Addressing World Hunger: Loss & Waste of Cereals and Pulses

WGU – Data Analytics Capstone

Michael Lawson
Western Governors University
D214 – Data Analytics Capstone
PA Task 1: DATA ANALYTICS REPORT AND EXECUTIVE SUMMARY
Daniel Smith
10/20/2022

Michael C. Lawson | Student ID: #001479557 | Program: MS Data Analytics (5/1/21) |
Program Mentor: Emil Stoica | 859-265-1888 (EST) | mlaws77@wgu.edu

Addressing World Hunger: Loss & Waste of Cereals and Pulses

Research Question

The Food and Agricultural Organization (FAO) of the United Nations is convinced that hunger and malnutrition can be eradicated in our lifetime (FAO, 2015). Several factors impact the lack of global food security including politics, stability of governments, social unrest, food prices, natural disaster, and many other factors. One way the FAO is actively working to end hunger is by promoting open data that is shared freely to allow anyone with the interest or ability to analyze the data to find answers for where and why world hunger and malnutrition still exists, and what measures can be taken to eradicate the issue. Some great things are already happening around the world where food is being given away rather than thrown away when there is overstock. While this capstone was being written, over 300,000 avocados were handed out in Philadelphia, Pennsylvania by the nonprofit group Sharing Excess (Conde, 2022). The data provided by the FAO is vast because it covers many different topics related to tracking food and agriculture and the indicators of hunger and malnutrition worldwide. The capstone author selected this topic because of personal experience volunteering in Haiti, Honduras, Eastern Kentucky Appalachia and other areas affected by hunger. This will be an exploratory data analysis (EDA) investigating the data for patterns and anomalies to test a specific hypothesis given below.

Question: Is there a statistically significant difference in the percentage of food loss between the farm, harvest, and retail food supply stages?

Cereals and pulses were selected as the focus of this analysis as they are staple foods feeding every population no matter the geographic location, economic stability or social environment. This analysis will seek to discover whether statistically significant differences of the percentage of food loss exist within the data shared by the FAO related to food waste and loss. The goal of this analysis will be to help stakeholders identify problems and create a plan of action that will reduce loss and allow more food to be available to reach populations most affected by hunger and malnutrition. According to FAO, "our food systems cannot be resilient if they are not sustainable, hence the need to focus on the adoption of integrated approaches designed to reduce food loss and waste. Actions are required globally and locally to maximize the use of the food we produce" (FAO, 2022).

Hypothesis:

H_0: The percentage of food loss does NOT have statistically significant differences between the food supply stages 'Farm', 'Harvest' and 'Retail'.

H_A: The percentage of food loss does have statistically significant differences between the food supply stages 'Farm', 'Harvest' and 'Retail'.

Data Collection

Data will be gathered for this analysis from the Food Loss and Waste database found on the FAO website, fao.org. The United Nations (UN) is a trusted entity worldwide and the FAO is a specialized agency of the UN focused on ending world hunger. The data provided by this organization can be trusted as true and accurate because it has been collected and provided by the FAO. The FAO website provides a data extraction tool for the Food Loss and Waste Database among many other databases (FAO, n.d.). This extraction tool allows a user to create visualizations using various features and gives the option to download specific data to a comma separated value (.csv) file. The tool allows the user to choose among options to limit the downloaded data to the value chain stage, commodity or basket commodity items, the country or world region. As mentioned before, the basket commodities cereals and pulses were selected for this analysis, and all other options of the tool were left to select "All" options available to have as much data as possible related to the loss and waste of cereals and pulses.

