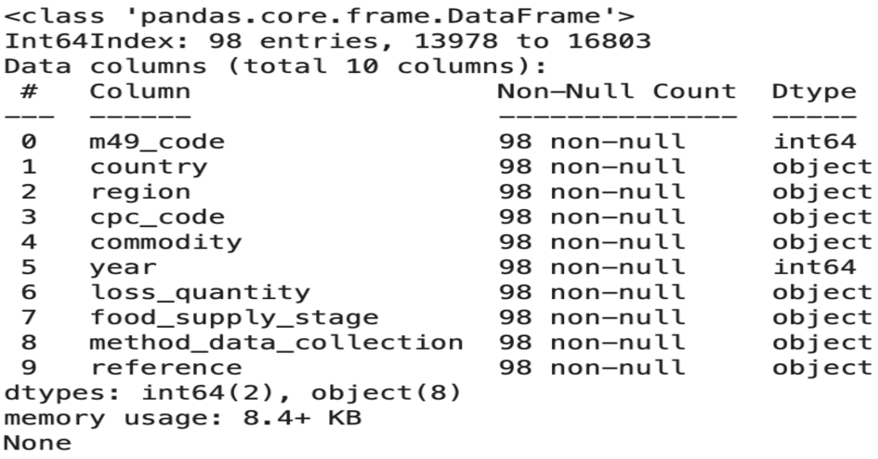
	m49_code	country	region	cpc_code	commodity	year	loss_quantity	food_supply_stage	method_data_collection	reference
13978	50	Bangladesh	Rangpur	0111	Wheat	2010	0.51kg/quintal	Farm	Survey	Esmat Ara Begum et al. / IJAR-BAE (July 2012) ...
13979	50	Bangladesh	Rangpur	0111	Wheat	2010	0.32kg/quintal	Farm	Survey	Esmat Ara Begum et al. / IJAR-BAE (July 2012) ...
13982	50	Bangladesh	Rangpur	0111	Wheat	2010	0.32kg/quintal	Farm	Survey	Esmat Ara Begum et al. / IJAR-BAE (July 2012) ...
13983	50	Bangladesh	Rangpur	0111	Wheat	2010	0.96kg/quintal	Harvest	Survey	Esmat Ara Begum et al. / IJAR-BAE (July 2012) ...
13984	50	Bangladesh	Rangpur	0111	Wheat	2010	0.13kg/quintal	Transport	Survey	Esmat Ara Begum et al. / IJAR-BAE (July 2012) ...

Fig: X dataset.

Now we need to encode and decode the data because machine can not read the string value.

Now we will work on 10 attribute.we can print our dataset information.

print(X.info())



