

أكاديمية بيان
BAYAN ACADEMY



حيث البيانات العربية لها كيان

اسم المشروع: معالجة و تحليل المصرفيات للقرض المقدمة من بنك التنمية الاجتماعية

👥: تحديد أعضاء الفريق أو اسم الطالب

• نورة الحربي

• مرام الزهراني

اسم الدورة

مقدمة في علم البيانات

نبذة عن البيانات Data Overview

البيانات المستخدمة تم تنزيلها كصيغة (CSV) من منصة البيانات المفتوحة و هي عبارة عن بيانات خاصة بالقروض التي تم صرفها من بنك التنمية الاجتماعية لعام 2021 (1)

A	B	C	D	E	F	G	H	I	J	K	L	M	N
ID	مدينة	صنف	نوع	نوع العميل	مبلغ التمويل	مبلغ القسط	تاريخ الصرف	جنس العميل	نوع العميل	نوع الإحتياج	نوع الإحتياج	نوع الإحتياج	نوع الإحتياج
1	حائل	قرض مشروع ناشئ	مشروع	غير معرف	30000	<1000	2021/1	ذكر	>= 30	متزوج	سليم	غير معروف	<5000
2	الطائف	قرض سيارة أجرة نقل	نقل	غير معرف	54694	<1000	2021/1	ذكر	>= 30	متزوج	سليم	>= 02	<5000
8	الرياض	قرض التميز	مشروع	قطاع خاص	1585000	>=1000	2021/1	أنثى	>= 40	أعزب	سليم	< 02	>=10000
12	مكة المكرمة	قرض أجرة خاصة نقل	نقل	قطاع حكومي	97230	>=1000	2021/1	ذكر	>= 40	متزوج	سليم	< 02	<5000
13	جازان	قرض أجرة خاصة نقل	نقل	قطاع حكومي	97230	>=1000	2021/1	ذكر	< 30	أعزب	سليم	< 02	<5000
16	مكة المكرمة	قرض أجرة خاصة نقل	نقل	قطاع حكومي	97230	>=1000	2021/1	أنثى	>= 30	مطلق	سليم	>= 02	<5000
20	الرياض	قرض التميز	مشروع	قطاع حكومي	1000000	>=1000	2021/1	ذكر	>= 40	متزوج	سليم	< 02	>=10000
24	جازان	قرض التميز	مشروع	غير معرف	2269000	>=1000	2021/1	ذكر	>= 30	متزوج	سليم	< 02	<5000
33	الأحساء	قرض مشروع حل	مشروع	غير معرف	192000	>=1000	2021/1	ذكر	>= 40	أعزب	سليم	< 02	<5000
45	نجران	قرض أجرة خاصة نقل	نقل	غير معرف	108000	>=1000	2021/1	أنثى	>= 40	متزوج	سليم	< 02	<5000
55	بريده	قرض أجرة خاصة نقل	نقل	غير معرف	86810	>=1000	2021/1	ذكر	>= 30	أعزب	سليم	< 02	<5000
56	الخرج	قرض مشروع ناشئ	مشروع	غير معرف	224000	>=1000	2021/1	ذكر	>= 40	متزوج	سليم	< 02	<5000
60	عرعر	قرض أجرة خاصة نقل	نقل	قطاع حكومي	97230	>=1000	2021/1	ذكر	< 30	متزوج	سليم	< 02	<5000
77	الدمام	قرض مشروع ناشئ	مشروع	غير معرف	298000	>=1000	2021/1	أنثى	>= 30	متزوج	سليم	< 02	<5000
83	مكة المكرمة	قرض أجرة خاصة نقل	نقل	غير معرف	94160	>=1000	2021/1	ذكر	>= 60	متزوج	سليم	< 02	<5000
88	الرياض	قرض مشروع ناشئ	مشروع	غير معرف	230000	>=1000	2021/1	ذكر	>= 30	أعزب	سليم	< 02	<5000
89	الرياض	عربات البيع المتنقلة	مشروع	غير معرف	164000	>=1000	2021/1	ذكر	< 30	أعزب	سليم	< 02	<5000
97	الدمام	برنامج دائم	مشروع	غير معرف	2700000	>=1000	2021/1			غير معرف		غير معروف	
98	مكة المكرمة	قرض مشروع ناشئ	مشروع	غير معرف	198000	>=1000	2021/1	أنثى	< 30	متزوج	سليم	< 02	<5000
105	بيشة	قرض مشروع ناشئ	مشروع	غير معرف	72000	<1000	2021/1	ذكر	< 30	أعزب	سليم	< 02	<5000
108	جازان	قرض مشروع حل	مشروع	غير معرف	300000	>=1000	2021/1	أنثى	< 30	متزوج	سليم	< 02	<5000
109	ينبع	قرض التميز	مشروع	قطاع خاص	2533000	>=1000	2021/1	ذكر	< 30	متزوج	سليم	< 02	>=10000
113	بريده	قرض التميز	مشروع	غير معرف	2409033	>=1000	2021/1	ذكر	>= 30	متزوج	سليم	< 02	<5000
119	النماص	قرض مشروع ناشئ	مشروع	غير معرف	276000	>=1000	2021/1	أنثى	>= 40	متزوج	سليم	< 02	<5000
122	حائل	قرض مشروع ناشئ	مشروع	غير معرف	77000	<1000	2021/1	ذكر	< 30	أعزب	سليم	< 02	<5000

نبذة عن البيانات Data Overview

A	B	C	D	E	F	G	H	I	J	K	L	M	N
ID	مدينة	مبنى	منتج	أع_العمل	مبلغ_التمويل	أ_القسط	أ_الصراف	س_العمل	أ_العمل	أ_الإجتماع	أ_الاجتماع	أ_الاجتماع	أ_الدخل
1	حائل	مشروع	قرض مشروع ناشئ	غير معرف	30000	<1000	2021/1	ذكر	>= 30	متزوج	سليم	غير معروف	<5000
2	الطائف	نقل	قرض سيارة أجرة	غير معرف	54694	<1000	2021/1	ذكر	>= 30	متزوج	سليم	>= 02	<5000
8	الرياض	مشروع	قرض التميز	قطاع خاص	1585000	>=1000	2021/1	أنثى	>= 40	أعزب	سليم	< 02	>=10000
12	مكة المكرمة	نقل	قرض أجرة خاصة	قطاع حكومي	97230	>=1000	2021/1	ذكر	>= 40	متزوج	سليم	< 02	<5000
13	جازان	نقل	قرض أجرة خاصة	قطاع حكومي	97230	>=1000	2021/1	ذكر	< 30	أعزب	سليم	< 02	<5000
16	مكة المكرمة	نقل	قرض أجرة خاصة	قطاع حكومي	97230	>=1000	2021/1	أنثى	>= 30	مطلق	سليم	>= 02	<5000
20	الرياض	مشروع	قرض التميز	قطاع حكومي	1000000	>=1000	2021/1	ذكر	>= 40	متزوج	سليم	< 02	>=10000
24	جازان	مشروع	قرض التميز	غير معرف	2269000	>=1000	2021/1	ذكر	>= 30	متزوج	سليم	< 02	<5000
33	الأحساء	مشروع	قرض مشروع حل	غير معرف	192000	>=1000	2021/1	ذكر	>= 40	أعزب	سليم	< 02	<5000
45	نجران	نقل	قرض أجرة خاصة	غير معرف	108000	>=1000	2021/1	أنثى	>= 40	متزوج	سليم	< 02	<5000
55	بريده	نقل	قرض أجرة خاصة	غير معرف	86810	>=1000	2021/1	ذكر	>= 30	أعزب	سليم	< 02	<5000
56	الخرج	مشروع	قرض مشروع ناشئ	غير معرف	224000	>=1000	2021/1	ذكر	>= 40	متزوج	سليم	< 02	<5000
60	عرعر	نقل	قرض أجرة خاصة	قطاع حكومي	97230	>=1000	2021/1	ذكر	< 30	متزوج	سليم	< 02	<5000
77	الدمام	مشروع	قرض مشروع ناشئ	غير معرف	298000	>=1000	2021/1	أنثى	>= 30	متزوج	سليم	< 02	<5000
83	مكة المكرمة	نقل	قرض أجرة خاصة	غير معرف	94160	>=1000	2021/1	ذكر	>= 60	متزوج	سليم	< 02	<5000
88	الرياض	مشروع	قرض مشروع ناشئ	غير معرف	230000	>=1000	2021/1	ذكر	>= 30	أعزب	سليم	< 02	<5000
89	الرياض	مشروع	عربات البيع المتننة	غير معرف	164000	>=1000	2021/1	ذكر	< 30	أعزب	سليم	< 02	<5000
97	الدمام	مشروع	برنامج دائم	غير معرف	2700000	>=1000	2021/1			غير معرف		غير معروف	