Data Extraction and Preparation

The following steps will be taken in Jupyter Notebook:

Cleaning & prep:
1. Load Python libraries appropriate for data visualization and regression
2. Load the data using read_csv()
3. Examine the header to see what column names and the values they contain using .head()
4. Examine the shape, dtype, and all column names using .info()
5. Create the reduced data set
6. If null values exist, treat them.
7. Combine categories of 'food_supply_stage' into broader categories: 'Farm', 'Harvest' & 'Retail'
8. Select a sample of the same size from each group
9. Visualize the distribution of the target feature 'loss_percentage'
10. Normalize 'loss_percentage'

Analysis:
11. Perform 2-Way ANOVA
12. Use Tukey method to determine which pairs are significantly different
13. Create boxplots to compare means and variability of loss percentage
14. Create Q-Q plot to test normality of data (since ANOVA assumes normality)
15. Check for equality of variances of the treatments using the Levene test
16. Export clean data set

The Food Loss and Waste Database data collection tool provided by the FAO website was used to create a comma separated value file that was downloaded to a hard drive. Jupyter Notebook was used to run the Python programming language, and the read_csv tool of the Pandas library for Python was used to extract the data from the csv file to a data frame in Jupyter Notebook. The following is a list of all Python packages and libraries used for this analysis:

Library	***Package***	***Notes***
Pandas		*fast and powerful data analysis and manipulation library.*
Numpy		*wide range of math functions*
statsmodels	*Formula, stats*	*Python module that provides classes and functions for the estimation of many different statistical models*
Matplotlib	*pylab*	*convenience module that bulk imports matplotlib.pyplot (for plotting) and NumPy (for Mathematics and working with arrays) in a single name space (Tutorialspoint, n.d.)*
Matplotlib	*pyplot*	*visualizations of data*
seaborn		*visualizations of data*
scipy		*equations and algorithms*
statsmodels		*provides classes and functions for the estimation of many different statistical models*
bioinfokit	*analys*	*easy-to-use functionalities to analyze, visualize, and interpret the biological data*
warnings	*filterwarnings*	*loaded to remove filter warnings*

After the data was loaded to a data frame, exploration began. The initial dataset as it was downloaded from the FAO had 19,329 rows and 18 features. The initial dataset included a column for the year, a continuous variable for food loss percentage, two numerical columns that indexed the countries and commodities and several categorical columns. Several of the columns were over 98% null, and the numerical columns with codes for country and commodity are categorical, so their numerical values have no statistical meaning for analysis. Two columns were chosen from the 18 to eliminate most nulls and select the data directly related to answering the question. The columns for 'region', 'loss_quantity', 'treatment', 'cause_of_loss', 'sample_size', 'reference', and 'notes' were eliminated because those columns were around 98% empty, and not useful for this analysis. The columns 'm49_code' and 'cpc_code' were eliminated because even though values are numerical, the data is still categorical in nature as they directly correlate with 'country' and 'commodity'. Rather than lose data by dropping it from the data frame, a new data frame was created by selecting the columns that are useful to the analysis. The features selected for this analysis are:

column	data type	sample data
loss_percentage	Continuous	"3.50", "4.87", "2.50", "4.43", "4.00", "1.1"…
food_supply_stage	Categorical	"Farm", "Harvest", "Transport", "Storage"…

In [1]:
```
# import all possible packages useful for multiple linear regression
import pandas as pd
import numpy as np
from pandas import Series, DataFrame
import matplotlib.pyplot as plt
plt.rc("font", size = 14)
%matplotlib inline
import seaborn as sns
from scipy import stats
sns.set(style="white")
sns.set(style="whitegrid", color_codes=True)
import statsmodels.api as sm
import statsmodels.formula.api as smf
from statsmodels.formula.api import ols
from statsmodels.stats.anova import anova_lm
import statsmodels.stats.multicomp as multi
import warnings
warnings.filterwarnings('ignore')
```

In [2]:
```
#Import data set from hard drive
food_df = pd.read_csv(r"C:\Users\mlaws\OneDrive - Western Governors University\Documents\WGU\D214\world_food_waste.csv", skiprows=0, delimiter=",")
```

In [3]:
```
# View the # of rows/columns
food_df.shape
```

Out[3]: (19329, 18)

