CCT College Dublin

Assessment Cover Page

To be provided separately as a word doc for students to include with every submission

Module Title:	*Programming for DA* *Statistics for Data Analytics* *Machine Learning for Data Analysis* *Data Preparation & Visualisation*
Assessment Title:	*MSC_DA_CA2*
Lecturer Name:	*Sam Weiss* *John O'Sullivan* *Muhammad Iqbal* *David McQuaid*
Student Full Name:	*Usukhbayar Tsendgombo*
Student Number:	*2022418*
Assessment Due Date:	*6th January 2023*
Date of Submission:	*6th January 2023*

Declaration

By submitting this assessment, I confirm that I have read the CCT policy on Academic Misconduct and understand the implications of submitting work that is not my own or does not appropriately reference material taken from a third party or other source. I declare it to be my own work and that all material from third parties has been appropriately referenced. I further confirm that this work has not previously been submitted for assessment by myself or someone else in CCT College Dublin or any other higher education institution.

Group ID - MSc in Data Analytics

Author: Usukhbayar Tsendgombo
E-mail: 2022418@student.cct.ie
Student ID: 2022418

Abstract

My assignment has been tasked with analyzing Ireland's Agricultural data and comparing the Irish Agri sector with other countries worldwide. My Research is on cows' milk products comparing analysis with Ireland and Spain. In this work, I have focused on comparing the product types and amount of dairy products of these two countries and predicting the future production volume.

The analysis plan is to understand the dataset, prepare dataset, modeling, analysis and prediction.

List of Figures

Figure 1 - Milk and milk product statistics 5
Figure 2 - Animal output 6
Figure 3 - The dataset head 9
Figure 4 - Renamed Dataframe 10
Figure 5 - Datatype 10
Figure 6 - Count of missing data 11
Figure 7 - Statistical information of dairy product 11
Figure 8 - Relationship between Raw cows' milk delivered to dairies and Year 12
Figure 9 - Relationship between Drinking milk and Year 12
Figure 10 - Geographical information of total amount of milk product 13
Figure 11 - Production and utilization of milk on the counties 13
Figure 12 - Drinking milk by Altair chart 14
Figure 13 - Drinking milk by Dash 14
Figure 14 - Mean and Standard deviation 16
Figure 15 – Result of one sample T Test 17
Figure 16 – Plot of T test result 17
Figure 17 – One-Way ANOVA test result 18
Figure 18 – Wilcoxon result 19
Figure 19 – Two-sample KS test plot 19
Figure 20 – Heatmap of all variables 22
Figure 21 – Values and Regression 23

Report plan

The report will be divided into following phases:

- Phase 1: Data understanding
- Phase 2: Data preparation (cleaning, formatting, handling missing value, detectoutliers and any other)
- Phase 3: Anaysis of statistics (summarise data, plots, discrete distribution, Normaldistribution)
- Phase 4: Machine Learning(Choice of modelling techniques, model building,analysis, present final results)

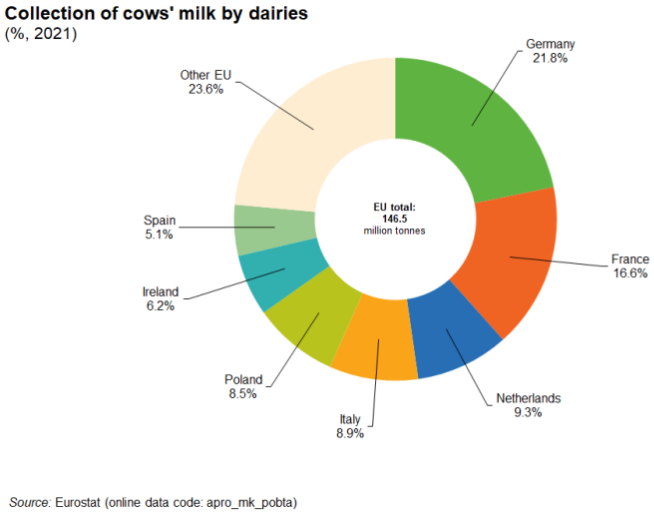
Milk production

The production of raw milk on the EU's farms was an estimated 161.0 million tonnes in 2021, which would represent a year-on-year increase of 0.7 million tonnes. This relative stability in the level of EU production can be put in some context by looking at production levels in the run-up to the abolition of quotas when EU farms produced 149.7 million tonnes of raw milk in 2014 and 145.1 million tonnes in 2013.

In 2021, the vast majority of raw milk was delivered to dairies. Only 10.4 million tonnes was used on farms, either being consumed by the farmer and his family, sold directly to consumers, used as feed or processed directly. Of the 150.7 million tonnes of milk delivered to dairies, 146.5 million tonnes were cows' milk, the rest being ewes' milk, goats' milk or buffaloes' milk.

In 2021, a little more than one-fifth (20.9 %) of the EU's raw cows' milk was produced on farms in Germany and a similar proportion (21.8 %) was processed by German dairies. Indeed, just as Germany, France, the Netherlands, Poland and Italy together provided about two-thirds (64.2 %) of the EU's raw cows' milk in 2021, they also accounted for two-thirds (65.2 %) of cows' milk collected by dairies. (Figure 1).

(Milk and milk product statistics, Eurostat)



[Figure 1] – Milk and milk product statistics

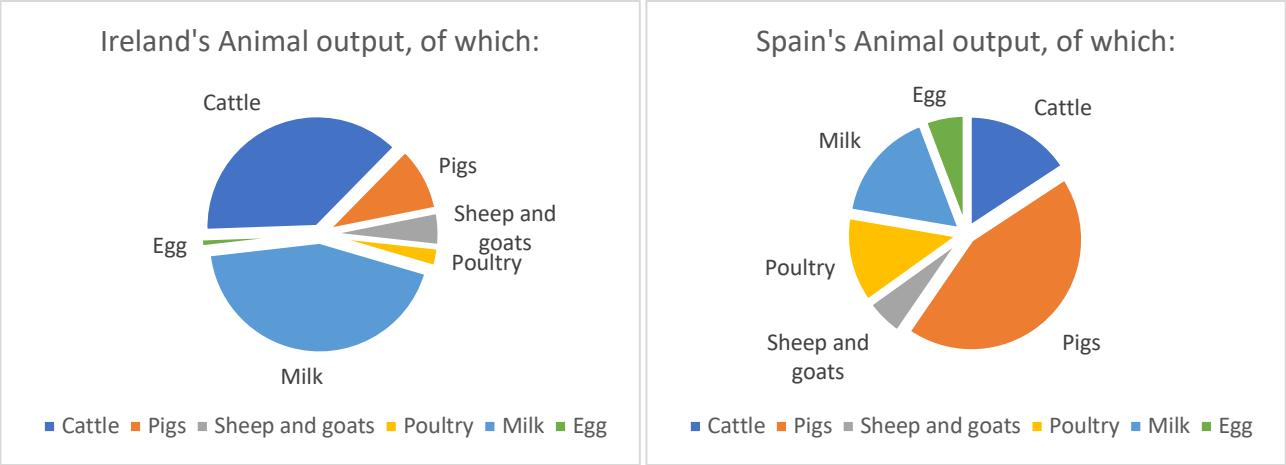
From these statistics, I have decided to compare and deep analyze Ireland's Milk production sector with Spain's Milk production sector.

Agricultural production is a key driver of the Ireland economy. A major livestock producer, Ireland manufactures many derivatives and value-added products from its predominantly cattle base. However, the vast majority (80-90 percent) of its beef and dairy products are exported.