نلاحظ أن جميع المتغيرات المدخلة هي من النوع الاسمي باستثناء متغير مبلغ التمويل (متغير رقمي)، بالإضافة أن تم تعريف المتغيرات المفقودة ك(غير معرف، غير معروف)

المنهجية Method

تم استدعاء المكتبات

```
In [1]: # import librarys
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

import seaborn as sns

import warnings
warnings.simplefilter(action='ignore')

from sklearn.model_selection import train_test_split
%matplotlib inline

from sklearn.linear_model import LinearRegression
```


المنهجية Method

قراءة ملف البيانات و إظهار الصفوف الخمس الاولى باستخدام دالة head().

```
In [2]: dataCsv= pd.read_csv('project_data.csv',encoding = 'utf-8') # read data as csv
```

```
In [3]: dataCsv.head() #previews data
```

```
Out[3]:
```

	ID	المدينة	التصنيف	المنتج	قطاع_العميل	مبلغ_التمويل	قيمة_القسط	تاريخ_الصرف	جنس_العميل	عمر_العميل	أحواله_الإجتماعية	إحتياجات_خاصة	عدد افراد الاسرة	قيمة الدخل
0	1	حائل	مشروع	قرض مشروع ناشيء	غير معرف	30000.0	<1000	2021/1	ذكر	>= 30	متزوج	سليم	غير معروف	<5000
1	2	الطائف	نقل	قرض سيارة أجرة	غير معرف	54694.0	<1000	2021/1	ذكر	>= 30	متزوج	سليم	>= 02	<5000
2	8	الرياض	مشروع	قرض التميز	قطاع خاص	1585000.0	>=1000	2021/1	أنثى	>= 40	أعزب	سليم	< 02	>=10000
3	12	مكة المكرمة	نقل	قرض أجرة خاصة	قطاع حكومي	97230.0	>=1000	2021/1	ذكر	>= 40	متزوج	سليم	< 02	<5000
4	13	جازان	نقل	قرض أجرة خاصة	قطاع حكومي	97230.0	>=1000	2021/1	ذكر	< 30	أعزب	سليم	< 02	<5000

المنهجية Method

إظهار نوع البيانات لكل عمود باستخدام **dtypes**.
استخدام دالة **shape** لمعرفة عدد الأعمدة و الصفوف

```
[In [5]: da_col = data.columns # get columns
```

```
[In [6]: data.dtypes # show types of features (variables)
```

```
Out[6]: ID                int64  
المدينة                object  
التصنيف                object  
المنتج                object  
قطاع_العمل            object  
مبلغ_التمويل           float64  
قيمة_القسط             object  
تاريخ_الصرف             object  
جنس_العمل              object  
عمر_العمل              object  
الحالة_الإجتماعية       object  
إحتياجات_خاصة         object  
عدد افراد الاسرة       object  
قيمة الدخل             object  
dtype: object
```

```
[In [7]: data.shape # show dim. (rows, cols)
```

```
Out[7]: (55129, 14)
```

المنهجية Method

قمنا بحذف عمود ("تاريخ الصرف" ، "المنتج" ، "ID")
باستخدام **drop**.

و قمنا بإعادة تسمية للأعمدة المتبقية بواسطة **rename**.

```
data.drop(['ID', 'المنتج', 'تاريخ الصرف'], axis=1, inplace=True) # remove 2 columns
```


```
data.rename (columns= {'المدينة': 'Region', 'التصنيف': 'class', 'قطاع العميل': 'customer_sector', 'مبلغ التمويل': 'supply_amount',  
                        'قيمة القسط': 'payment_value', 'جنس العميل': 'gender', 'عمر العميل': 'age', 'أحواله الإجتماعية': 'status',  
                        'احتياجات خاصة': 'special_needs', 'عدد افراد الاسرة': 'No_of_members_family', 'قيمة الدخل': 'income_value'}, inplace=True)
```

المنهجية Method

تم استدعاء دالة `describe()` لإظهار الإحصاءات الوصفية كالمتوسط والانحراف المعياري

```
In [11]: data.describe().round(2) # get descriptive stats.  
# remark .describe() work and present descriptive stats with  
### it only has supply amount as float type
```

Out[11]:



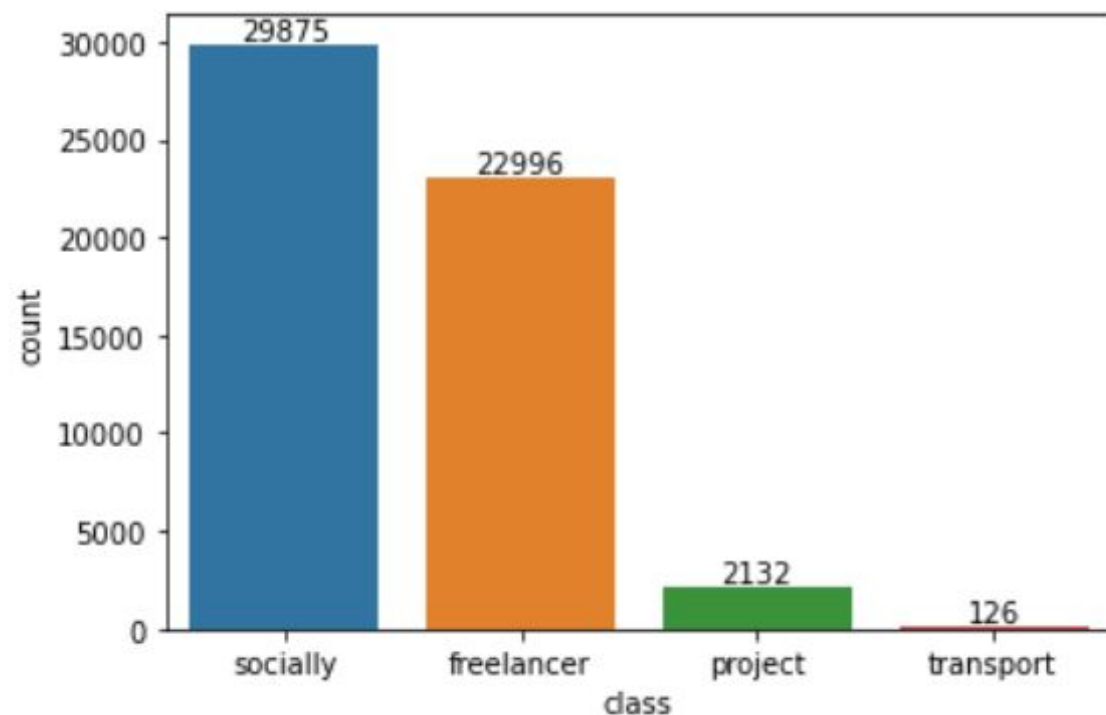
	supply_amount
count	55129.00
mean	62014.57
std	145252.90
min	18000.00
25%	42000.00
50%	60000.00
75%	60000.00
max	10000000.00

نلاحظ أن الدالة تم تطبيقها على متغير واحد ألا وهو (supply_amount) لأنه من النوع الرقمي

المنهجية Method

```
data['class'] = data['class'].map({'نقل': 'transport', 'مشروع': 'project', 'حر': 'freelancer',  
                                'اجتماعي': 'socially' }) # rename
```

```
ax = sns.countplot(x = 'class', data= data,  
                  order = data['class'].value_counts(ascending = False).index); # to display countplot of  
abs_values = data['class'].value_counts(ascending=False).values  
ax.bar_label(container= ax.containers[0], labels = abs_values) # show above each bar label  
plt.show() # show plt  
#plt.savefig() # to save bar plot
```



قمنا باستدعاء دالة **value_count** لإظهار عدد التصنيفات بداخل متغير (Class)

و من ثم عملنا على إعادة تسمية للتصنيفات كما هو موضح باستخدام **map**.