In [4]:
```
# View data types and null counts
food_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 19329 entries, 0 to 19328
Data columns (total 18 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   m49_code                19329 non-null  int64
 1   country                 19329 non-null  object
 2   region                  455 non-null    object
 3   cpc_code                19329 non-null  float64
 4   commodity               19329 non-null  object
 5   year                    19329 non-null  int64
 6   loss_percentage         19329 non-null  float64
 7   loss_percentage_original 19329 non-null  object
 8   loss_quantity           89 non-null     object
 9   activity                18928 non-null  object
 10  food_supply_stage       19309 non-null  object
 11  treatment               476 non-null    object
 12  cause_of_loss           291 non-null    object
 13  sample_size             402 non-null    object
 14  method_data_collection  19229 non-null  object
 15  reference               1282 non-null   object
 16  url                     18489 non-null  object
 17  notes                   492 non-null    object
dtypes: float64(2), int64(2), object(14)
memory usage: 2.7+ MB
```

In [5]:
```
#  view data
food_df.head()
```

Out[5]:

	m49_code	country	region	cpc_code	commodity	year	loss_percentage	loss_percentage_original	loss_quantity	activity	food_supply_stage	treatment	cause_of_loss
0	108	Burundi	NaN	111.0	Wheat	2020	3.50	3.5	NaN	Shelling, Threshing	Farm	NaN	NaN
1	108	Burundi	NaN	111.0	Wheat	2020	4.87	4.87	NaN	Storage	Farm	NaN	NaN
2	108	Burundi	NaN	111.0	Wheat	2020	2.50	2.5	NaN	Transportation	Farm	NaN	NaN
3	108	Burundi	NaN	111.0	Wheat	2020	4.43	4.43	NaN	Drying, Harvesting	Harvest	NaN	NaN
4	108	Burundi	NaN	112.0	Maize (corn)	2020	4.00	4	NaN	Drying	Farm	NaN	NaN

In [6]:
```
data = food_df[['food_supply_stage','loss_percentage']]
```

The column ‘food_supply_stage’ has 20 null cells according to the readout above. Since the number of null values was so small, these rows were dropped from the data. The ‘food_supply_stage’ column contains 16 unique values as seen below. These groups were combined into broader groups for the food supply stages: ‘Farm’, ‘Harvest’ and ‘Retail’. This was accomplished by changing the values in the ‘food_supply_stage’ column to one of the three categories based on the agricultural experience of the author. The number of rows under each value was examined, and then samples were taken so that the sample sizes were close in size between groups. ANOVA does not require an equal number of tuples per group, so this step was simply a preference of the author to avoid any unknown issues that might be cause by unequal sample sizes between the groups.

In [7]:
```
data=data.dropna()
```

In [8]:
```
data['food_supply_stage'].unique()
```

Out[8]:
```
array(['Farm', 'Harvest', 'Storage', 'Processing', 'Whole supply chain',
       'Retail', 'Trader', 'Wholesale', 'Post-harvest', 'Food Services',
       'Transport', 'Pre-harvest', 'Households', 'Distribution', 'Market',
       'Stacking'], dtype=object)
```

In [9]:
```
data['food_supply_stage'].value_counts()
```

Out[9]:
```
Farm                  11755
Harvest                3439
Storage                2054
Transport              1703
Whole supply chain      161
Processing               86
Post-harvest             29
Wholesale                28
Retail                   23
Households               10
Trader                    8
Distribution              4
Food Services             4
Market                    2
Pre-harvest               2
Stacking                  1
Name: food_supply_stage, dtype: int64
```

