Sprocket Central Pty Ltd - Data Quality Report

Hi there,

This is in regards to the Dataset given from your side.

The Datasets i recieved is devided in four parts:

1. Customer Demographic
2. Customer Addresses
3. transactions
4. New Customer list The Quality Issues with theses datasets are as follows:
5. Client Demographic: a. The datatypes of the features are wrongly classified as 'object.' b. There is class imbalance of the clear cut segments in the information. c. The orientation section has multipe sort of passages for 'male' and 'female' sexes likewise some of them have spelling
   `botches.`
   d. The segment deceased_indicator have just 1 sort of information i.e., 'N.' e. The information for 'int' sort of the information isn't 'ordinary.' f. This dataframe contains previously mentioned invalid upsides of which 2 sections' qualities are not neglible.
   `So we demand you kindly give some more data about it.`
6. Client Addresses: a. The datatypes of the dataframe are inaccurately named 'object.' b. There is class awkwardness of the clear cut segments in the information. c. The information in section 'country' has just 1 worth i.e., 'Australia.' d. The information in segment 'property' isn't 'regularly circulated.'
7. Transaction: a. The datatypes of the dataframe are erroneously delegated 'object.' b. There is 'class irregularity' of the clear cut sections in the information. c. The information of the mathematical sections isn't 'ordinary.' d. There are '1.8 and less invalid qualities' in the dataset. e. The 'mean list_price' and 'standard_cost' by 'product_size , product_line and product_class' isn't 'same' for
   `a few sections. So there is some sort of 'logical inconsistency' in the information.`
8. New Customer Data: a. The datatypes of the dataframe are mistakenly delegated object. b. Likewise the section product_first_sold_date are the digits which are succeed dates and should be changed over completely to
   `short date design.`
   c. There is class irregularity of the absolute sections in the information. d. The segment deceased_indicator have just 1 sort of information i.e., 'N.' e. Likewise there are a few sections named 'NaN' which are a few computations done.
   ```
   These estimations are to be deciphered by us or the data with respect to the equivalent ought to be asked to
   the client. The part that isn't justifiable is that they took the 'irregular' number for the computations.
   In view of which the Rank section is made. Along these lines, this section isn't legitimate!
   ```
   f. The information for a portion of the unmitigated sections isn't 'typical.' g. This dataframe contains previously mentioned invalid upsides of which 2 segments' qualities are not neglible.

Taking into account on these issues, I request you to look into this and if it's not too much trouble, give greatest conceivable data. So that we can further proceed with our examination more throughly.

i am also attaching the quality report for the reference.

Much obliged and Regards.

Pratiksha patil.

Importing neccessary Libraries

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

In [2]:
```
import warnings
warnings.filterwarnings('ignore')
```

Dataframe 1

In [3]:
```
transaction = pd.read_excel('KPMG_VI_New_raw_data_update_final.xlsx',sheet_name='Transactions')
```

In [4]:
```
transaction.rename(columns=transaction.iloc[0],inplace=True)
```

In [5]:
```
transaction.drop(index=0,inplace=True)
```

In [6]:
```
transaction.head()
```

Out[6]:

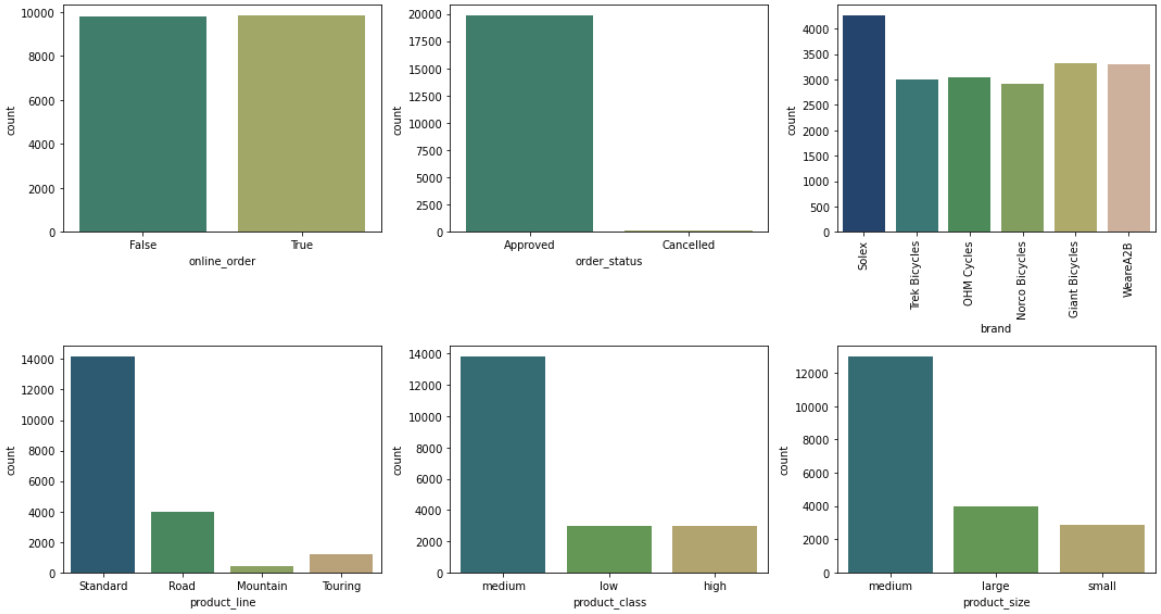
	transaction_id	product_id	customer_id	transaction_date	online_order	order_status	brand	product_line	product_class	product_size	list_price	standard_cost	product_first_sold_date
1	1	2	2950	2017-02-25 00:00:00	False	Approved	Solex	Standard	medium	medium	71.49	53.62	2012-12-02 00:00:00
2	2	3	3120	2017-05-21 00:00:00	True	Approved	Trek Bicycles	Standard	medium	large	2091.47	388.92	2014-03-03 00:00:00
3	3	37	402	2017-10-16 00:00:00	False	Approved	OHM Cycles	Standard	low	medium	1793.43	248.82	1999-07-20 00:00:00
4	4	88	3135	2017-08-31 00:00:00	False	Approved	Norco Bicycles	Standard	medium	medium	1198.46	381.1	1998-12-16 00:00:00
5	5	78	787	2017-10-01 00:00:00	True	Approved	Giant Bicycles	Standard	medium	large	1765.3	709.48	2015-08-10 00:00:00