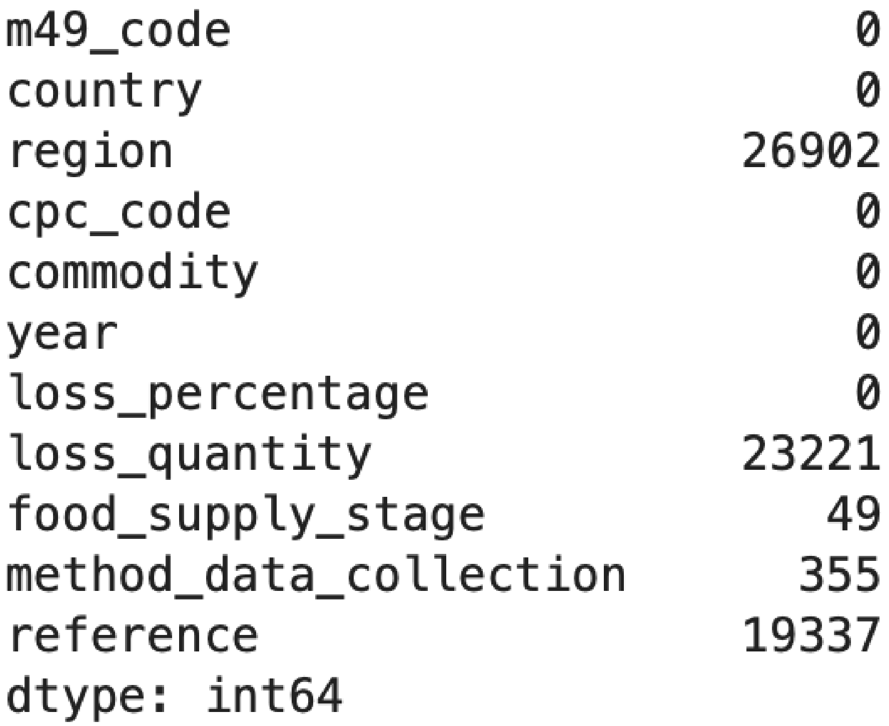
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 98 entries, 13978 to 16803
Data columns (total 10 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   m49_code                98 non-null     int64
 1   country                 98 non-null     object
 2   region                  98 non-null     object
 3   cpc_code                98 non-null     object
 4   commodity               98 non-null     object
 5   year                    98 non-null     int64
 6   loss_quantity           98 non-null     object
 7   food_supply_stage       98 non-null     object
 8   method_data_collection  98 non-null     object
 9   reference               98 non-null     object
dtypes: int64(2), object(8)
memory usage: 8.4+ KB
None
```

Fig: Information about X data set.

From the data set information, we can see there are no null value present in the data set. Because previously we find out all the null value and erase them from data.

df.isna().sum()

is a pandas DataFrame method that will show the number of missing values.



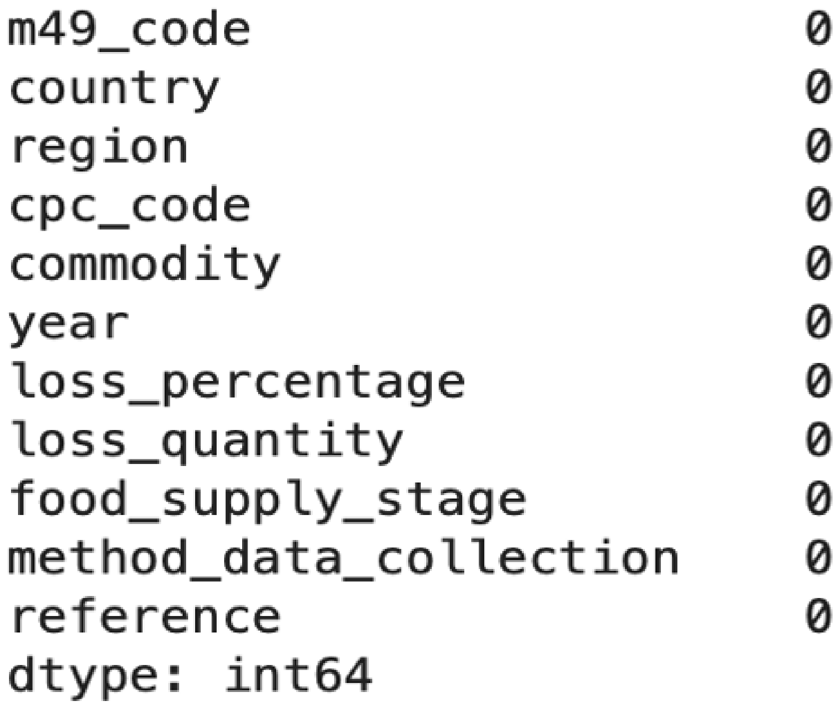
```
m49_code                      0
country                       0
region                    26902
cpc_code                      0
commodity                     0
year                          0
loss_percentage               0
loss_quantity             23221
food_supply_stage            49
method_data_collection      355
reference                 19337
dtype: int64
```

Now we can find out all the missing value and can remove all the rows.

df=df.dropna()

df.isnull().sum()

now we will get a clean dataset.