In [10]:
```
data['food_supply_stage'] = data['food_supply_stage'].replace('Storage','Retail')
data['food_supply_stage'] = data['food_supply_stage'].replace('Processing','Retail')
data['food_supply_stage'] = data['food_supply_stage'].replace('Whole supply chain','Retail')
data['food_supply_stage'] = data['food_supply_stage'].replace('Trader','Retail')
data['food_supply_stage'] = data['food_supply_stage'].replace('Wholesale','Retail')
data['food_supply_stage'] = data['food_supply_stage'].replace('Post-harvest','Retail')
data['food_supply_stage'] = data['food_supply_stage'].replace('Households','Retail')
data['food_supply_stage'] = data['food_supply_stage'].replace('Distribution','Retail')
data['food_supply_stage'] = data['food_supply_stage'].replace('Food Services','Retail')
data['food_supply_stage'] = data['food_supply_stage'].replace('Market','Retail')
data['food_supply_stage'] = data['food_supply_stage'].replace('Food_Services','Retail')
data['food_supply_stage'] = data['food_supply_stage'].replace('Transport','Retail')
data['food_supply_stage'] = data['food_supply_stage'].replace('Pre-harvest','Farm')
data['food_supply_stage'] = data['food_supply_stage'].replace('Stacking','Retail')
```

In [11]:
```
data['food_supply_stage'].unique()
```

Out[11]:
```
array(['Farm', 'Harvest', 'Retail'], dtype=object)
```

In [12]:
```
data['food_supply_stage'].value_counts()
```

Out[12]:
```
Farm       11757
Retail      4113
Harvest     3439
Name: food_supply_stage, dtype: int64
```

In [13]:
```
data = data.groupby('food_supply_stage').apply(lambda x: x.sample(3400)).reset_index(drop=True)
```

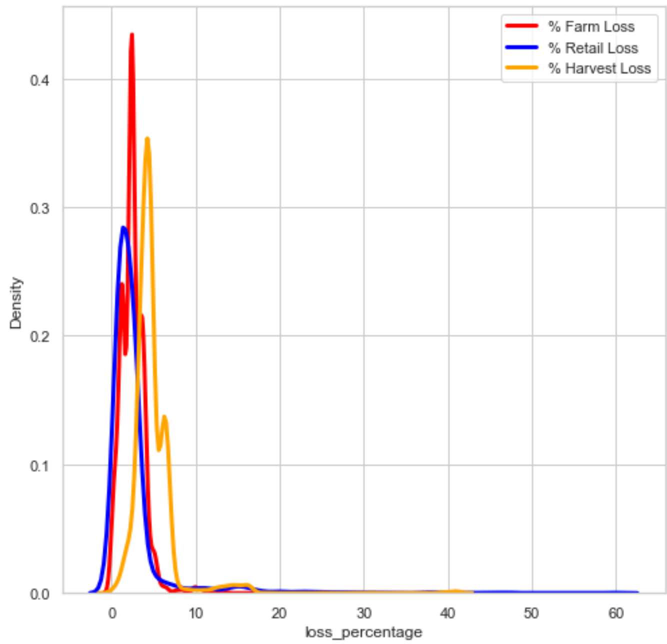
In [15]:
```
data.head()
```

Out[15]:

	food_supply_stage	loss_percentage
0	Farm	0.45
1	Farm	2.17
2	Farm	4.28
3	Farm	5.43
4	Farm	3.72

Normalizing the data is the final step in the cleaning process for this analysis. ANOVA assumes the data has a normal distribution. A density plat and a histogram were used to visualize the distribution of the 'loss_percentage' column. The three categories were layered and labeled on each visualization. The visualizations revealed that the data has a positive skew. This was resolved by removing the outliers of data that created the tail of the skew. By examining the visualizations, the values under 8 in the loss_percentage column contain almost all the data. Very few rows created the tails. All rows with a loss percentage of less than 8% were retained and new visualizations were created to check distribution. The new distribution appeared to be normal, so the data cleaning process was complete. The values of categories in the 'food_supply_stage' were counted and all three groups have nearly 3300 rows each.