و تم عرضها باستخدام الاعمدة البيانية و بالمثل تم التعامل مع بقية الاعمدة

اشرنا سابقاً أن المتغيرات المفقودة عُرفت (غير معرف، غير معروف) لذلك يتم استبدالها بواسطة دالة **np.nan**

المنهجية Method

في المتغير (Region) قمنا بجمع المدن التي تنتمي الى منطقة واحدة و من ثم قمنا بعرضها باستخدام الاعمدة البيانية محور x يمثل مناطق المملكة و محور y يمثل قيمة التمويل

Replace Region

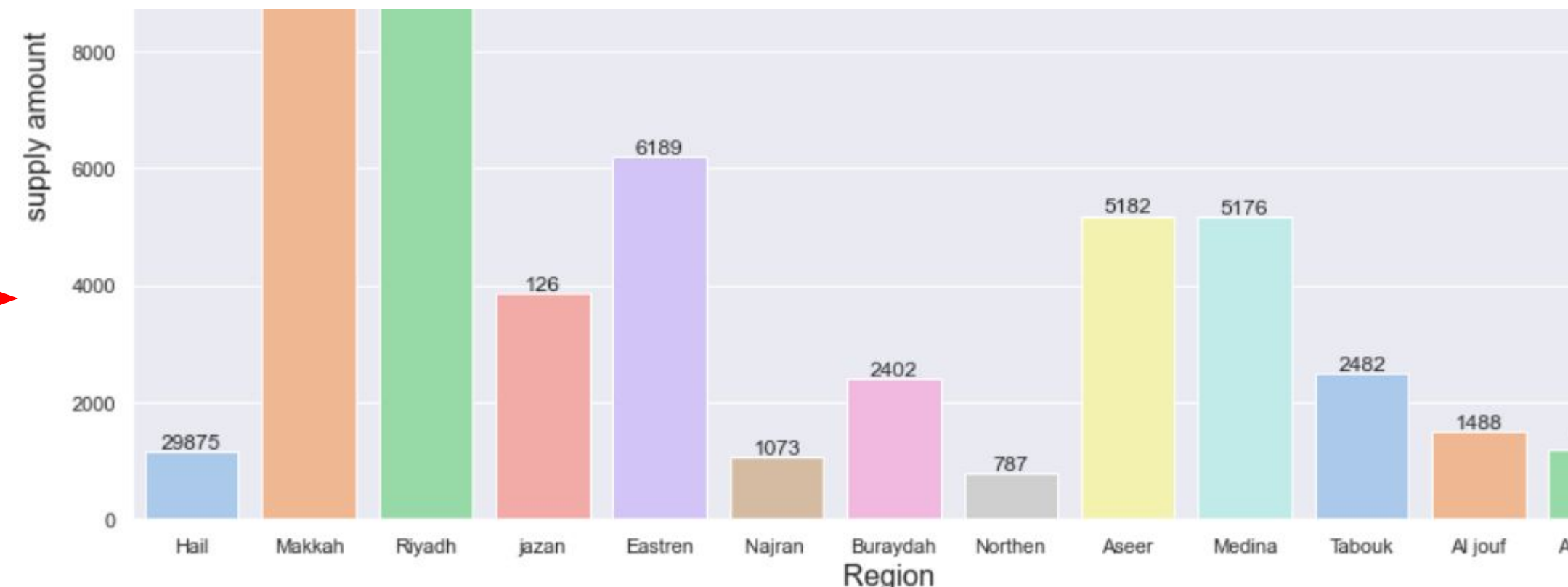
```
data['Region'] = data['Region'].replace(['الطائف', 'مكة المكرمة', 'جدة', 'القصبة'], 'Makkah'); # replace
```

Replace & combine (string) to Riyadh

```
data['Region'] = data['Region'].replace(['الخرج', 'الدوادمي', 'الرياض', 'المجمعة', 'وادي الدواسر'], 'Riyadh');
```

```
data['Region'] = data['Region'].replace(['الينبع', 'المدينة المنورة'], 'Medina'); # replace & combine (string)
```

```
data['Region'] = data['Region'].replace(['الرس', 'بريدة'], 'Buraydah'); # replace & combine (string)
```



المنهجية Method

تم استدعاء الصفوف الخمس الأولى بعد عمل التغييرات السابقة
و من ثم استخدمنا دالة **isnull()** لفحص ما إذا كان هناك قيم
مفقودة

```
[41]: data.head() # repreview data after modify
```

```
[41]:
```

	Region	class	customer_sector	supply_amount	payment_value	gender	age	status	special_needs	No_of_members_family	income_value
0	Hail	project	NaN	30000.0	<1000	male	>= 30	married	no	NaN	<5000
1	Makkah	transport	NaN	54694.0	<1000	male	>= 30	married	no	>= 02	<5000
2	Riyadh	project	pravite sector	1585000.0	>=1000	female	>= 40	single	no	< 02	>=10000
3	Makkah	transport	public sector	97230.0	>=1000	male	>= 40	married	no	< 02	<5000
4	jazan	transport	public sector	97230.0	>=1000	male	< 30	single	no	< 02	<5000

```
[42]: data.isnull().any() # check missing value
```

```
# there are a missing values in customer true , status, No.of members family and income vlaue
```

```
[42]: Region          False
      class          False
      customer_sector  True
      supply_amount   False
      payment_value   False
      gender          False
      age             True
      status          True
      special_needs   False
      No_of_members_family  True
      income_value    True
      dtype: bool
```

نلاحظ وجود قيم مفقودة في بعض المتغيرات (customer_sector , age, status ,..)

المنهجية Method

اضفنا دالة **sum()** الى الدالة **isnull()** لإظهار مجموع القيم المفقودة لكل متغير

```
[43]: data.isnull().sum() # check sum of null value

# there 25713 null of obs. in customer sector
# there 50 null of obs. in status
# there 23119 null of obs. in No of members family
# there 38 null of obs. in income value
# there 147 null of obs. in age
```

```
t[43]: Region      0
      class      0
      customer_sector  25713
      supply_amount  0
      payment_value  0
      gender      0
      age        147
      status      50
      special_needs  0
      No_of_members_family  23119
      income_value  38
      dtype: int64
```

```
[44]: data.dropna(inplace=True) # delete missing value
```

```
45]: data.isnull().any() # re check missing value
```

```
45]: Region      False
      class      False
      customer_sector  False
      supply_amount  False
      payment_value  False
      gender      False
      age        False
      status      False
      special_needs  False
      No_of_members_family  False
      income_value  False
      dtype: bool
```

اعدنا استدعاء دالة **isnull().any()** للتأكد من عدم وجود اي قيمة مفقودة

المنهجية Method

تم تحويل المتغيرات الاسمية كالجنس و الدخل إلى متغير رقمية باستخدام **map**.

```
# convert gender to numrical which 0 = male , 1 = female
data.gender = data.gender.map({'male':1, 'female':0})
```

```
# convert income to numrical
data['income_value'] = data['income_value'].map({
    '<5000':0, '>=5000': 1, '>=7500':2, '>=10000':3
})
```

```
# conver age to numrical
data['age'] = data['age'].map({'< 30':0, '>= 30':1, '>= 40':2, '>= 60':3})
```

```
#convert paymet value to numriac
data.payment_value = data['payment_value'].map({'>=1000':1, '<1000':0})
```

```
# convert no. mebers family to numrical
data.No_of_members_family = data['No_of_members_family'].map({'>= 02':1,
: data.head() # view head
:
```

```
# convert customer to numrical
data.customer_sector = data['customer_sector'].map({'pravite sector':2, '
:
```

```
# convert class to numriac
data['class'] = data['class'].map({'transport':0, 'project':1, 'freelance
'socially':2 })
```