The datatypes of the dataframe are incorrectly classified as `object.`
Also the column `product_first_sold_date` are the digits which are `excel dates` and need to be converted to `short date` format.

💫Accuracy (Correct Values)

```
gsize=(15,8))
```

```
n=1
for i in transaction[['online_order' , 'order_status' , 'brand' , 'product_line' , 'product_class' , 'product_size']].columns:
    plt.subplot(2,3,n)
    n+=1
    sns.countplot(transaction[i],palette='gist_earth')
    if len(transaction[i].value_counts()) > 4:
        plt.xticks(rotation=90)
plt.tight_layout()
plt.show()
```



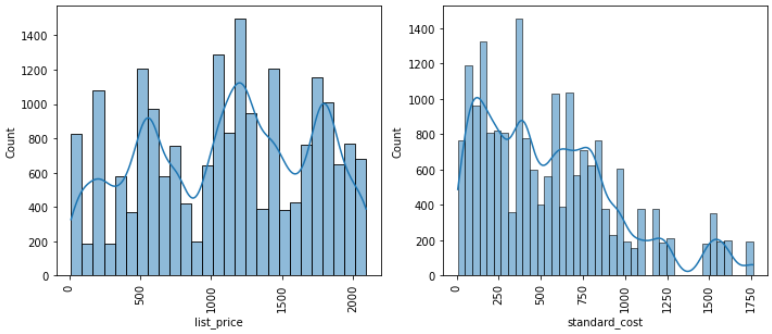
As we can see, there is `class imbalance` in the data.

In [8]:
```
transaction.dtypes
```

Out[8]:
```
transaction_id             object
product_id                 object
customer_id                object
transaction_date           object
online_order               object
order_status               object
brand                      object
product_line               object
product_class              object
product_size               object
list_price                 object
standard_cost              object
product_first_sold_date    object
dtype: object
```

In [ ]:
```
```

In [9]:
```
plt.figure(figsize=(15,8))
n=1
for i in transaction[[ 'list_price','standard_cost']].dropna().astype('int').columns:
    plt.subplot(2,3,n)
    n+=1
    sns.histplot(transaction[i],palette='gist_earth' , kde=True)
    if len(transaction[i].value_counts()) > 4:
        plt.xticks(rotation=90)
plt.tight_layout()
plt.show()
```



As we can see, the data for int type of the data is `not normal.`

💫Completeness (Data Fields with Values)

In [10]:
```
print('Column Name\t\tNull Values Percentage')
print(transaction.isnull().sum() / len(transaction) * 100)
print('\nThis dataframe contains above mentioned null values. Which are neglible and can be dropped.')
```

```
Column Name             Null Values Percentage
transaction_id             0.000
product_id                 0.000
                           0.000
```

```
transaction_date           0.000
online_order               1.800
order_status               0.000
brand                      0.985
product_line               0.985
product_class              0.985
product_size               0.985
list_price                 0.000
standard_cost              0.985
product_first_sold_date    0.985
dtype: float64
```

This dataframe contains above mentioned null values. Which are neglible and can be dropped.