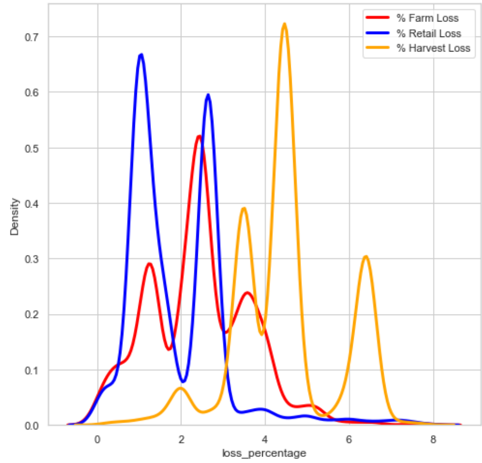
In [16]:
```
# Density plot to see distribution of loss percentage
farm = pd.DataFrame(data[data['food_supply_stage']=='Farm'])
harvest = pd.DataFrame(data[data['food_supply_stage']=='Harvest'])
retail = pd.DataFrame(data[data['food_supply_stage']=='Retail'])
plt.rcParams["figure.figsize"] = (8,8)
sns.distplot(farm['loss_percentage'], hist=False, kde=True, label = '% Farm Loss',
             kde_kws = {'linewidth': 3}, color='red',
              hist_kws={'edgecolor':'black'})
# Density plot to see distribution of loss percentage
plt.rcParams["figure.figsize"] = (8,8)
sns.distplot(retail['loss_percentage'], hist=False, kde=True, label='% Retail Loss',
             kde_kws = {'linewidth': 3}, color='blue',
              hist_kws={'edgecolor':'black'})
# Density plot to see distribution of loss percentage
plt.rcParams["figure.figsize"] = (8,8)
sns.distplot(harvest['loss_percentage'], hist=False, kde=True, label='% Harvest Loss',
             kde_kws = {'linewidth': 3}, color='orange',
              hist_kws={'edgecolor':'black'})
plt.legend()
```



In [17]:
```
# reduce data to create a normal distribution of loss %
df = pd.DataFrame(data[data['loss_percentage'] < 8])
```

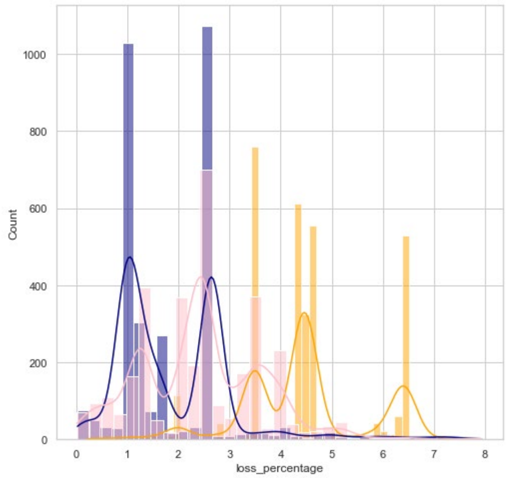
In [18]:
```
# Density plot to see distribution of loss percentage
farm2 = pd.DataFrame(df[df['food_supply_stage']=='Farm'])
harvest2 = pd.DataFrame(df[df['food_supply_stage']=='Harvest'])
retail2 = pd.DataFrame(df[df['food_supply_stage']=='Retail'])
plt.rcParams["figure.figsize"] = (8,8)
sns.distplot(farm2['loss_percentage'], hist=False, kde=True, label = '% Farm Loss',
             kde_kws = {'linewidth': 3}, color='red',
              hist_kws={'edgecolor':'black'})
# Density plot to see distribution of loss percentage
plt.rcParams["figure.figsize"] = (8,8)
sns.distplot(retail2['loss_percentage'], hist=False, kde=True, label='% Retail Loss',
             kde_kws = {'linewidth': 3}, color='blue',
              hist_kws={'edgecolor':'black'})
# Density plot to see distribution of loss percentage
plt.rcParams["figure.figsize"] = (8,8)
sns.distplot(harvest2['loss_percentage'], hist=False, kde=True, label='% Harvest Loss',
             kde_kws = {'linewidth': 3}, color='orange',
              hist_kws={'edgecolor':'black'})

plt.legend()
```