	Region	class	customer_sector	supply_amount	payment_value	gender	age	status	special_needs	No_of_members_family	income_value
2	Riyadh	1	2	1585000.0	1	0	2	1	0	0	3
3	Makkah	0	1	97230.0	1	1	2	0	0	0	0
4	jazan	0	1	97230.0	1	1	0	1	0	0	0
5	Makkah	0	1	97230.0	1	0	1	2	0	1	0
6	Riyadh	1	1	1000000.0	1	1	2	0	0	0	3

قمنا بتحويلها الى بيانات رقمية لإظهار العلاقات بين المتغيرات و إجراء الانحدار

المنهجية Method

قمنا باستدعاء دالة `.describe()`

```
data.describe().round(2) # get descriptive stats. for feautres which has (int, float)
```

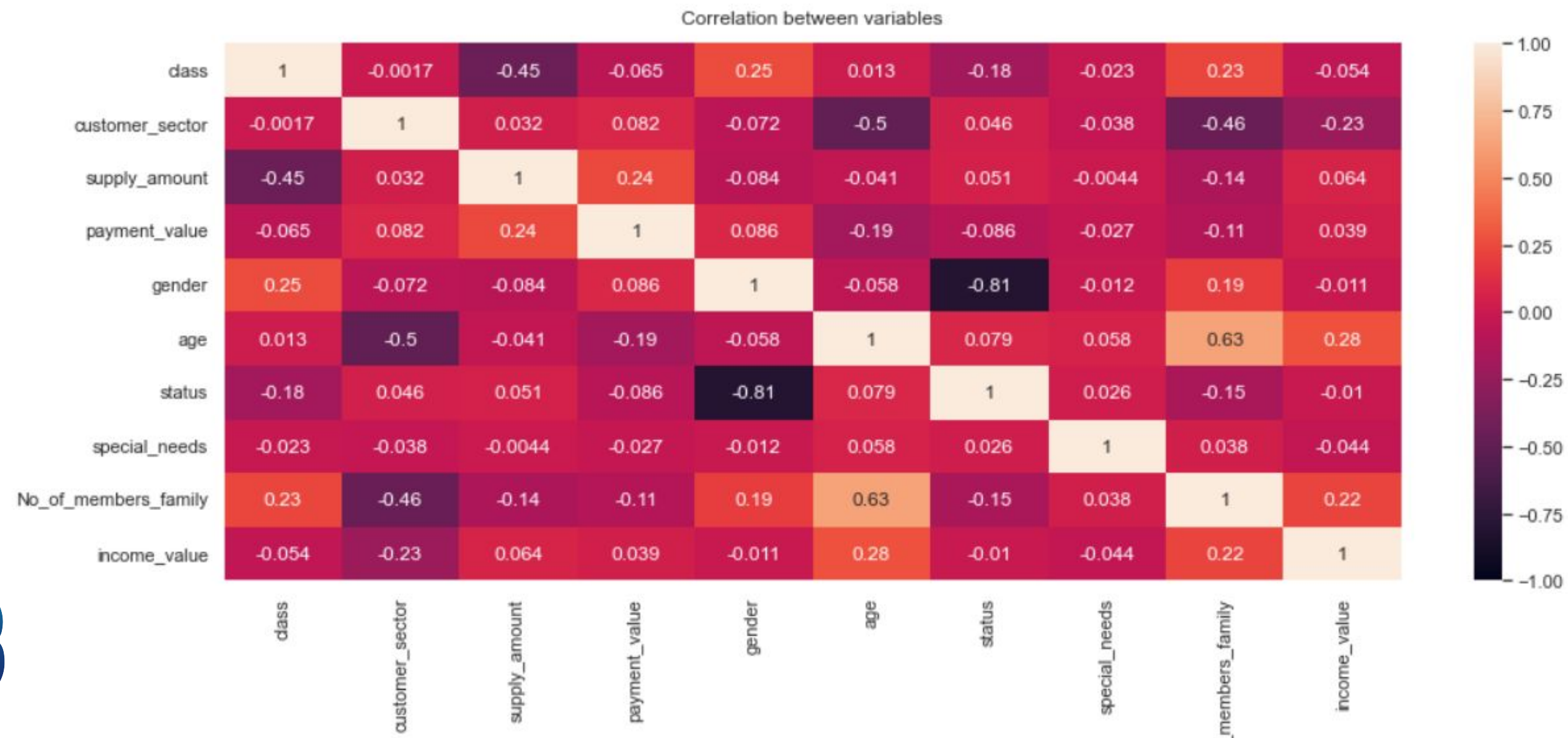
	class	customer_sector	supply_amount	payment_value	gender	age	status	special_needs	No_of_members_family	income_value
count	23145.00	23145.00	23145.00	23145.00	23145.00	23145.00	23145.00	23145.00	23145.00	23145.00
mean	2.97	1.10	56001.72	0.72	0.98	0.89	0.06	0.03	1.37	0.91
std	0.24	0.63	52457.78	0.45	0.16	0.84	0.36	0.18	0.61	1.00
min	0.00	0.00	18000.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
25%	3.00	1.00	54000.00	0.00	1.00	0.00	0.00	0.00	1.00	0.00
50%	3.00	1.00	60000.00	1.00	1.00	1.00	0.00	0.00	1.00	1.00
75%	3.00	2.00	60000.00	1.00	1.00	2.00	0.00	0.00	2.00	2.00
max	3.00	3.00	3925050.00	1.00	1.00	3.00	4.00	1.00	3.00	3.00

نلاحظ أن جميع المتغيرات تم ايجاد لها الإحصاءات الوصفية بعد تغيرها الى متغيرات رقمية

المنهجية Method

استخدمنا دالة **corr()** لإيجاد و عرض العلاقات بين المتغيرات

```
text(0.5, 1.0, 'Correlation between variables')
```



المنهجية Method

لاستكشاف القيم المتطرفة قمنا بحساب الربع الاول باستدعاء

quntile(0.25)

و الربع الثالث **quntile(0.75)**

و من ثم حساب المدى الربيعي الاول و الثاني

اوجدنا مجموع القيم المتطرفة اللي اقل من

قيمة الربع الاول و اللي اكبر من قيمة الربع الثالث

```
[63]: #detecting outliers

# q1 get the first quantile
q1=data['supply_amount'].quantile(0.25)

#q3 get the theird quantile
q3=data['supply_amount'].quantile(0.75)

# If a data point is 1.5xIQR Less the first quartile (Q1) then it is an outlier.
out_lower = q1 - 1.5*(q3 - q1)

# or 1.5xIQR above the third quartile (Q3) then it is an outlier.
out_upper = q3 + 1.5*(q3 - q1)

print("25th quantile of supply amount : {}".format(q1) + ' and ' + 'lower outlier limit : 
print("75th quantile of supply amount : {}".format(q3) + ' and ' + 'lower outlier limit : 

25th quantile of supply amount : 54000.0 and lower outlier limit : 45000.0
75th quantile of supply amount : 60000.0 and lower outlier limit : 69000.0
```

```
[64]: out_lowValue = data['supply_amount'][data['supply_amount'] < out_lower].count()
out_upValue = data['supply_amount'][data['supply_amount'] > out_upper].count()
print("count of lower outliers:", out_lowValue)
print("count of upper outliers :", out_upValue)
print("total outliers:", out_upValue + out_lowValue)
```

```
count of lower outliers: 4804
count of upper outliers : 275
total outliers: 5079
```