Statistical Factsheet

	Ireland	Spain
Main figures - Year 2020		
Population (1st January)	4 964 440 (persons)	47 332 614 (persons)
Area*	69 947 km^2	505 983 km^2
Currency	EUR	EUR
Nominal GDP at current prices	366 506 million EUR	1 121 698 million EUR
GDP per capita at current prices	70 373 EUR	23 281 EUR
Exports (goods & services)	448 899 million EUR (current prices)	333 583 million EUR (current prices)
Imports (goods & services)	344 283 million EUR (current prices)	312 725 million EUR (current prices)
Balance (goods & services)	104 615 million EUR (current prices)	20 858 million EUR (current prices)
Exports of agricultural products	13 285 million EUR (current prices)	51 426 million EUR (current prices)
Imports of agricultural products	9 536 million EUR (current prices)	30 724 million EUR (current prices)
Economic accounts of agriculture Agricultural output (current basic prices):		
Animal output, of which:	**74.0%**	**39.1%**
Cattle	27.1%	6.0%
Pigs	6.8%	16.7%
Sheep and goats	3.5%	2.1%
Poultry	2.0%	4.8%
Milk	**31.2%**	**6.3%**
Egg	0.9%	2.2%

Ireland and Spain's animal output products percentage of Economic accounts of agriculture



[Figure 2] – Animal output

The figure shows those two countries' animal outputs are totally different. Milk is the highest present (31.2%) of animal agriculture products in Ireland. On the contrary, milk is not a high percentage of animal agriculture products in Spain. Also, animal agriculture products are not the main Economic accounts of agriculture in Spain.

1. Data understanding

1.1. Data Set

Milk and dairy products are the most important field of agriculture industry. These days, value of usage is increasing day by day all around the world and it's mean industry keep a key driver of every countries' economy.

I have chosen two kinds of dataset for this analyze which are a total number Dairy products and Production of cow's milk on farms by years. Those datasets Published by Eurostat and is an official website of the European Union.

The Dairy products dataset include 10 type of Dairy and other animal products (except meat) since 1968 and unit of measure is thousand tones. Types include:

- Raw cows' milk delivered to dairies
- Raw cream delivered to dairies (in milk equivalent)
- Drinking milk
- Cream for direct consumption
- Milk and cream powders, excluding skimmed milk powders,
- Skimmed milk powder
- Concentrated milk
- Acidified milk (yoghurts and other)
- Butter, incl. dehydrated butter and ghee, and other fats and oils derived from milk; dairy spreads
- Cheese from cows' milk (pure)

For the geographical data, I used Production of cow's milk on farms by NUTS 2 regions. This dataset includes total number of production of cow's milk (1000t) by year, GEO(codes) and GEO (Labels). I can see the total number of milk production by year and countries from this dataset, so which is help me to compare Ireland's milk production to other EU countries.

1.2. Describe data

Ireland and Spain's dairy product datasets contains each 55 rows and 6 columns. Type of variables are object. Using method describe () in Python to have a look at the statistics of the data. By this code, I can find total count, mean value, standard deviation, minimum and maximum values of the all columns.
The dataset of Production of cow's milk contains 34 rows and 29 columns. After melt dataset, type of variables: GEO (codes), GEO (Labels) are object, Year is int64, Milk Products obtained (1000 t) is float64.

2. Data prepration and Visualation

Preparators

A preparator is a method that transforms a set of input values into a set of output values that are of higher quality or more useful for the use-case at hand. A preparator's complexity can vary from being fairly simple, such as upper-casing all strings, to being quite complex, such as geocoding address fields. The number of input and output attributes can vary and be of any datatype, although in this work we focus only on alphanumeric values.

- Split attribute: Extract parts of an attribute, moving them into other attribute

- Normalize address: Convert address to its commonly accepted form, fixing inconsistencies
- Geocode: Get the geolocation of an address
- Remove special characters: Remove non-alphanumeric characters: [!@#&$*]
- Transliterate: Remove diacritics from words
- Merge attributes: Merge multiple attributes into a single one
- Acronymize Keep the first character of all tokens
- Capitalize characters Convert all characters to upper case
- Syllabify: Word →syllables preparation
- Phonetic encode: Convert value to its pronunciation representation
- Stem: Reduce word to base form

(Ioannis Koumarelas, Lan Jiang, and Felix Naumann. 2020. Data Preparation for Duplicate Detection.)

2.1 Exploratory data analysis

Exploratory Data Analysis (EDA) is a crucial step in any data science project. However, existing Python libraries fall short in supporting data scientists to complete common EDA tasks for statistical modeling.

Exploring the data in this step is about understanding the usage of tables and producing visualizations to have a suitable approach to the storyline of the case and a better understanding of the data.

2.1.1. Import the required libraries for EDA

Below are the libraries that use to perform EDA (Exploratory data analysis). Imported required libraries: pandas, numpy, seaborn, matplotlib, spicy, statsmodels. And import warnings.filterwarning for the suppress the warnings.

2.1.2. Load the data into the data frame

Load the data into the pandas data frame is certainly one of the most important steps in EDA, as we can see that the value from the data set is comma-separated. So I have to do is to just read the EXCEL into a data frame and pandas data frame does the job for me.

```
df_ie=pd.read_excel("Ireland.Fixed.Dataset.xlsx")
df_sp=pd.read_excel("Spain.Fixed.Dataset.xlsx")
```

Out[3]:

	Dairy and other animal products (except meat)	Raw cows' milk delivered to dairies	Raw cream delivered to dairies (in milk equivalent)	Drinking milk	Cream for direct consumption	Milk and cream powders, excluding skimmed milk powders	Skimmed milk powder	Concentrated milk	Acidified milk (yoghurts and other)	Butter, incl. dehydrated butter and ghee, and other fats and oils derived from milk; dairy spreads	Cheese from cows'milk (pure)
0	Unit of measure	Thousand tonnes	Thousand tonnes	Thousand tonnes	Thousand tonnes	Thousand tonnes	Thousand tonnes	Thousand tonnes	Thousand tonnes	Thousand tonnes	Thousand tonnes
1	1968	:	:	:	:	:	:	:	:	73.25	:
2	1969	:	:	:	3.13	15.29	37.71	:	:	75.39	27.26
3	1970	2790.65	:	409.94	4.6	18.95	34.22	:	:	70.74	28.62
4	1971	2891.01	:	422.92	4.71	13.56	47.49	:	:	73.45	33.07

[Figure.3] The dataset head

2.1.3. Drop rows from DataFrames and Renaming columns

Pandas DataFrame drop() function drops specified labels from rows and columns. The drop() function removes rows and columns either by defining label names and corresponding axis or by directly mentioning the index or column names.

This step removes any features that unnecessary. In this case, the rows Unit of measure is irrelevant in datasets. So, I dropped irrelevant rows by drop() function.