In [19]:
```
# Visualize distribution of 'loss_percentage'
plt.rcParams["figure.figsize"] = (8,8)

sns.histplot(data=harvest2, x='loss_percentage', kde=True, color='orange', label='% Harvest Loss')
sns.histplot(data=retail2, x='loss_percentage', kde=True, color='navy', label='% ')
sns.histplot(data=farm2, x='loss_percentage', kde=True, color='pink')
plt.show()
```



In [20]:
```
#Check how many tuples are in each category
df['food_supply_stage'].value_counts()
```

Out[20]:
```
Farm       3365
Harvest    3255
Retail     3225
Name: food_supply_stage, dtype: int64
```

Analysis

One-way analysis of variance (ANOVA) is a statistical method for testing for differences in the means of three or more groups (JMP, 2022). If only two groups were being tested, ANOVA could have still been used, but there is no point. A T-test works fine with two groups, and ANOVA is specifically designed to model three or more groups. Since one dependent continuous variable is being used to compare three groups, one-way ANOVA is the appropriate method to test whether the mean of the loss percentage is significantly different between the three groups. One disadvantage to using ANOVA is it requires further testing after the model is run to assure the model is a good fit.

If the P-value of the ANOVA table is less than 0.05, the null-hypothesis is rejected, and the question can be answered that there are statistically significant differences in the mean percentage of food loss between the three groups. If the P-value is more than 0.05, there is no need to look at the F statistic in the ANOVA table; however, if the null was rejected the F statistic will help determine whether the model is a good fit. The F critical value table is used to determine if the F critical value is higher than the critical value in the table, which would confirm that the null hypothesis can be rejected. After the ANOVA model is run, the Tukey method was used to inspect specific pairs of groups and the statistical differences in each group. A boxplot was generated to visualize the means and variance of the three groups to further validate the ANOVA model. A Q-Q plot was created to verify normality because ANOVA assumes that the distribution of data is normal. Finally, the Levene test was used to test the variance. If the p-value of the Levene test is higher than 0.05, the variance between the three groups is equal. The ANOVA method assumes that the variance between the groups is equal, so the Levene test verifies the ANOVA produced a good model. Finally, the clean data set was exported.

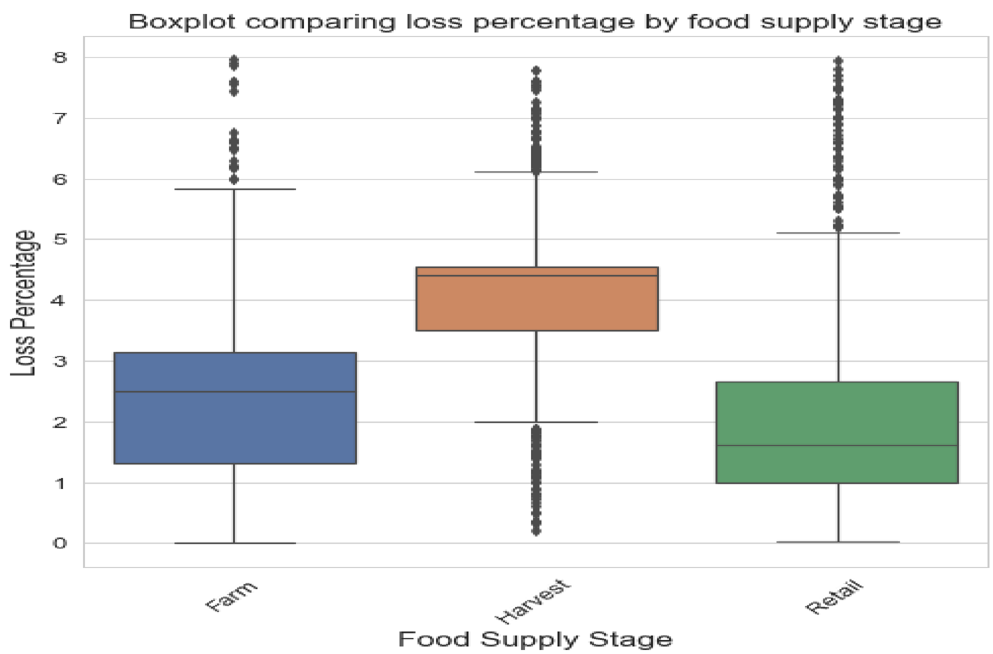
In [21]:
```
# Performing two-way ANOVA
model = smf.ols('loss_percentage ~ food_supply_stage', data=df).fit()
aov_table = anova_lm(model, typ=2)
print(aov_table)
```

```
                       sum_sq      df           F  PR(>F)
food_supply_stage  12102.187175     2.0  4293.92311     0.0
Residual           13869.569056  9842.0         NaN     NaN
```