المنهجية Method

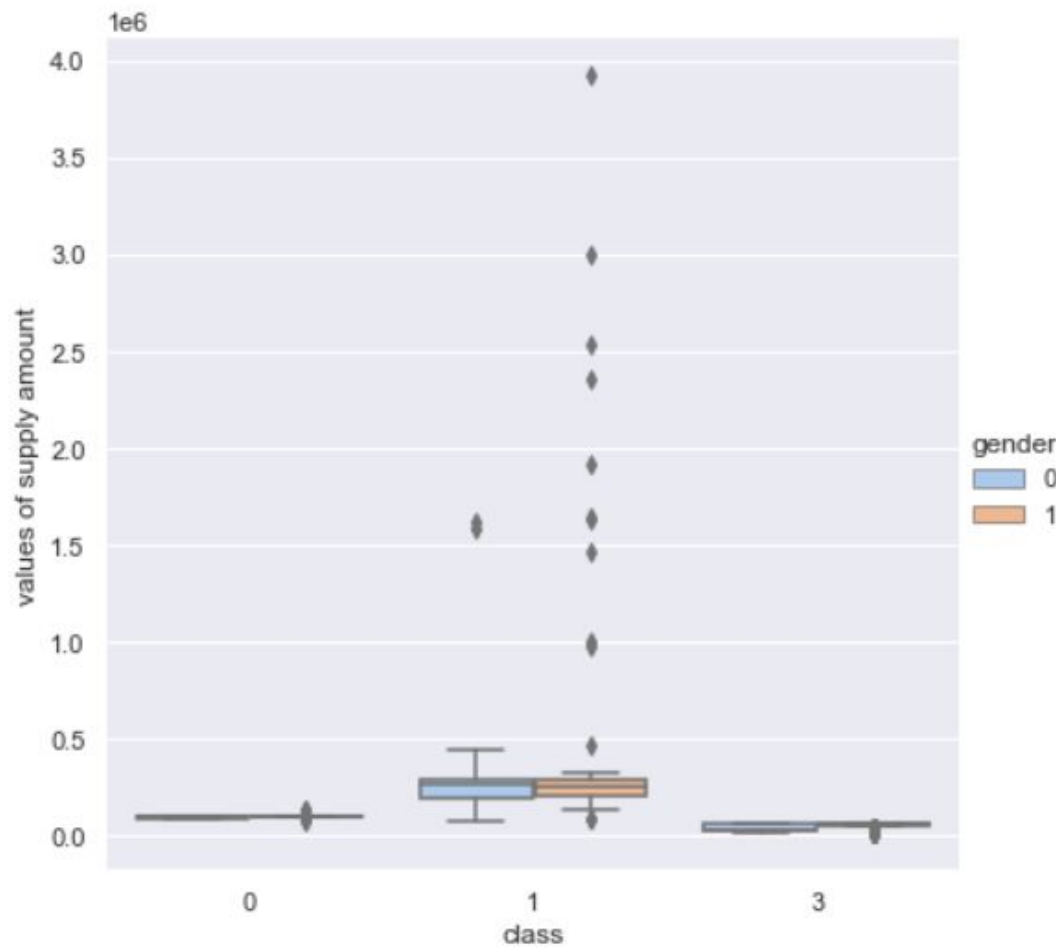
تم تجميع البيانات بحسب المنطقة وإيجاد المتوسط لجميع المتغيرات

```
data.groupby('Region').mean() # show mean for each features based on Region
```

	class	customer_sector	supply_amount	payment_value	gender	age	status	special_needs	No_of_members_family	income_value
Region										
Al Baha	2.984169	1.042216	53759.894459	0.691293	0.989446	0.981530	0.026385	0.036939	1.432718	1.044855
Al jouf	2.977966	0.974576	55069.881356	0.725424	0.977966	0.945763	0.050847	0.035593	1.603390	0.889831
Aseer	2.974859	0.858580	55289.415462	0.685732	0.977373	1.065368	0.064111	0.052797	1.497172	1.010057
Buraydah	2.968460	1.038961	55062.333024	0.719852	0.970315	0.983302	0.078850	0.017625	1.388683	1.097403
Eastren	2.987716	1.389441	55839.249608	0.758495	0.987454	0.704391	0.033455	0.032933	1.263983	0.751438
Hail	2.978972	0.941589	53219.953271	0.677570	0.976636	1.025701	0.072430	0.056075	1.436916	0.915888
Makkah	2.968318	1.115399	56727.448733	0.713518	0.969662	0.960445	0.074309	0.026114	1.370392	0.919739
Medina	2.969668	1.134597	56815.950711	0.716114	0.981043	0.910900	0.041232	0.015640	1.399052	0.915166
Najran	2.961988	1.008772	53349.970760	0.687135	0.982456	1.096491	0.043860	0.081871	1.564327	0.929825
Northen	2.918239	1.034591	59463.710692	0.754717	0.977987	0.830189	0.062893	0.034591	1.396226	0.864780
Riyadh	2.981387	1.059449	56285.140741	0.740551	0.968471	0.808547	0.071985	0.032289	1.318329	0.951377
Tabouk	2.971765	0.944706	54237.647059	0.671765	0.977647	0.910588	0.071765	0.040000	1.441176	0.821176
jazan	2.947414	0.953448	55720.025862	0.661207	0.972414	1.046552	0.072414	0.043103	1.474138	0.848276


```
g = sns.factorplot(x="class", y="supply_amount", hue="gender", data=dat
                  size=6, kind="box", palette="pastel")
g.despine(left=True)
g = g.set_ylabels("values of supply amount")
```

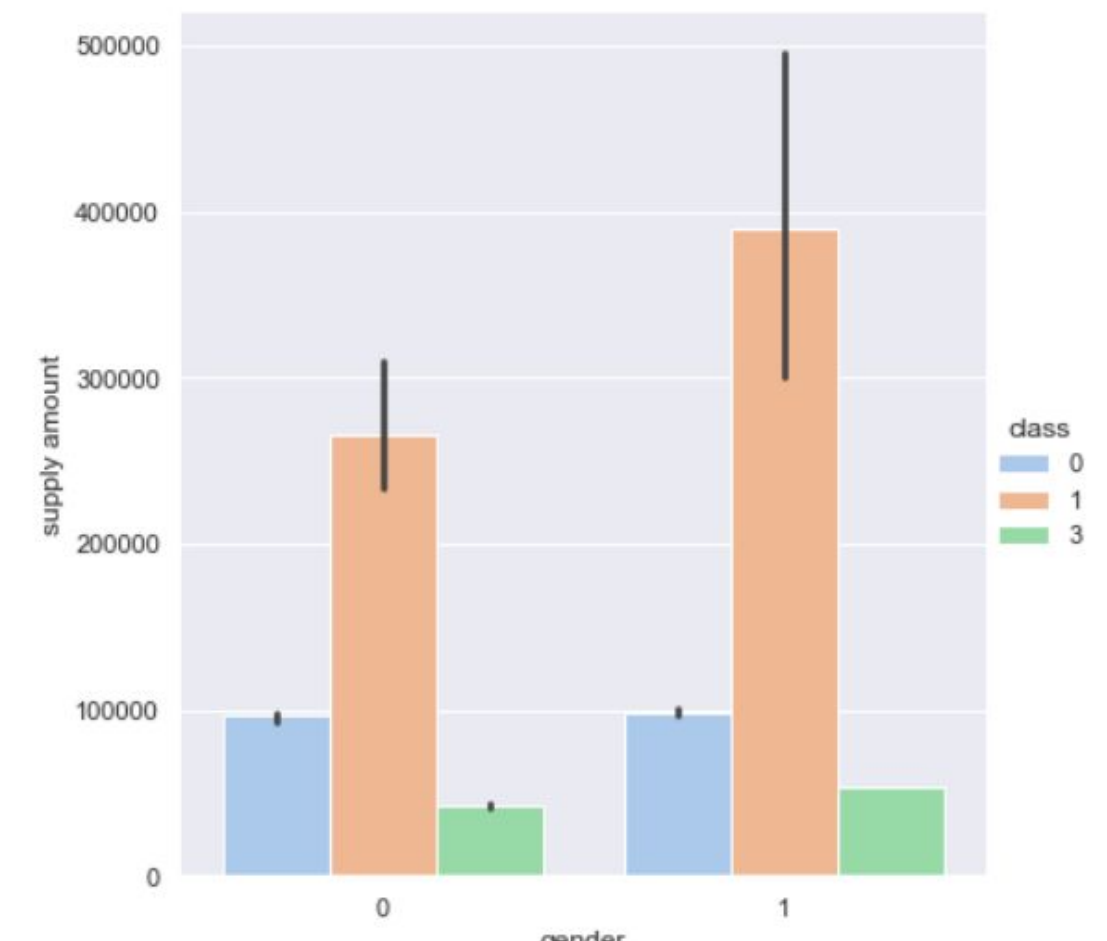
<Figure size 864x576 with 0 Axes>



المنهجية Method

عرض Box plot للبيانات مقسمة بحسب (gender) و يمثل محور الاكس (class) و محور واي (supply amount) نلاحظ وجود قيم متطرفة عند (project) الذي تم استبداله بقيمة 1

```
|: # explore bar plot of x axis = gender vs y axis = supply am
g = sns.factorplot(x="gender", y="supply_amount", hue="class",
                  size=6, kind="bar", palette="pastel")
g.despine(left=True)
g = g.set_ylabels("supply amount")
```