- ***axis****: It has values 0 and 1. We put 0 in the parameter if we want to drop from the index and 1 when we drop from columns. By default, it is 0.*

  df_ire=df_ie.drop([0], axis=0)
  df_spa=df_sp.drop([0], axis=0)

In this dataset, some of the column names are unclear to understand, so I changed name of columns. This is a good approach of improve the readability of the data set.

df_ireland=df_ire.rename(columns = {'Dairy and other animal products (except meat)':'Year', "Butter, incl. dehydrated butter and ghee, and other fats and oils derived from milk; dairy spreads" : "Dairy (butter, ghee, oils and other fats)"})

In [13]:
```
#Renaming columns
df_spain=df_spa.rename(columns ={"Butter, incl. dehydrated butter and ghee, and other fats and oils derived from mi
print(df_spain.columns)
```

```
Index(['Year', 'Raw cows' milk delivered to dairies',
       'Raw cream delivered to dairies (in milk equivalent)', 'Drinking milk',
       'Cream for direct consumption',
       'Milk and cream powders, excluding skimmed milk powders',
       'Skimmed milk powder', 'Concentrated milk',
       'Acidified milk (yoghurts and other)',
       'Dairy (butter, ghee, oils and other fats)',
       'Cheese from cows'milk (pure)'],
      dtype='object')
```

In [80]:
```
#Renaming columns
df_ireland=df_ire.rename(columns = {'Dairy and other animal products (except meat)':'Year', "Butter, incl. dehydrat

print(df_ireland.columns)
```

```
Index(['Year', 'Raw cows' milk delivered to dairies',
       'Raw cream delivered to dairies (in milk equivalent)', 'Drinking milk',
       'Cream for direct consumption',
       'Milk and cream powders, excluding skimmed milk powders',
       'Skimmed milk powder', 'Concentrated milk',
       'Acidified milk (yoghurts and other)',
       'Dairy (butter, ghee, oils and other fats)',
       'Cheese from cows'milk (pure)'],
      dtype='object')
```

[Figure.4] Renamed Dataframe

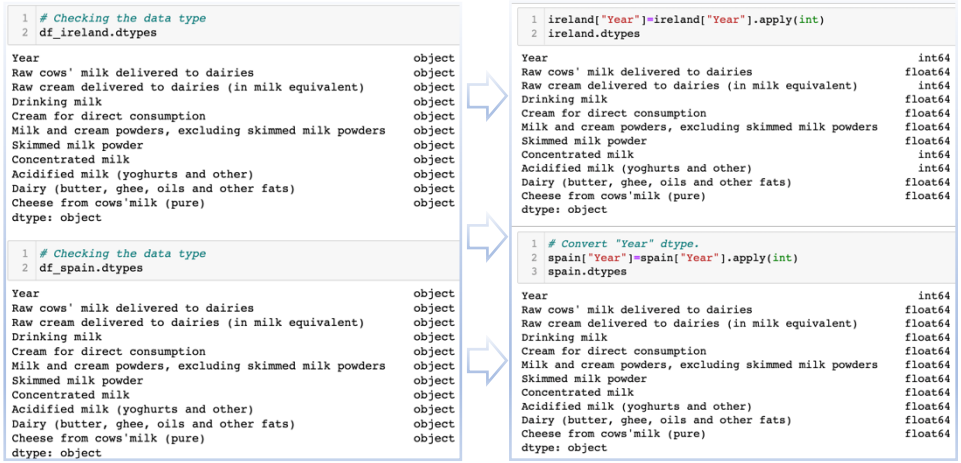
2.1.4. Check the types of dataset

Check for the datatypes because sometimes features (variables) be stored as a string or an object. In this dataset, all columns data type is object, because have a ":" symbol in empty . For the machine learning methods and stats, I have to convert my dataset type to int.
By using replace() function replaced all ":" values to 0.

```
ireland=df_ireland.replace({":": 0})
spain=df_spain.replace({":":0})
```

The column "Year" was a object. So, by the apply() function I changed that columns object format to integer.

```
ireland["Year"]=ireland["Year"].apply(int)
spain["Year"]=spain["Year"].apply(int)
```



[Figure.5] Data type.

2.1.5. Checking duplicates

Duplicate observations can usually cause confusion in a data analysis. I need to check the number of duplicates. In this case I did not have an any duplicated rows and columns.

```
df.duplicated().sum()
(0)
```

2.1.6. Drop the missing or null values

Using Pandas and NumPy for handling missing values in dataset. A common occurrence in a data-set is missing values. This can happen due to multiple reasons like unrecorded observations or data corruption.

In this work, in this dataset have not any missing values.

```
# Get the number of missing data points per column, # Ireland
missing_values_count = ireland.isnull().sum()
missing_values_count[0:10]
```

```
Year                                                0
Raw cows' milk delivered to dairies                 0
Raw cream delivered to dairies (in milk equivalent) 0
Drinking milk                                       0
Cream for direct consumption                        0
Milk and cream powders, excluding skimmed milk powders 0
Skimmed milk powder                                 0
Concentrated milk                                   0
Acidified milk (yoghurts and other)                 0
Dairy (butter, ghee, oils and other fats)           0
dtype: int64
```

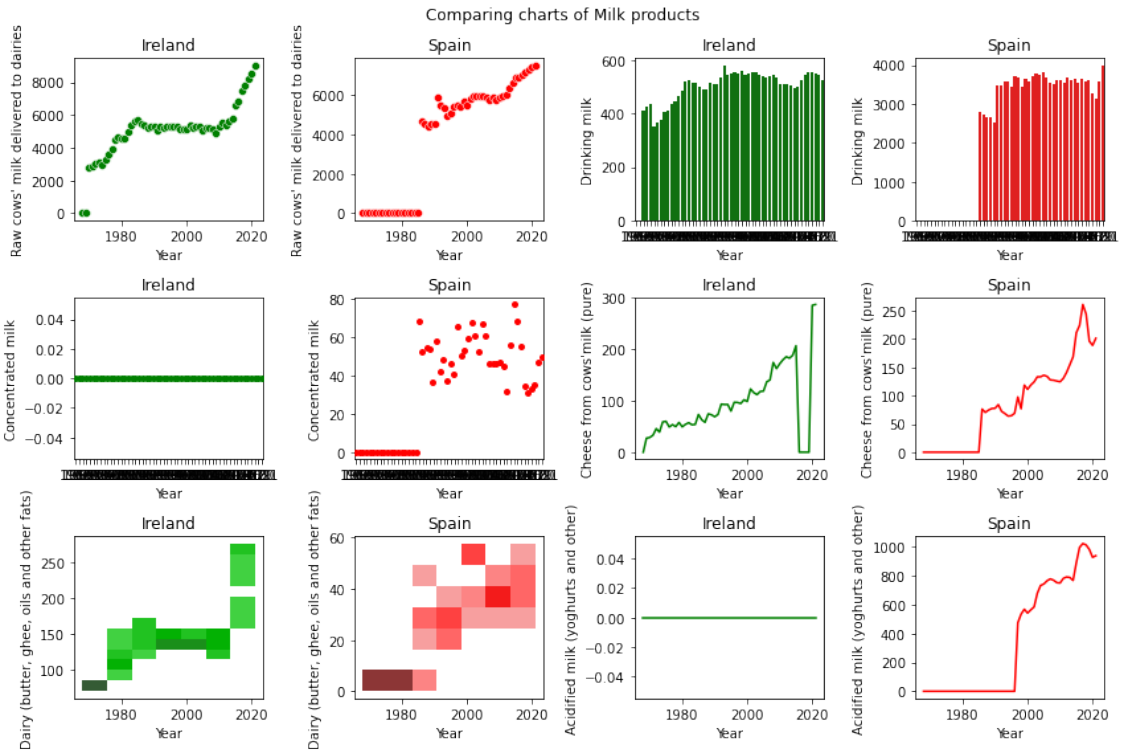
```
# Spain
missing_values_counts = spain.isnull().sum()
missing_values_counts[0:10]
```

```
Year                                                0
Raw cows' milk delivered to dairies                 0
Raw cream delivered to dairies (in milk equivalent) 0
Drinking milk                                       0
Cream for direct consumption                        0
Milk and cream powders, excluding skimmed milk powders 0
Skimmed milk powder                                 0
Concentrated milk                                   0
Acidified milk (yoghurts and other)                 0
Dairy (butter, ghee, oils and other fats)           0
dtype: int64
```