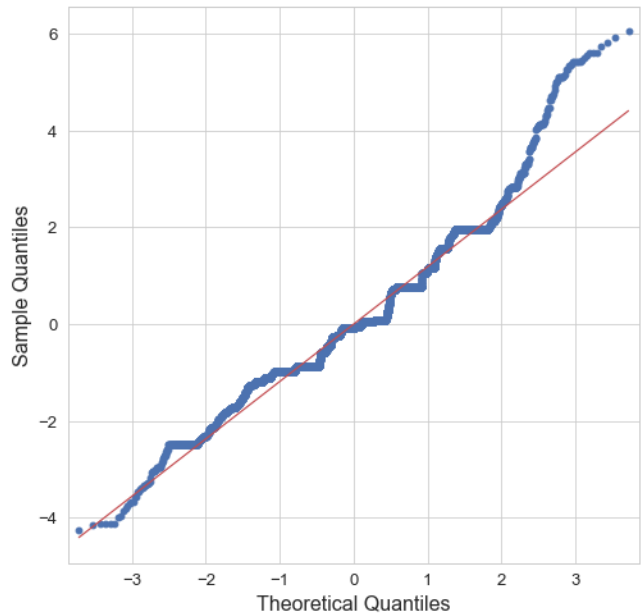
In [22]:
```
# Which stage is significantly different using Tukey method
mcStage = multi.MultiComparison(df['loss_percentage'], df['food_supply_stage'])
results_stage = mcStage.tukeyhsd()
print(results_stage.summary())
```

```
 Multiple Comparison of Means - Tukey HSD, FWER=0.05
=====================================================
 group1  group2 meandiff p-adj  lower   upper  reject
-----------------------------------------------------
   Farm Harvest   2.0351 0.001  1.9667  2.1035   True
   Farm  Retail  -0.5553 0.001 -0.6239 -0.4867   True
Harvest  Retail  -2.5904 0.001 -2.6595 -2.5212   True
-----------------------------------------------------
```

In [23]:
```
# Visualize the loss % of the 3 categories
sns.set_context("paper",font_scale=1.5, rc={"font.size":16, "axes.titlesize":16,"axes.labelsize":16})
plt.title('Boxplot comparing loss percentage by food supply stage')
sns.boxplot('food_supply_stage', y='loss_percentage', data=df)
plt.xlabel('Food Supply Stage')
plt.ylabel('Loss Percentage')
plt.xticks(rotation=45)
plt.show()
```



In [24]:
```
# ANOVA assumes normality. Q-Q plot diplays normality of data
residuals = model.resid
fig = sm.qqplot(residuals, line='s')
plt.show()
```