ايضا تم عرض الاعمدة البيانية مقسمة حسب التصنيف (class) حيث أن محور اكس يمثل (gender) و محور واي يمثل (supply amount) و تظهر في الرسمة القيم المتطرفة لكلا الجنسين (male = 1, 0 = female) و اعلى تصنيف يحتوي على القيم المتطرفة هو (1 = project)

المنهجية Method

استخرجنا من عمود (supply amount) الصفوف التي تساوي (gender == 0) و تم تسميته Fdata

و ايضا اخذنا جميع الصفوف التي تساوي (gender == 1) و تسميته Mdata

من ثم عملنا على إنشاء مجموعة بيانات عرفناها (supply_gender) مكونة من عمودين (male supply amount) ، (female supply amount)

```
|: Fdata = data.loc[data['gender']== 0, 'supply_amount'] # get a col(supply amount) ba
```

```
|: Fdata
```

```
|: 2      1585000.0
   5        97230.0
  101       73500.0
  117      265000.0
  150      296000.0
   ...
 54189      30000.0
 54197      30000.0
 54235      60000.0
 54368      42000.0
 54428      30000.0
Name: supply_amount, Length: 571, dtype: float64
```

```
|: Mdata = data.loc[data['gender']== 1, 'supply_amount'] # get a col(supply amount) ba
```

```
|: Mdata.head()
```

```
|: 3        97230.0
   4        97230.0
   6     1000000.0
  12        97230.0
  21     2533000.0
Name: supply_amount, dtype: float64
```

```
|: supply_gender = ({'male supply amount': Mdata, 'female supply amount': Fdata}) # get
supply_gender = pd.DataFrame(supply_gender)
print(supply_gender)
```

```
male supply amount  female supply amount
```

المنهجية Method

تم فحص البيانات (supply_gender) لايجاد البيانات المفقودة

```
[73]: supply_gender.isnull().sum()
```

```
[73]: male supply amount      571  
female supply amount    22574  
dtype: int64
```

```
[74]: #visualite highlight of max, min and null values
```

```
supply_gender.style.highlight_max(color='pink').highlight_min(color='lawngreen').highlight_null()
```

```
[74]:
```

	male supply amount	female supply amount
2	nan	1585000.000000
3	97230.000000	nan
4	97230.000000	nan
5	nan	97230.000000
6	1000000.000000	nan
12	97230.000000	nan
21	2533000.000000	nan
32	1461563.000000	nan
60	134000.000000	nan
72	60000.000000	nan
76	60000.000000	nan

تم استخدام الدالة `style.highlight_null()` لعرض جدول البيانات و لتمييز الخلايا التي تحتوي على قيم مفقودة

المنهجية Method

تم استبدال القيم المفقودة بالوسيط (**median**) نظرا لوجود بيانات متطرفة

```
[165]: #replace NAN value with median value of col.  
supply_gender['male supply amount'].fillna(supply_gender['male supply amount'].median(), inplace=True)  
# replace NAN value with median value of col.  
supply_gender['female supply amount'].fillna(supply_gender['female supply amount'].median(), inplace=True)  
print(supply_gender)
```

	male supply amount	female supply amount
2	60000.0	1585000.0
3	97230.0	54000.0
4	97230.0	54000.0
5	60000.0	97230.0
6	1000000.0	54000.0
...
55113	42000.0	54000.0
55116	42000.0	54000.0
55117	60000.0	54000.0
55120	60000.0	54000.0
55122	60000.0	54000.0

[22145 rows x 2 columns]

المنهجية Method

قمنا باستدعاء مكتبة `scipy.stats` من ثم اجرينا اختبار `t-indepence two test` من دالة `(stats.ttest_ind)`

```
import scipy.stats as stats # import library
# to peforem t.test ind. of two samples
```

```
tres = stats.ttest_ind(a = supply_gender['female supply amount'],b = supply_gender['male supply amount'] )
print(tres)
```

```
Ttest_indResult(statistic=-2.000614390763595, pvalue=0.04543979116231453)
0.04543979116231453
```

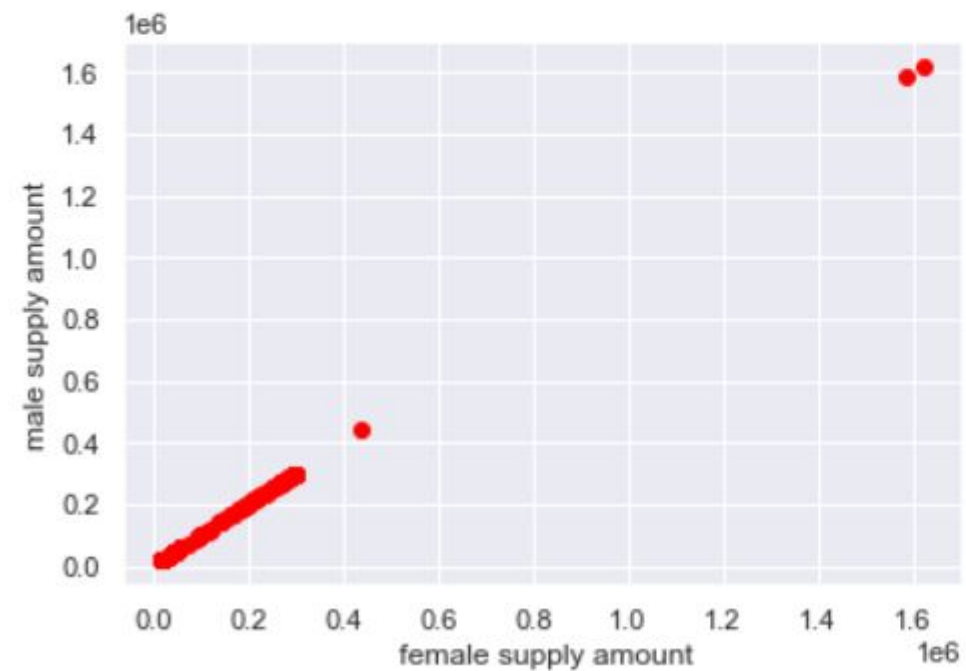
```
print(' P-value = {} is less than \u03B1 = 0.05 , then there difference between means of two samples (female supply , male suppl
```

```
P-value = 0.045 is less than  $\alpha = 0.05$  , then there difference between means of two samples (female supply , male supply)
```