[Figure.6] Count of missing data

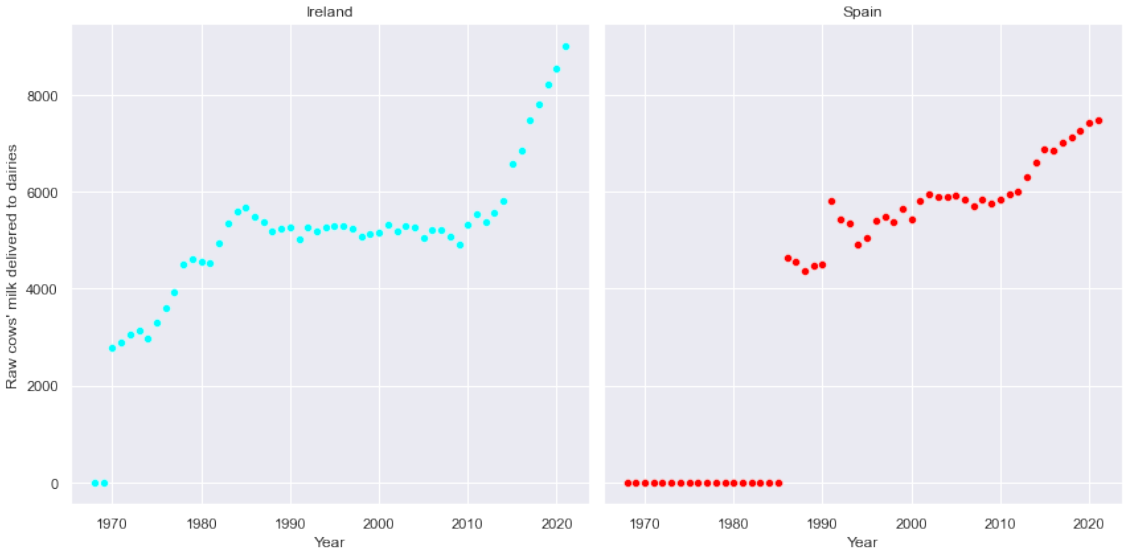
2.2. Summarise my data by statistics and plots

First, I needed to compare these two countries' dairy products by year. Figure.6 shows statistical information of both counties' all 10 types of dairy products. For this comparing figure, used scatterplot, Barplot, Stripplot, Lineplot, and Histplot from the seaborn library. By this step, I can compare and check the variables' relationship with "Year". Also, showing the difference of datasets of Ireland and Spain.

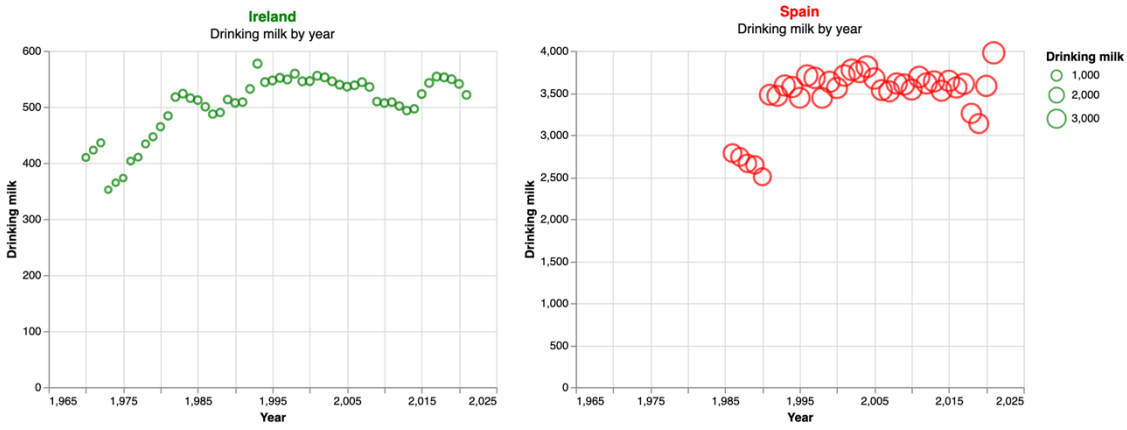


[Figure.7] Statistical information of dairy product

From (fig 7), I found the variables of "Raw cows' milk delivered to dairies" and "Drinking milk" are a high positive correlation with "Year".
For the deeply evaluate this result, I checked the relationship between "Raw cows' milk delivered to dairies", “Drinking milk” and "Year" variables by using a Scatterplot and Altair visualization.



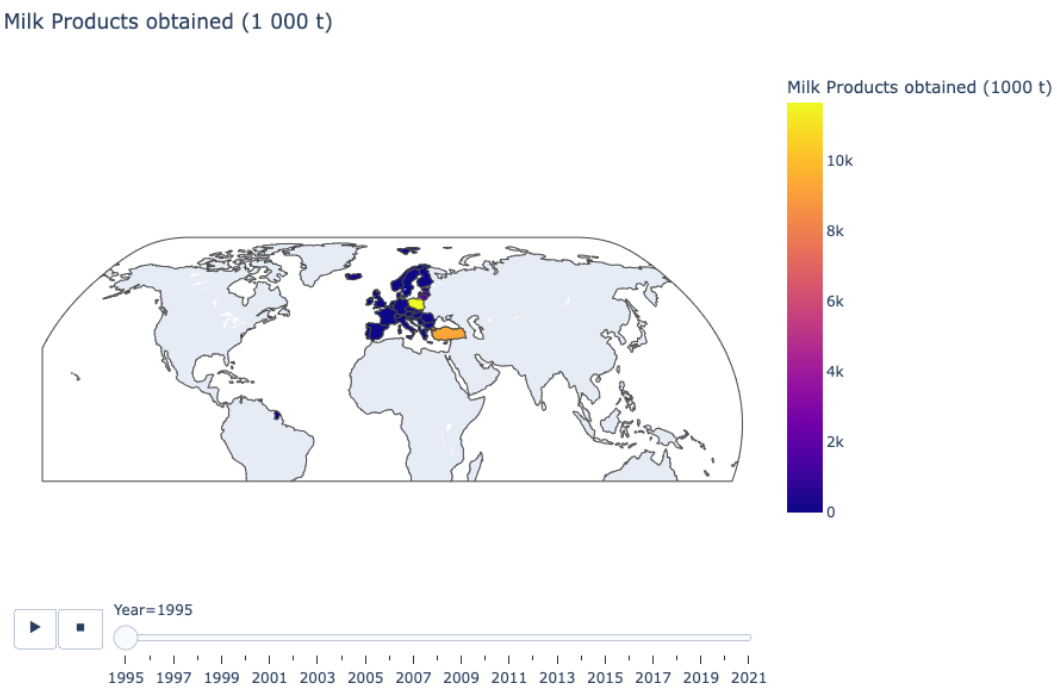
[Figure.8] Relationship between Raw cows' milk delivered to dairies and Year



[Figure.9] Relationship between Drinking milk and Year

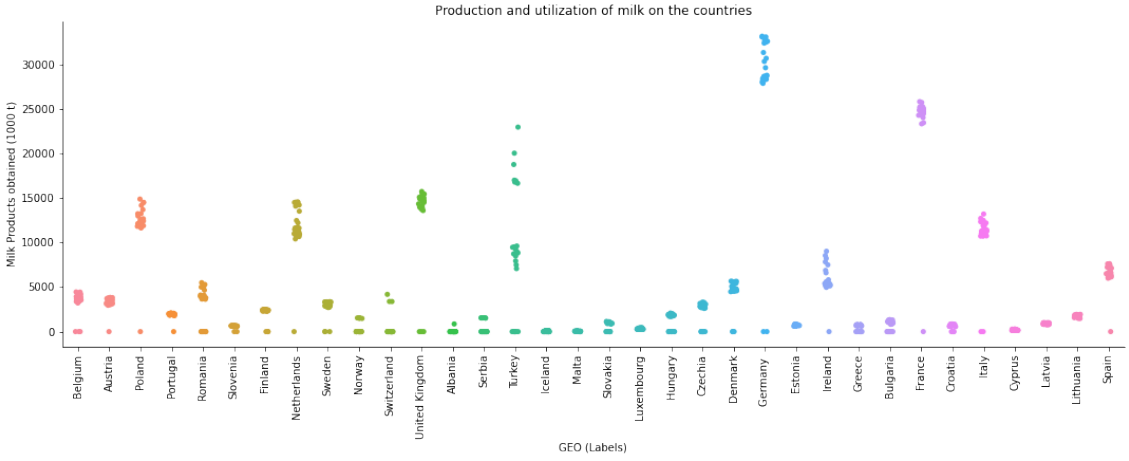
2.2.1. Geographical data

For the geodata, I used data of "Production of cow's milk on farms by NUTS 2 regions [AGR_R_MILKPR".
This figure shows the total milk Products obtained (1000 t) by the countries in 1995-2021. I can see the difference in the total amount of milk products by the countries and time.