💫Consistency (Values free from contradiction)

In [11]:
```
transaction['list_price']=transaction['list_price'].astype('float')
transaction['standard_cost']=transaction['standard_cost'].astype('float')
```

In [12]:
```
transaction.groupby(['product_size','product_line','product_class']).mean()[['list_price','standard_cost']]
```

Out[12]:

product_size	product_line	product_class	list_price	standard_cost
large	Road	high	389.319119	233.594482
		medium	1633.080333	731.503048
	Standard	high	1842.920000	1105.750000
		medium	1457.405335	552.551388
	Touring	medium	1808.898470	449.254211
medium	Mountain	low	574.640000	459.710000
	Road	low	980.370000	234.430000
		medium	757.090498	495.496054
	Standard	high	1019.530301	611.718295
		low	912.564852	335.496576
		medium	1002.618560	367.519121
	Touring	low	1073.070000	933.840000
		medium	1466.680000	363.250000
small	Mountain	low	688.630000	612.880000
	Road	low	1131.447739	1006.985728
		medium	1758.834286	1565.361169
	Standard	high	1824.646984	1623.936190
		medium	1129.478565	1001.811568

The `mean list_price` and `standard_cost` by `product_size , product_line & product_class` is `not same` for some entries. So there is some kind of `contradiction` in the data.

💫Currency (Values up to date)

In [13]:
```
print(f"The data is between the dates : {pd.to_datetime(transaction['product_first_sold_date']).max()} and {pd.to_datetime(transaction['product_first_sold_date
```

```
The data is between the dates : 2016-12-06 00:00:00 and 1991-01-21 00:00:00
```

💫Uniqueness (Records that are Duplicated.)

In [15]:
```
print(f'The duplicate data in the dataframes are {transaction.duplicated().sum()}')
```

```
The duplicate data in the dataframes are 0
```

Dataframe 2

In [16]:
```
NewCustomer = pd.read_excel('KPMG_VI_New_raw_data_update_final.xlsx',sheet_name='NewCustomerList')
```

In [17]:
```
NewCustomer.rename(columns=NewCustomer.iloc[0],inplace=True)
```

In [18]:
```
NewCustomer.drop(index=0,inplace=True)
```

In [19]:
```
NewCustomer.head()
```

Out[19]:

	first_name	last_name	gender	past_3_years_bike_related_purchases	DOB	job_title	job_industry_category	wealth_segment	deceased_indicator	owns_car	...	state	country	property_valu
1	Chickie	Brister	Male	86	1957-07-12	General Manager	Manufacturing	Mass Customer	N	Yes	...	QLD	Australia	
2	Morly	Genery	Male	69	1970-03-22	Structural Engineer	Property	Mass Customer	N	No	...	NSW	Australia	
3	Ardelis	Forrester	Female	10	1974-08-28 00:00:00	Senior Cost Accountant	Financial Services	Affluent Customer	N	No	...	VIC	Australia	
4	Lucine	Stutt	Female	64	1979-01-28	Account Representative III	Manufacturing	Affluent Customer	N	Yes	...	QLD	Australia	
5	Melinda	Hadlee	Female	34	1965-09-21	Financial Analyst	Financial Services	Affluent Customer	N	No	...	NSW	Australia	