المنهجية Method

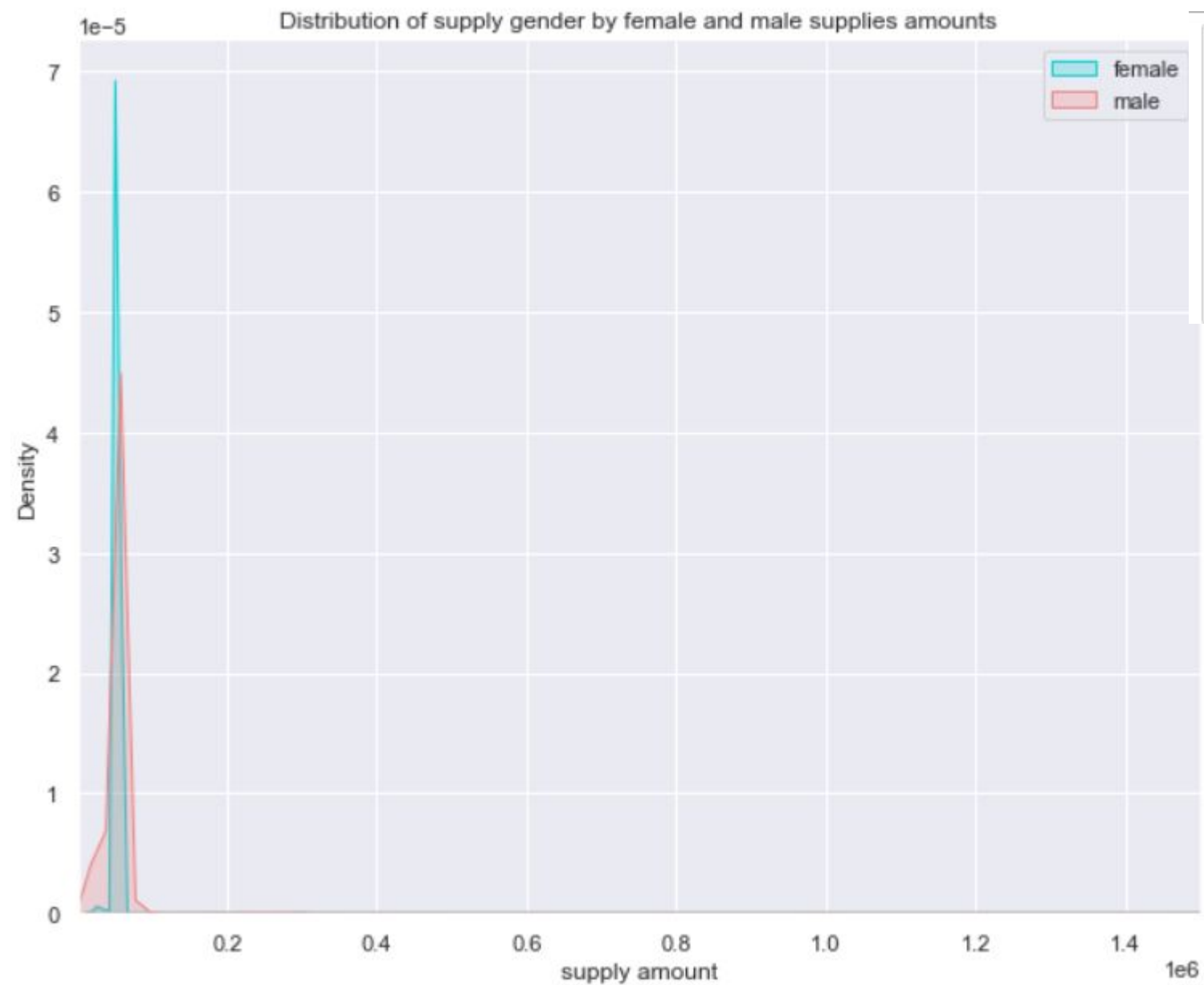
قمنا برسم (scatter plot) لداتا (supply_gender) حيث محور اكس يمثل (female supply amount) و محور واي يمثل (male supply amount)

```
In [82]: plt.scatter(supply_gender['female supply amount'],supply_gender['female supply amount'], c = "red")  
  
#set label of axes  
plt.xlabel(" female supply amount")  
plt.ylabel("male supply amount")  
plt.show() # to show plot
```



المنهجية Method

عرض توزيع بيانات (supply gender) بالنسبة ل (male supply) و (female supply)



```
plt.figure(figsize=(10,8))
ax = sns.kdeplot(supply_gender["female supply amount"], color="darkturquoise", shade=True, label = 'female')
sns.kdeplot(supply_gender["male supply amount"], color="lightcoral", shade=True, label = 'male')
plt.legend()
plt.title('Distribution of supply gender by female and male supplies amounts')
ax.set(xlabel='supply amount')
plt.xlim(1000,1500000)
plt.show()
```

المنهجية Method

انشأنا متغير x حيث يمثل جميع الاعمدة من `data` ماعدا ('Region',supply amount)
ايضا انشأنا y و يمثل فقط متغير ('supply amount') من `data`

Linear Regrission

```
x = data.drop(['supply_amount','Region'], axis = 1) # set x variable
```

```
y = data['supply_amount'] # set y as dependet variable
```

```
print("X shape = ",x.shape,"\n y shape =", y.shape) # to show dimension of vairables
```

```
X shape = (23145, 9)  
y shape = (23145,)
```

```
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.3,random_state=42) # split datase
```

```
print("x_train shape = {}".format(x_train.shape)+ 'and ' + 'x_test = {}'.format( x_test.shape))  
print("y_train = {}".format(y_train.shape)+ 'and ' + 'y_test = {}'.format(y_test.shape))
```

```
x_train shape = (16201, 9) and x_test = (6944, 9)  
y_train = (16201,) and y_test = (6944,)
```

```
lr = LinearRegression()
```

```
lr.fit(x_train, y_train)
```

```
LinearRegression()
```

قسمنا البيانات الى مجموعتي التدريب و
الاختبار و من ثم تم بناء نموذج الانحدار
الخطي

المنهجية Method

من الدالة `_coef` تظهر لنا معاملات الانحدار
و من ثم تم تحويلها الى داتا فريم و عرضها

```
lr.coef_ # coefficient of linear regresssion  
array([-92293.1 ,  1912.71, 24309.04, 11093.05,  2187.6 ,  2351.99  
       -1456.85, -4046.17,  1967.94])
```

```
pd.DataFrame(lr.coef_.round(2), x.columns, columns=['Coefficients'])
```

Coefficients	
customer_sector	-92293.10
supply_amount	1912.71
payment_value	24309.04
gender	11093.05
age	2187.60
status	2351.99
special_needs	-1456.85
No_of_members_family	-4046.17
income_value	1967.94

```
y_pred = lr.predict(x_test) # get predction value of linear model
```