[Figure.10] Geographical information of total amount of milk product

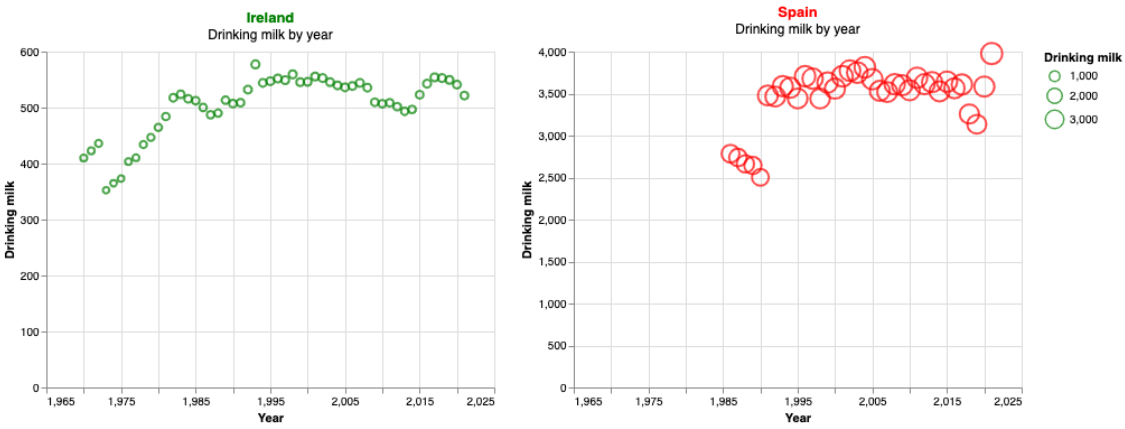
By using the catplot from the seaborn library, I can see Milk Products obtained (1000 t) by year. This figure shows the difference in milk production by year in the countries.



[Figure.11] Production and utilization of milk on the counties

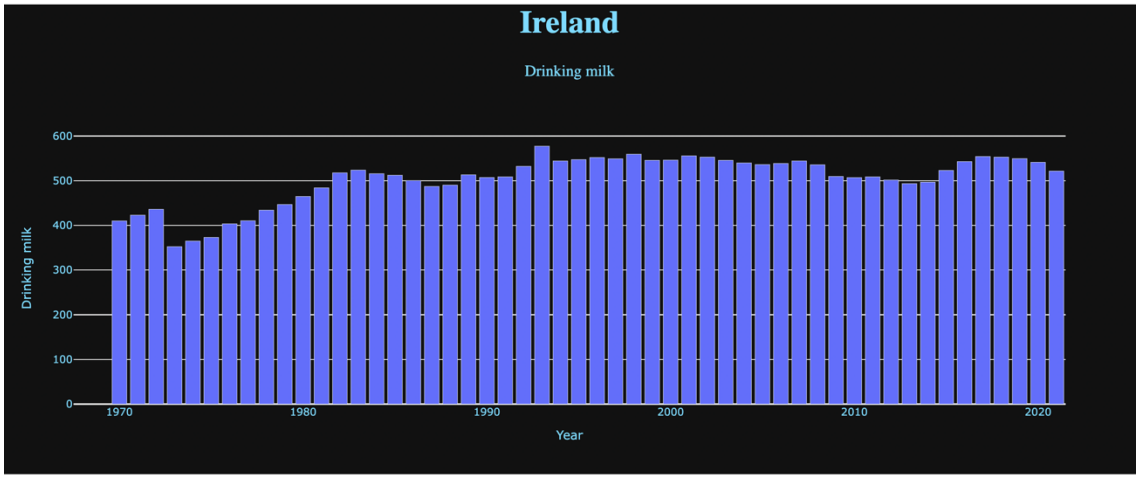
In these figures, I can see Ireland and Spain are more comparable than other countries.

2.2.2. Altair chart



[Figure.12] Drinking milk by Altair chart

2.2.3. Dash



[Figure.13] Drinking milk by Dash

3. Statistics

Statistical Hypothesis Testing

Hypothesis testing is an important activity of empirical research and evidence-based medicine. A well worked up hypothesis is half the answer to the research question. For this, both knowledge of the subject derived from extensive review of the literature and working knowledge of basic statistical concepts are desirable. The present paper discusses the methods of working up a good hypothesis and statistical concepts of hypothesis testing.

(Amitav Banerjee, U. B. Chitnis, S. L. Jadhav, J. S. Bhawalkar, S. Chaudhury, 2009, *Hypothesis testing, type I and type II errors)*

Step 1. Select Test Statistic

Put simply, a test statistic quantifies a data sample. In statistics the term ‘statistic’ refers to any mapping (or function) between a data sample and a numerical value. Popular examples are the mean value or the variance. Formally, the test statistic can be written as

$$tn = T(D(n))$$

whereas $D(n) = \{x1, \ldots, xn\}$ is a data sample with sample size n. Here we denoted the mapping by T and the value we obtain by tn. Typically the test statistic can assume real values, that is, $tn \in R$ but restrictions are possible.

A test statistic assumes a central role in a hypothesis test because by deciding which test statistic to use one determines a hypothesis test to a large extend. The reason is that it will enter the hypotheses we formulate in step 2. For this reason one needs to carefully select a test statistic that is of interest and importance for the conducted study.

We would like to emphasize that in this step

Step 2. Null Hypothesis H_0 and Alternative Hypothesis H_1

At this step, we define two hypotheses which are called the null hypothesis H_0 and the alternative hypothesis H_1. Both hypotheses make statements about the population value of the test statistic and are mutually exclusive. For the test statistic $t = T(D)$ we selected in step 1, we call the population value of t as θ. Based on this we can formulate the following hypotheses:

null hypothesis: $H_0: \theta = \theta_0$

alternative hypothesis: $H_1: \theta > \theta_0$

As one can see, the way the two hypotheses are formulated, the value of the population parameter θ can only be true for one statement but not for both. For instance, either $\theta = \theta_0$ is true but then the alternative hypothesis H_1 is false or $\theta > \theta_0$ is true but then the null hypothesis H_0 is false.

(Frank Emmert-Streib, Matthias Dehmer. 2019, Understanding Statistical Hypothesis Testing: The Logic of Statistical Inference, Review)

What Is T Test?

First, let's understand what the T test is, also known as the student's test. It is an inferential statistical approach to finding the relation between two samples using their means and variances. T test is basically used to accept or reject a null hypothesis H0.. However, to accept or reject the null hypothesis depends on the P value. Mainly if the **P > alpha value** which in most cases is **0.05**, we reject the null hypothesis and consider that there is a significant difference between the two samples.