```
m49_code                  0
country                   0
region                    0
cpc_code                  0
commodity                 0
year                      0
loss_percentage           0
loss_quantity             0
food_supply_stage         0
method_data_collection    0
reference                 0
dtype: int64
```

Fig: showing null values in all attributes.

Machine cannot read the categorical value that is why we need to encode and decode the string value for used as input to machine learning algorithms need to careful about using label encoding, as it may introduce biases into the data.

cols = ['m49_code','country', 'region', 'cpc_code', 'commodity','year', 'loss_quantity', 'food_supply_stage', 'method_data_collection', 'reference']

X[cols] = X[cols].apply(LabelEncoder().fit_transform)

Explore:

In this OSEMN process we explore the data through statistical analysis. We create function to see the relationship between attributes. We visualize the data using different visualization. Visualizing data can help us identifying pattern and relation in the data. We can find out potential variables that may be important for analysis.

In my food waste data set we can see the country, region, commodity, year, loss percentage attributes. By visualizing those data, we can find out interesting fact about food waste. Finding the relation between those data then we can find out the trends of food waste in countries in future.

I select the countries region with loss percentage to know how much food waste happened. For visualizing we use bar chat. Bar char can compare value, showing trends and highlight difference.

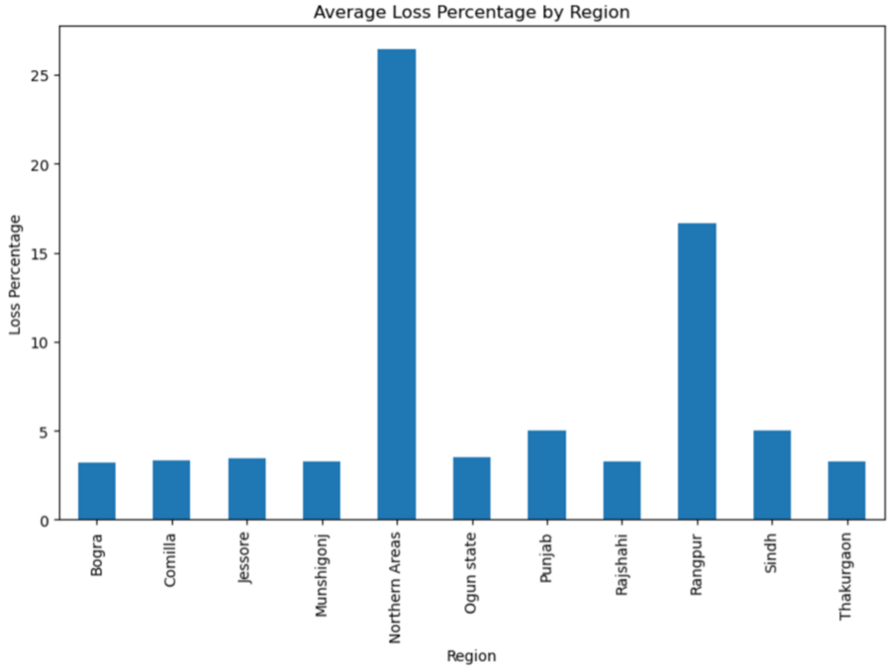


Fig: Loss percentage and region.

We can see in Northern Areas are food waste much more than another region. Most of the region food waste are similar.