للحصول على قيم التنبؤ لنموذج الانحدار من
الدالة `predict()`

المنهجية Method

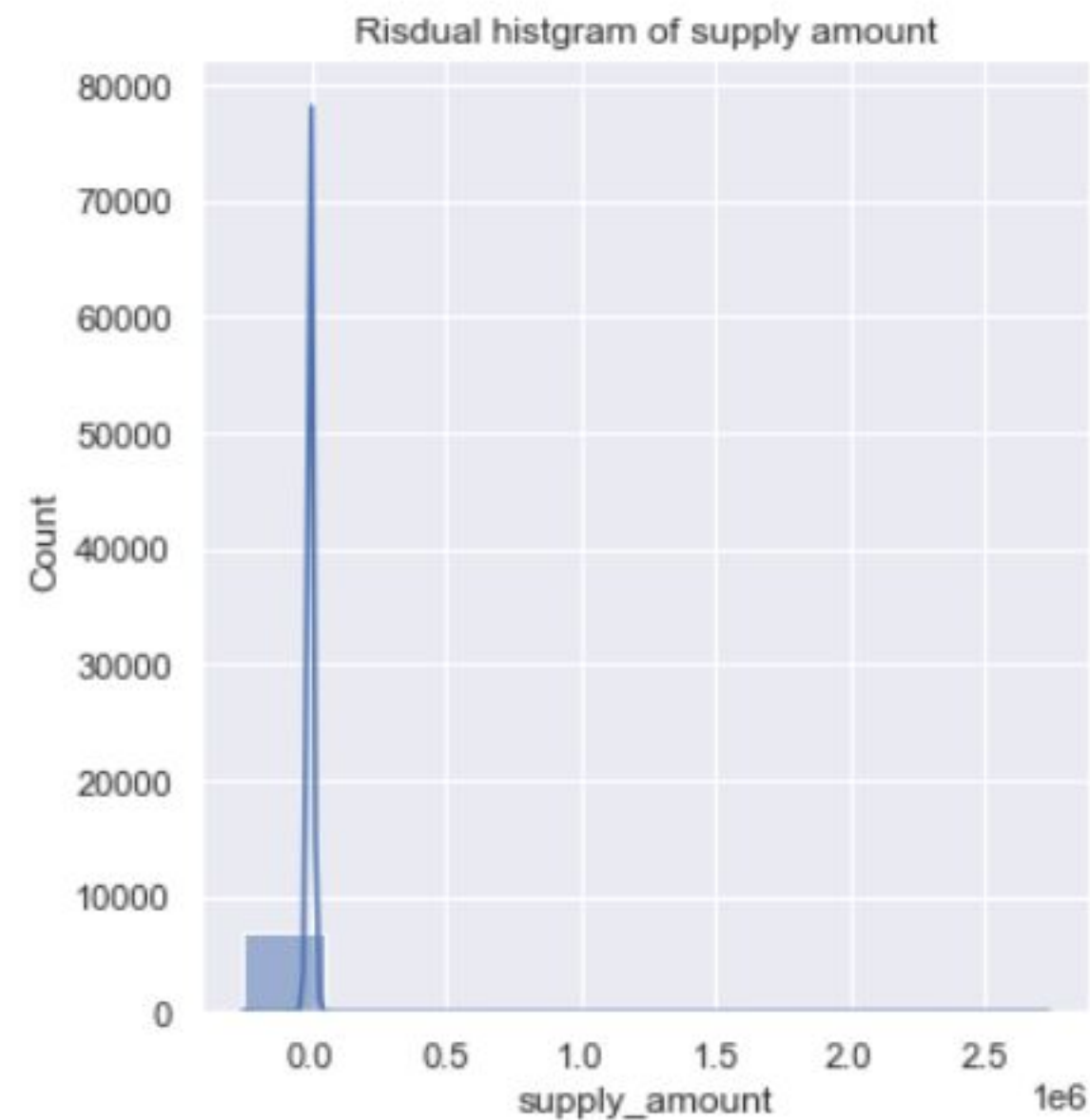
اوجدنا ($y_{residual}$) بطرح مجموعة بيانات التنبؤ (y_{pred}) من مجموعة بيانات الاختبار (y_{test})

من ثم عرض توزيع ($y_{residual}$)

```
y_residual = y_test - y_pred
```

```
sns.displot(y_residual, bins= 10, kde=True) # display histogram of resi  
plt.title(' Risdual histogram of supply amount ')
```

```
Text(0.5, 1.0, ' Risdual histogram of supply amount ')
```



المنهجية Method

```
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.3,random_state=42)
```

```
print("x_train shape = {}".format(x_train.shape)+ 'and ' + 'x_test = {}'.format( x_test.shape))  
print("y_train = {}".format(y_train.shape)+ 'and ' + 'y_test = {}'.format(y_test.shape))
```

```
x_train shape = (16201, 9) and x_test = (6944, 9)  
y_train = (16201,) and y_test = (6944,)
```

```
: # import library to apply logistic regression  
from sklearn.metrics import classification_report  
from sklearn import metrics  
from sklearn.metrics import confusion_matrix
```

```
: log = LogisticRegression(multi_class='multinomial')
```

```
: log.fit(x_train,y_train)
```

```
: LogisticRegression(multi_class='multinomial')
```

```
: log.score(x_train,y_train)
```

```
: 0.9979630886982285
```

```
: log.score(x_test,y_test)
```

```
: 0.9981278801843319
```

قسمنا البيانات الى مجموعتي التدريب و الاختبار و من ثم تم بناء نموذج الانحدار اللوجستي

اولا قمنا بتنزيل المكتبات لتطبيق الانحدار اللوجستي
و من ثم تم بناء نموذج انحدار لوجستي

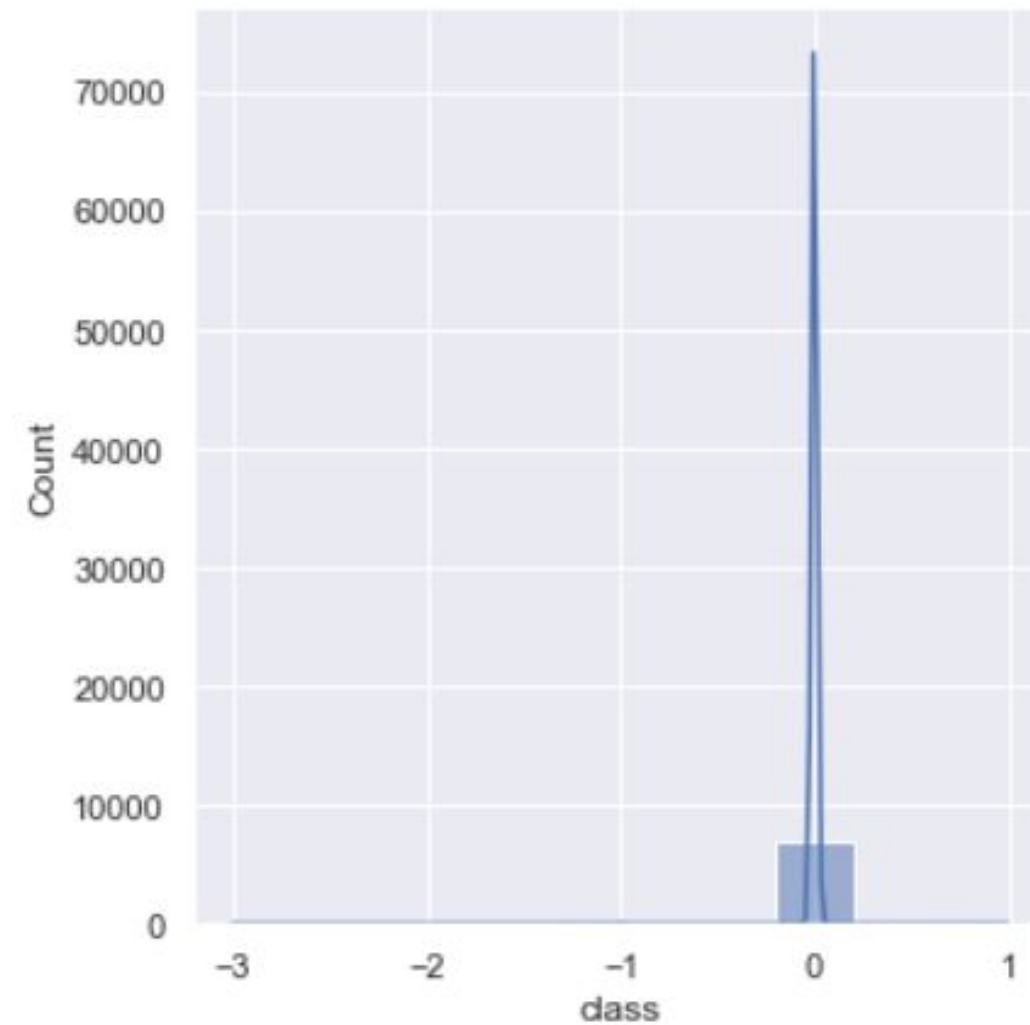
المنهجية Method

```
y_pred = log.predict(x_test)
```

```
y_residual = y_test - y_pred
```

```
sns.displot(y_residual, bins=10, kde=True)
```

```
<seaborn.axisgrid.FacetGrid at 0x159347073d0>
```



اوجدنا (y_{pred}) و ($y_{residual}$)
من ثم عرض توزيع ($y_{residual}$) الخاص بنموذج
الانحدار اللوجستي

لتقييم النموذج نستدعي `classification_report()`

```
print(classification_report(y_test,y_pred))
```

	precision	recall	f1-score	support
0	0.80	0.27	0.40	15
1	0.89	0.97	0.93	58
3	1.00	1.00	1.00	6871
accuracy			1.00	6944
macro avg	0.90	0.74	0.78	6944
weighted avg	1.00	1.00	1.00	6944

المراجع المستخدمة

1. <https://data.gov.sa/Data/ar/dataset/social-development-bank-loans-for-2021>