Types Of T Test In Python
There are four types of T test you can perform in Python. They are as follows:

1. **One sample T test**
2. **Two sample T test (paired)**
3. **Two sample T test (independent)**
4. **Welch T test**

3.1. One Sample Test:

In one sample T test, we usually test the difference between a mean of the sample from a particular group and a mean that we know or we have hypothesized. For example, we hypothesize the mean height of a person in a classroom of 25 students of 5feet. Further, we carry out a T test to know if the mean height is actually 5 feet or not.

$$\frac{x - \mu}{\frac{s}{\sqrt{n}}}$$

Where **x** is the sample mean, μ is hypothesized or known to mean, s is the sample standard deviation and **n** is the sample size.

In this dataset, need to find mean value and standard deviation for T test. Here, I am using Ireland and Spain's amount of the Drinking milk for the test.

```
In [40]:
1  # Find the dataset's StreetID's mean value
2  print(np.mean(ireland['Drinking milk']))
3
4  # Find the StreetId's standard deviation
5  print(np.std(spain['Drinking milk']))
```

```
484.82592592592584
1657.2071633705339
```

[Figure.14] Mean and Standard deviation

T Test Drinking milk

```
# import stats
from scipy import stats
import scipy.stats as stats
import numpy as np
# T-test Ireland's "Drinking milk"
stats.ttest_1samp(ireland['Drinking milk'], popmean=484.82, alternative = "two-sided")
```

```
Ttest_1sampResult(statistic=0.0003956032014123383, pvalue=0.9996858396225692)
```

If the p-value $< \alpha$, we reject H_0. If the p-value $\not< \alpha$, we fail to reject H_0.

p-value for one tailed test is 0.9996858396225692

p-value =0.9996858396225692 > alpha =0.05 We do not reject the null

```
# T-test Spain's "Drinking milk"
stats.ttest_1samp(spain['Drinking milk'], popmean=1657.2, alternative = "two-sided")
```

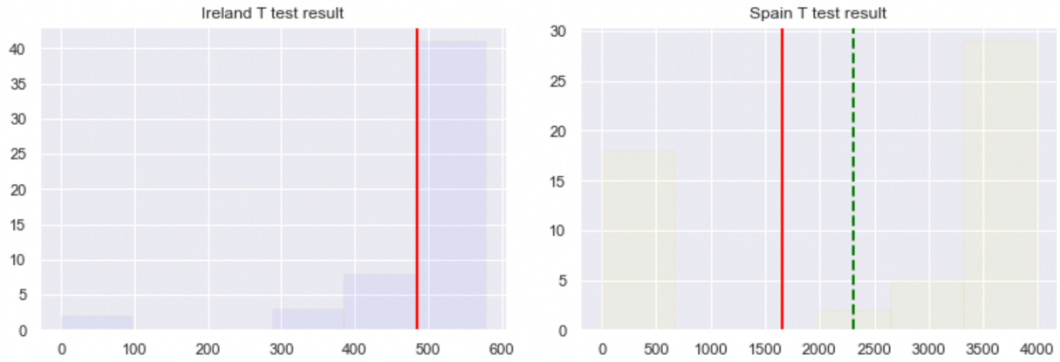
```
Ttest_1sampResult(statistic=2.859853925951753, pvalue=0.006049892935284493)
```

p-value for one tailed test is 0.006049892935284493

p-value =0.006049892935284493 < alpha =0.05 We do reject the null

[Figure.15] Result of one sample T Test

This result shows that in our data (alpha=1), I have enough evidence that the population mean μ is equal to the mean value.



[Figure.16] Plot of T test result

This result shows that in Ireland's Drinking milk value's (alpha=1), I have enough evidence that the population mean μ is equal to the mean value. On the contrary, this result shows that the population mean μ is not equal to the mean value of Spain's Drinking milk value.

3.2. Two Sample Test (paired):

In two sample test, which is paired, we carry out a T test between two means of samples that we take from the same population or group. For example, we apply pesticide on one part of a crop field and further take the mean of yields from the part where there is no pesticide and from the part where the pesticide is applied.

$$\frac{x_1 - x_2}{\sqrt{\frac{v_1^2}{s_1} + \frac{v_2^2}{s_2}}}$$

T test formula for two sample test (paired).

Where $\mathbf{x_1}$ and $\mathbf{x_2}$ are sample means, $\mathbf{v_1}$ and $\mathbf{v_2}$ are variances of two samples, respectively, and $\mathbf{s_1}$ and $\mathbf{s_2}$ are sample sizes.

- Perform the two-sample t-test with equal variances:

Ttest_indResult(statistic=-7.9928055845686865, pvalue=1.7266526728018275e-12)

- Perform the two-sample t-test:

Ttest_indResult(statistic=-7.9928055845686865, pvalue=1.084133367444915e-10)

3.3. ANOVA test:

One-Way ANOVA in Python: One-way ANOVA (also known as "analysis of variance") is a test that is used to find out whether there exists a statistically significant difference between the mean values of more than one group.

Hypothesis involved:
A one-way ANOVA has the below given null and alternative hypotheses:

- H_0 (null hypothesis): $\mu 1 = \mu 2 = \mu 3 = \ldots = \mu k$ (It implies that the means of all the population are equal)
- H_1 (null hypothesis): It states that there will be at least one population mean that differs from the rest

```
from statsmodels.stats.multicomp import pairwise_tukeyhsd
import scipy.stats as stat
stats.f_oneway(ireland['Drinking milk'], spain['Drinking milk'])
```

```
F_onewayResult(statistic=63.88494111271237, pvalue=1.726652672801847e-12)
```

[Figure.17] One-Way ANOVA result

The F statistic and p-value turn out to be equal to 63.88 and 1.726652672801847e-12 respectively. Since the p-value is less than 0.05 hence we would reject the null hypothesis. This implies that we have sufficient proof to say that every year has a different amount of Drinking milk in both countries.

3.4. The Wilcoxon Signed-Rank test:

The Wilcoxon signed-rank test is a non-parametric statistical hypothesis test used either to test the location of a population based on a sample of data, or to compare the locations

of two populations using two matched samples. The one-sample version serves a purpose similar to that of the one-sample t-test. For two matched samples, it is a paired difference test like the paired t-test (also known as the "t-test for matched pairs" or "t-test for dependent samples"). The Wilcoxon test can be a good alternative to the t-test when population means are not of interest; for example, when one wishes to test whether a population's median is nonzero, or whether there is a better than 50% chance that a sample from one population is greater than a sample from another population.

This is a non-parametric version of the dependent two-sample t-test.

```
stats.ttest_rel(ireland['Drinking milk'], spain['Drinking milk'])
```

```
Ttest_relResult(statistic=-8.327176313698242, pvalue=3.3710695979726986e-11)
```

```
stats.wilcoxon(ireland['Drinking milk'], y=spain['Drinking milk'])
# stats.wilcoxon(x=pre, y=post, method='exact')
```

```
WilcoxonResult(statistic=136.0, pvalue=4.7505054797683766e-07)
```

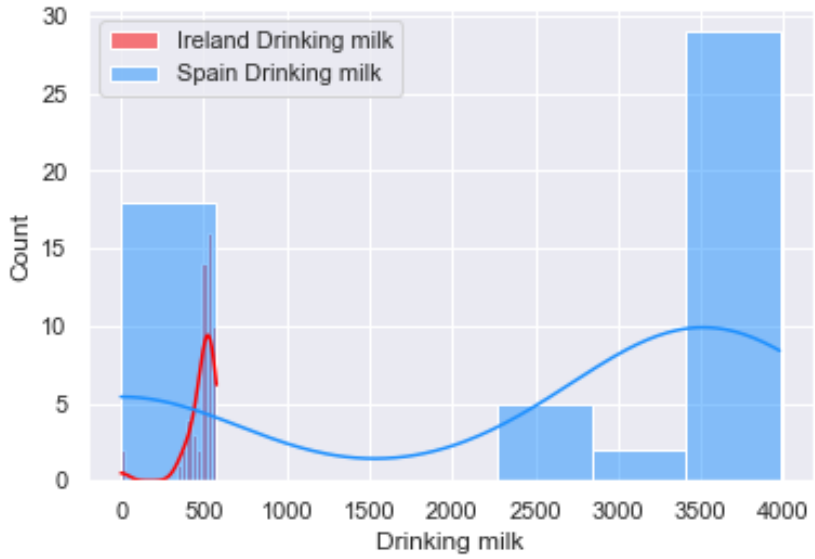
[Figure.18] Wilcoxon result

3.5. Two-sample KS test:

The two-sample KS test is one of the most useful and general nonparametric methods for comparing two samples, as it is sensitive to differences in both location and shape of the empirical cumulative distribution functions of the two samples.

```
# Running the test:
stats.ks_2samp(ireland['Drinking milk'], spain['Drinking milk'])
```

```
KstestResult(statistic=0.6666666666666666, pvalue=1.1204416093347664e-11)
```