Then I try to find out loss quantity, loss percentages and year by a scatter plot.

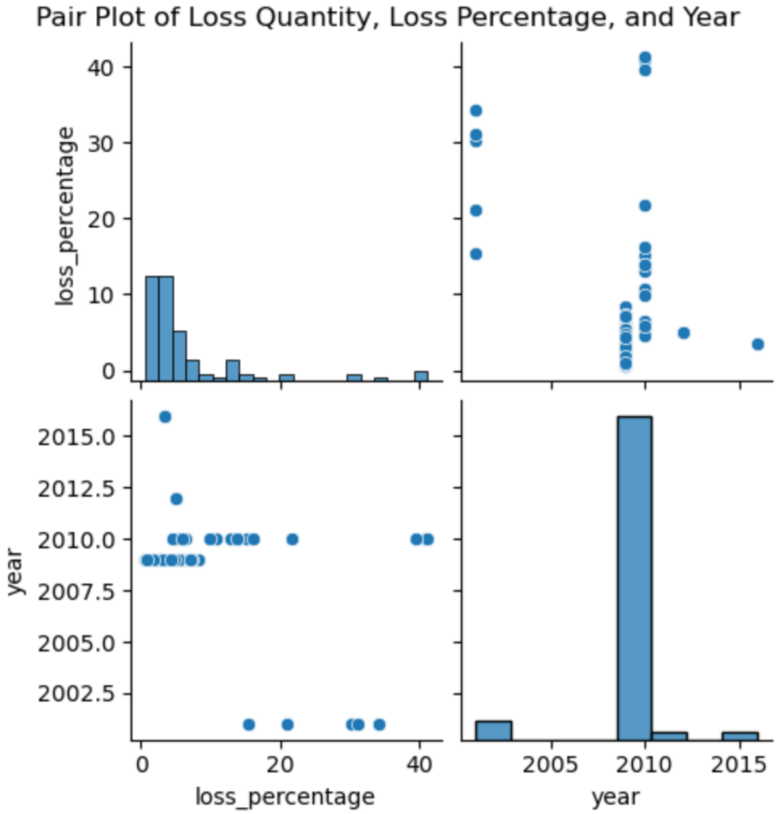


Fig: Scatter plot of loss percentage and year.

Now we can find how the data was collected. The collection happened survey, case study, controlled Experiment, no data collection specified.

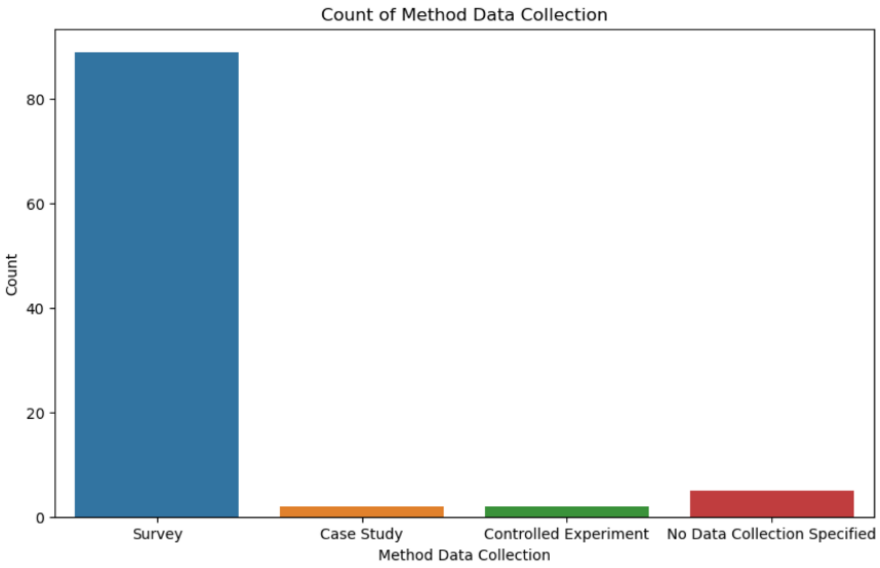
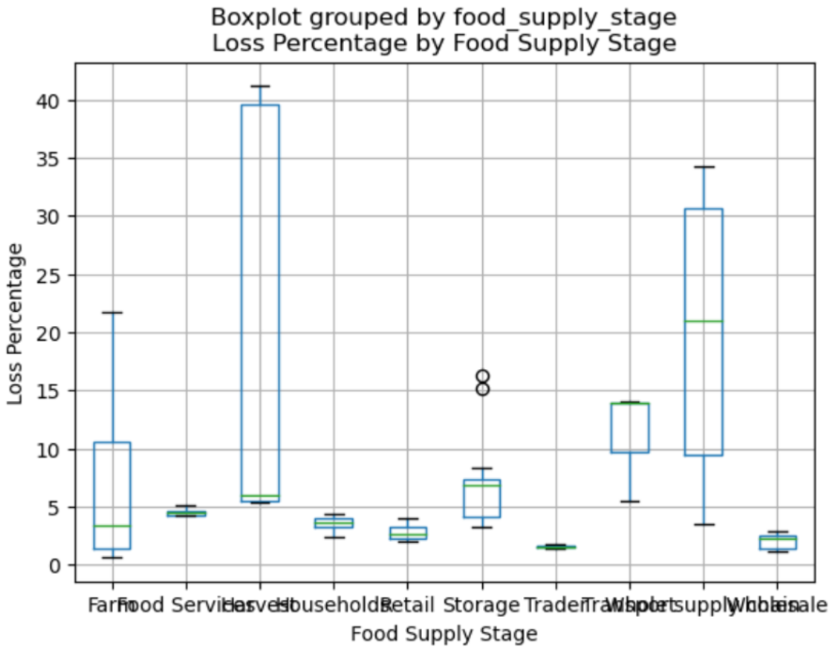


Fig: Collection of data.

From the bar char we can learn that most of the data collected by survey. Case study, controlled experiment and no data collection specified is almost same.

We can use box plot to find out loss percentage in supply stage.



In the box plot we can learn that most of the food being waste in storage and while supplying.

Model:

In the model stage we build statistical and machine learning models to analyze the data and make prediction. We can use various kind of model. In machine learning there are different types of learning supervised learning, unsupervised learning, semi supervised and reinforcement. According to our data we have label data. Supervised learning will fit our data. In supervised learning there are several models available for machine learning Linear regression, bayes theorem, Lasso, decision tree, support vector machine, Neural networks, nearest neighbor etc.

I used my dataset three algorithm

1.Linear regression

2.Random Forest

3. Decision tree

1.Linear regression is statistical method that used in supervised learning to find out the relationship between two data point. Linear regression used in for determine the strength between depend variable and independent variable. In our data set there are several data point exist we need to find out the relationship between all those datapoint to predict the future trend of food waste in various countries.