In [25]:
```
# Check for the equality of variances of the treatments using Levene test
# The p-value refers to the significance of variation, so a p-value > .05 means variance is equal and ANOVA model is ok
farm3 = df['loss_percentage'][df['food_supply_stage']=='Farm']
harvest3 = df['loss_percentage'][df['food_supply_stage']=='Harvest']
retail3 = df['loss_percentage'][df['food_supply_stage']=='Retail']
(test_statistic, p_value) = stats.levene(farm3,harvest3, retail3)
print("The test statistic is: ", round(test_statistic,5))
print("The P-value is: ", round(p_value,5))
```

```
The test statistic is:  2.88057
The P-value is:  0.05615
```

In [26]:
```
#Export prepared data
df.to_csv(r"C:\Users\mlaws\OneDrive - Western Governors University\Documents\WGU\D214\WFW_clean.csv")
```

Data Summary and Implications

The research question is *"Is there a statistically significant difference in the percentage of food loss between the farm, harvest, and retail food supply stages?"*. The null hypothesis states that the means of each group don't have statistically significant differences. The ANOVA table readout displayed a P-value of 0.0. The P-value is less than 0.05, so the null-hypothesis can be rejected. The F critical value table shows that an F statistic over 19 allows the null to remain rejected, and the F statistic readout was 4293.92311. Using a one-way ANOVA model, the null hypothesis was rejected. The Q-Q plot confirmed that the data has a normal distribution. The Levene test had a P-value of 0.06, so the data has equal variance of means. The boxplot further confirms the ANOVA model correctly determined that the null hypothesis should be rejected.

The analysis confirms that there are statistically significant differences in the mean food loss percentage of the three food supply stages. The Levene test revealed the biggest difference, which was between the groups 'Farm' and 'Harvest'. The boxplots show the greatest mean food loss happened during 'Harvest'. This analysis will allow stakeholders to direct research into finding why farming and harvest are so statistically different when measuring the percentage of food loss. The recommendation of this analysis is to collect more data to dig deeper into the answer that differences exist between the groups. If no differences had existed, the stakeholders would have no need to dig further, because no matter what stage of food supply, the loss would be the same. This analysis determined that there is a need to dig further into why more loss happens during harvest, and what the relationship is between farming and harvesting that contributes to the difference in the percentage of food loss.

For further analysis, the food supply stages could be broken down into the subgroups they were formed from in the beginning to explore the loss percentage at a more precise stage in the food supply. An analyst could also run a model on the different commodities in the data set. The location and date of loss is provided, both of which would make great representations. A Tableau dashboard was created for this analysis at: https://public.tableau.com/views/CerealsPulsesLostDuringFarming/CerealsPulsesLossDuringFarmingStage?:language=en-US&:display_count=n&:origin=viz_share_link . This dashboard allows stakeholders to explore additional features of the original data that was extracted. The year, commodity, activity, and location can all be filtered to visualize various aspects of the data.

References

FAO. (n.d.). *Food Loss and Waste Database.* https://www.fao.org/platform-food-loss-waste/flw-data/en/

FAO. (2022, September 29). *Stop Food Loss and waste, for the people, for the planet.* https://www.un.org/en/observances/end-food-waste-day

FAO. (2015). FAO Strategy for Partnerships with Civil Society Organizations. https://sustainabledevelopment.un.org/index.php?page=view&type=400&nr=2213&menu=1515

Bedre, Renesh. (2022, March 6) ANOVA using Python (with examples). https://www.reneshbedre.com/blog/anova.html

Tutorialspoint. (n.d.) *Matplotlib – Pylab module.* https://www.tutorialspoint.com/matplotlib/matplotlib_pylab_module.htm

Conde, Ximena. (2022, October 19). *They planned a three-day giveaway. But Philadelphians claimed 300,000 free avocados in less than 3 hours.* https://www.inquirer.com/news/philadelphia/avocado-free-giveaway-fdr-park-philadelphia-20221019.html

Luc Z. (2020, June 26) *One way ANOVA in Python 3.* https://www.youtube.com/watch?v=s6eZ806dqkI