[Figure.19] Two-sample KS test plot

4. Machine learning (CRISP-DM, KDD or SEMMA)

Machine learning is the intersection between theoretically sound computer science and practically noisy data. Essentially, it's about machines making sense out of data in much the same way that humans do. (Thoughtful Machine Learning with Python, Matthew Kirk)

The modelling techniques that better adapt to the problem are related to the supervised ML category.

Types of machine learning: Supervised Learning, Unsupervised Learning, Reinforcement Learning.

Types of Supervised Learning: Classification, Regression. I have chosen the Linear Regression method. Because my dataset is numerical and the predicting outputs are numerical as well.

Solutions under Regression: Simple Linear Regression, Multiple Linear Regression, Polynomial Linear regression.

Supervised - Regression – Linear Regression, Lasso Regression

Supervised:

Supervised learning is used whenever we want to predict a certain outcome from a given input, and we have examples of input/output pairs. We build a machine learning model from these input/output pairs, which comprise our training set. Our goal is to make accurate predictions for new, never-before-seen data. Super- vised learning often requires human effort to build the training set, but afterward automates and often speeds up an otherwise laborious or infeasible task.
There are two major types of supervised machine learning problems, called classification and regression.

Regression:

Regression is a statistical method used in finance, investing, and other disciplines that attempts to determine the strength and character of the relationship between one dependent variable (usually denoted by Y) and a series of other variables (known as independent variables).

Also called simple regression or ordinary least squares (OLS), linear regression is the most common form of this technique. Linear regression establishes the linear relationship between two variables based on a line of best fit. Linear regression is thus graphically depicted using a straight line with the slope defining how the change in one variable impacts a change in the other. The y-intercept of a linear regression relationship represents the value of one variable when the value of the other is zero. Non-linear regression models also exist, but are far more complex.

4.1. Linear Regression

Linear Regression or ordinary least squares (OLS) is statistical model used to predict the relationship between independent and dependent variables. Linear models (LMs) are a class of models that are widely used in practice and LMs make a prediction using a linear function of the input features.

Linear models are the simplest parametric methods and always deserve the right attention, because many problems, even intrinsically non-linear ones, can be easily solved with these models. A regression is a prediction where the target is continuous and its applications are several, so it's important to understand how a linear model can fit the data, what its strengths and weaknesses are, and when it's preferable to pick an alternative.

Giuseppe Bonaccorso. 2017, Machine Learning Algorithms : Build Strong Foundation for Entering the World of Machine Learning and Data Science with the Help of This Comprehensive Guide, APA 7th Edition (American Psychological Assoc.), MLA 9th Edition (Modern Language Assoc.)

Regression models are commonly used in statistical analyses. A popular use is to model the predicted risk of a likely outcome. Unfortunately, applying standard regression methods to a set of candidate variables to generate a model tends to lead to overfitting in terms of the number of variables ultimately included in the model, and also overestimation of how well the model performs in terms of using the included variables to explain the observed variability ('optimism bias'). The model tends to perform particularly poorly with predicting observations more 'extreme' (very high or very low) risk. Various (penalized or regularization) regression techniques, can be used to address these problems. LASSO (Least Absolute Shrinkage and Selection Operator) regression, a shrinkage and variable selection method for regression models, is an attractive option as it addresses both problems. Gains in computational power and incorporation into statistical software also mean that its computer-intensive nature is no longer off-putting. One area it has been used is for handing genetic data as the number of potential predictors is often large relative to the number of observations, and there is often little or no *a-priori* knowledge to inform variable selection.

J Ranstam, J A Cook. British Journal of Surgery, Journal article. 2018

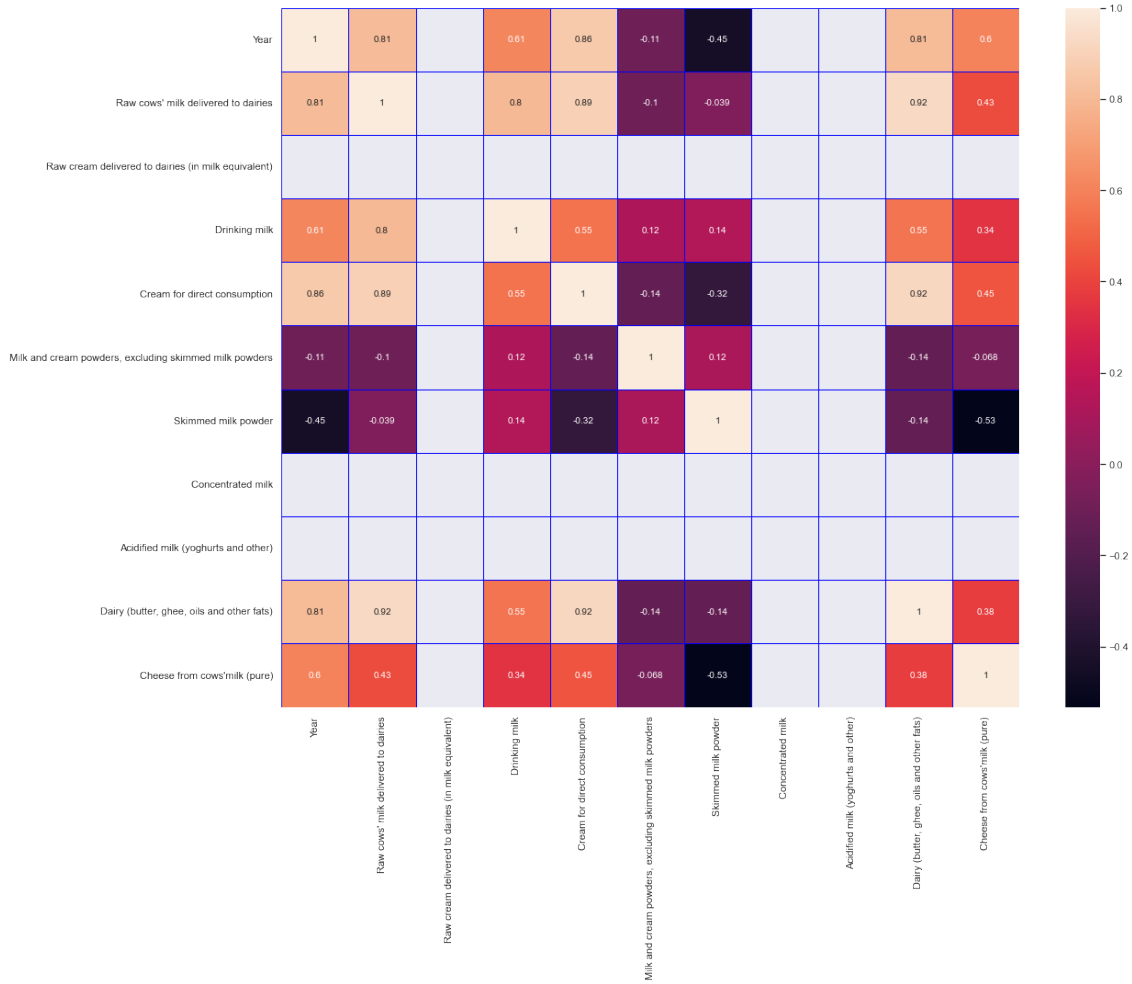
Independent variable:
A variable whose value does not change by effect of other variables and is used to manipulate the dependent variable. It is often denoted as **X**.