```
# Linear Regression

linear_model = LinearRegression()

linear_model.fit(X, y)

linear_pred = linear_model.predict(X)

linear_model.score(X, y)

linear_mae = metrics.mean_absolute_error(y, linear_pred)

linear_mse = metrics.mean_squared_error(y, linear_pred)

print('Linear Regression:')

print('MAE (Mean Absolute Error):', linear_mae)

print('MSE (Mean Squared Error):', linear_mse)
```

the output is

```
Linear Regression:MAE (Mean Absolute Error):
2.905993594581O897MSE (Mean Squared Error):
27.876125986505905
```

MAE is the average distance between predicted data and real data and MSE measure the average square difference between actual value and estimated value. Now we want to find out linear regression model accuracy.

```
linear_model.score(X, y)#linear
```

```
0.6315498089740754
```

The model accuracy is 63%. Which is moderately good.

2. Random Forest is combination of multiple decision tree’s average value. Random forest is a good algorithm for supervised learning. For food waste data set it can be a good choice. Applying random forest in food waste dataset.

```
# Random Forest

rf_model = RandomForestRegressor()

rf_model.fit(X, y)

rf_pred = rf_model.predict(X)

rf_mae = metrics.mean_absolute_error(y, rf_pred)

rf_mse = metrics.mean_squared_error(y, rf_pred)

print('Random Forest:')

print('MAE (Mean Absolute Error):', rf_mae)

print('MSE (Mean Squared Error):', rf_mse)
```

```
Random Forest:MAE (Mean Absolute Error):
0.6432285714285725MSE (Mean Squared Error):
1.787880884897969
```

Random forest is giving less error than linear regression. The MAE and MSE is less than linear regression. Now we need to find out accuracy of random forest.

```
rf_model.score(X, y)#random
```

```
0.9727223888501788
```

The random forest giving us 97% accuracy witch is good model compare to linear regression.

3. Now we can try another machine learning model which is decision tree. Decision tree is supervised algorithm. Our dataset is large if we use decision tree for this food waste dataset it may be overfit and cannot give us accurate result.

```
# Decision Tree
dt_model = DecisionTreeRegressor()
dt_model.fit(X, y)
dt_pred = dt_model.predict(X)

dt_mae = metrics.mean_absolute_error(y, dt_pred)
dt_mse = metrics.mean_squared_error(y, dt_pred)
print('Decision Tree:')
print('MAE (Mean Absolute Error):', dt_mae)
print('MSE (Mean Squared Error):', dt_mse)
```

```
Decision Tree:MAE (Mean Absolute Error): 0.0MSE (Mean
Squared Error): 0.0
```

Decision gives us no MAE and MSE error. It will give us 100% accurate result. But we know that no machine learning algorithm can give us 100% perfect result. So, we can avoid decision tree.

Interpret:

In the conclusion we can say we need a better data set to predict more accurately about food waste trends in future in countries. But we get a good result using random forest. I need to work more on that project to predict more accurately.

Reference:

1. Food and Agriculture Organization of the United Nations. Technical platform on the measurement and reduction of food loss and waste. . 2022.

>