Dependent variable
A variable whose value change when there is any manipulation in the values of independent variables. It is often denoted as **Y.**

For run the Machine learning I needed to import the libraries for LinearRegression.

- from sklearn.preprocessing import normalize
- from sklearn.model_selection import train_test_split
- from sklearn.preprocessing import StandardScaler
- from sklearn.metrics import r2_score, mean_squared_error, mean_absolute_error,explained_variance_score
- from sklearn.linear_model import LinearRegression,Ridge,Lasso
- from sklearn.ensemble import RandomForestRegressor
- from sklearn.model_selection import (GridSearchCV, cross_val_score, cross_val_predict, StratifiedKFold, learning_curve)
- from statsmodels.tools.eval_measures import mse, rmse
- from sklearn import preprocessing

After that I need to find Correlation between Features and Independent and Dependent variables (X, Y) from my dataset. From Independent and Dependent variables, separated dataset into Test and Train.



[Figure.20] Heatmap of all variables

From my dataset I used “Year” features into “x” independent variable and store the “Raw cows' milk delivered to dairies” feature into “y” dependent variable.

Created Model with LinearRegression() method and fitting training data with lrm.fit().

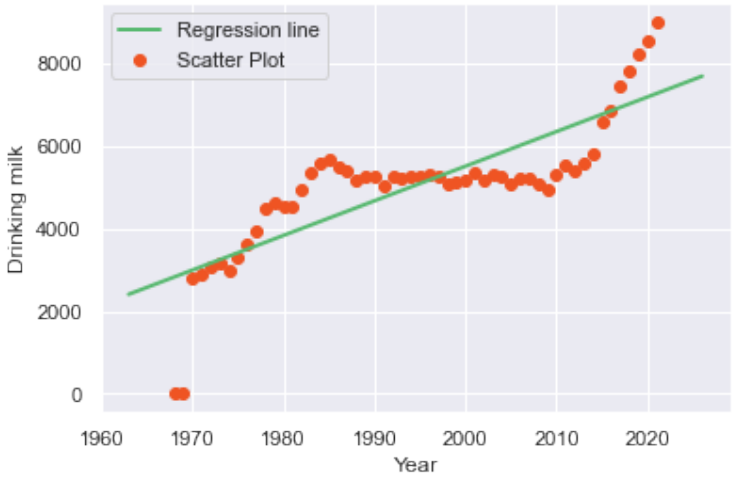
```
lrm = LinearRegression()
lrm.fit(X_train, Y_train)
```

Predict the result
Calculated the predicted value by calling a method predict lrm.predict().

```
y_preds_train = lrm.predict(X_train)
y_preds_test = lrm.predict(X_test)
```

After that, evaluated the model by calculating R-squared of the model in training set, R-squared of the model in test set, Root mean squared error of the prediction and Mean absolute percentage error of the prediction.

R-squared is a goodness-of-fit measure for linear regression models. R-squared measures the strength of the relationship between your model and the dependent variable on a convenient 0 – 100% scale.
The model's calculated R-square score is **0.63.** Is mean my model and the dependent variable are not very good convenient. (63.4%)



[Figure.21] Values and Regression

Finally, I have created a model that uses Linear regression to predict a amount of Ireland's Raw cows' milk delivered to dairies. The Accuracy of the model is 63.4%.

4.2. Lasso Regression:

Lasso can be used as a substitute for Ridge when regularizing linear regression. Using the Lasso restricts coefficients in Ridge Regression to be near to zero in a similar but somewhat different manner known as L1 regularization.

In my data, I Imported lasso from sklearn.linear_model.

```
from sklearn.linear_model import Lasso
```

From Independent and Dependent variables of Spain's dataset, I separated dataset into Test and Train, and renamed Xs and Ys. Because I needed to separate Ireland's X and Y value.

```
Xs=spain['Year'].values
Ys=spain["Raw cows' milk delivered to dairies"].values
```

Fitting model with alpha =1.0 and method of lasso().fit(). After this, I have made a prediction of train and test.

```
ys_preds_train = lasso.predict(Xs_train)
ys_preds_test_lasso = lasso.predict(Xs_test)
```

After that, evaluated the model by calculating R-squared of the model in training set, R-squared of the model in test set, Root mean squared error of the prediction and Mean absolute percentage error of the prediction.

```
R-squared of the model in training set is: 0.818412111982411
-----Test set statistics-----
R-squared of the model in test set is: 0.8861312193505876
Root mean squared error of the prediction is: 915.19369609
Mean absolute percentage error of the prediction is: inf
```

Implementation
By using Spain's dairy product dataset, I have created a model that uses Lasso Regression to predict production of Raw cows' milk delivered to dairies. The Accuracy of the model is 81.9%.

Conclusion

In the project, I used Python tools to complete the analysis and implemented in a Jupyter Notebook. For the research to find similarities I have used 5 different inferential statistics tests. The use of CRISP-DM methodology on Dairy product of two countries made possible to achieve the main object proposed at the beginning. The predictions of the Ireland and Spain's Raw cows' milk delivered to dairies value have been predicted by Liner Regression and Lasso Regression Models. The Liner Regression model accuracy is 63.4%, the Lasso Regression models accuracy is 81.9%.

References

Frank Emmert-Streib, Matthias Dehmer. 2019, Understanding Statistical Hypothesis Testing: The Logic of Statistical Inference, Review

Amitav Banerjee, U. B. Chitnis, S. L. Jadhav, J. S. Bhawalkar, S. Chaudhury, 2009, *Hypothesis testing, type I and type II errors*

Ioannis Koumarelas, Lan Jiang, and Felix Naumann. 2020. Data Preparation for Duplicate Detection.

Jiawei Han, Micheline Kamber, Jian Pei. 2012, *Data mining*, 3rd Edition, Addison-Wesley.

Erik Marsja. 2020, How to use Python to Perform a Paired Sample T-test

Zach. 2020, How to Perform a One-Way ANOVA in Python

Vinícius Trevisan. 2022, Comparing sample distributions with the Kolmogorov-Smirnov (KS) test

Pang-Ning Tan, Michael Steinbach, Vipin Kumar. 2006, Introduction to Data Mining, 1st Edition, Chapter 4

J Ranstam, J A Cook. 2018*,* British Journal of Surgery*,* Journal article.

Giuseppe Bonaccorso. 2017, Machine Learning Algorithms : Build Strong Foundation for Entering the World of Machine Learning and Data Science with the Help of This Comprehensive Guide, MLA 9th Edition (Modern Language Assoc.), APA 7th Edition (American Psychological Assoc.)

Charles Severance. 2009, *Pyhton for Everybody, Exploring data using python 3*

F.M. Dekking, C. Kraaikamp, H.P. Lopuhaa, L.E. Meester. 2005, A Modern Introduction to Probability and Statistics, Springer Texts in Statistics

Harvard Style *UseIt,* Available at http://www.library.uq.edu.au/training/citation/harvard.html

- https://ec.europa.eu/eurostat/web/agriculture/data/database
- https://agriculture.ec.europa.eu/cap-my-country/performance-agricultural-policy/agriculture-country/eu-country-factsheets_en
- https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.ttest_1samp.html