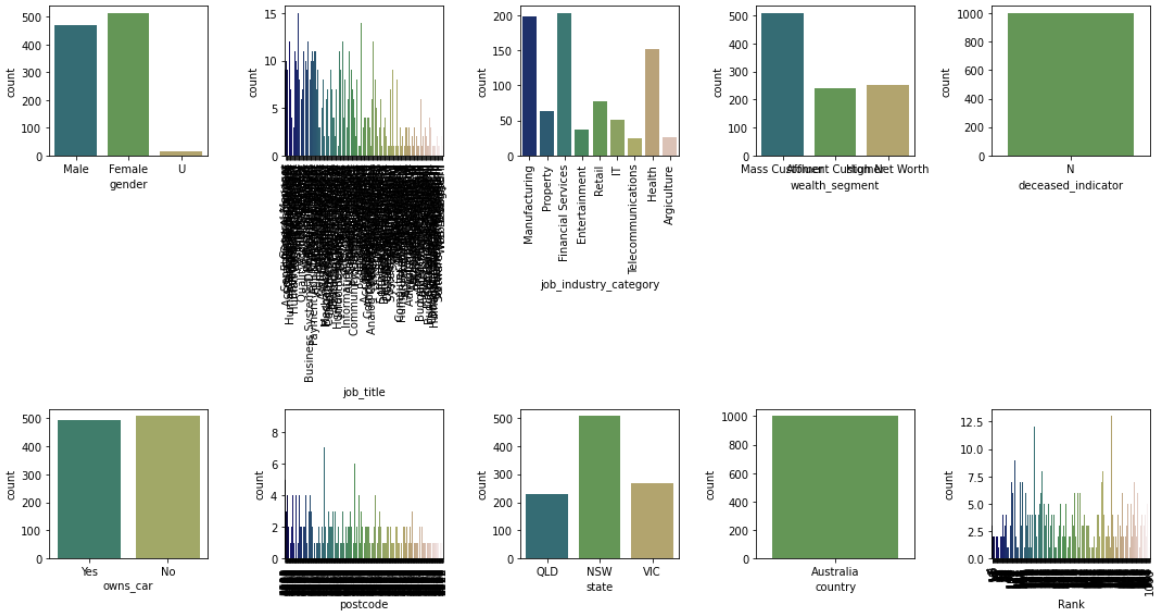
5 rows × 23 columns

The datatypes of the dataframe are incorrectly classified as `object.`
Also the column `product_first_sold_date` are the digits which are `excel dates` and need to be converted to `short date` format.

Also this dataset doesn't have any primary key to join with the other datas.
So we'll have to get the `Customer id` s for these new customers

💫Accuracy (Correct Values)

In [20]:
```
plt.figure(figsize=(15,8))
n=1
for i in NewCustomer[['gender' , 'job_title' , 'job_industry_category' , 'wealth_segment' , 'deceased_indicator' , 'owns_car' , 'postcode' , 'state' , 'country
    plt.subplot(2,5,n)
    n+=1
    sns.countplot(NewCustomer[i],palette='gist_earth')
    if len(NewCustomer[i].value_counts()) > 4:
        plt.xticks(rotation=90)
plt.tight_layout()
plt.show()
```



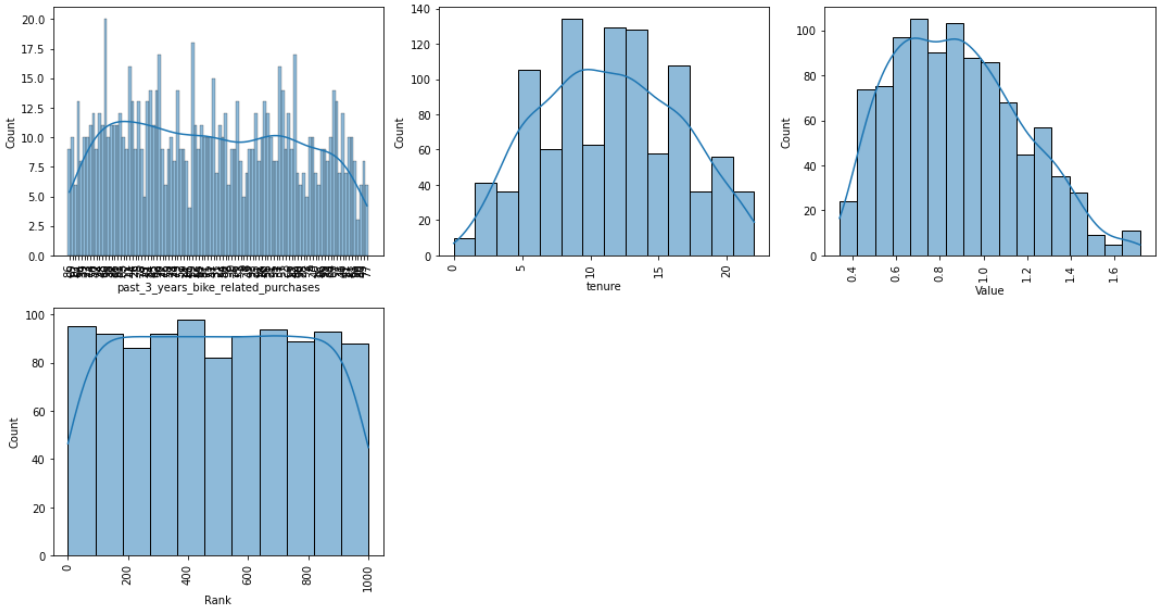
As we can see, there is `class imbalance` in the data.
The columns `country` and `deceased_indicator` have only 1 type of data i.e., `Australia` and `N` respectively.

In [21]:
```
NewCustomer.dtypes
```

Out[21]:
```
first_name                           object
last_name                            object
gender                               object
past_3_years_bike_related_purchases  object
DOB                                  object
job_title                            object
job_industry_category                object
wealth_segment                       object
deceased_indicator                   object
owns_car                             object
tenure                               object
address                              object
postcode                             object
state                                object
country                              object
property_valuation                   object
NaN                                 float64
NaN                                 float64
NaN                                 float64
NaN                                 float64
NaN                                 float64
Rank                                 object
Value                                object
dtype: object
```

Also there are some columns named `NaN` which are some calculations done.
These calculations are to be interpreted by us or the information regarding the same should be asked to the client.
The part that is not understandable is that they took the `random` number for the calculations.
Based on which the **Rank** column is made. **So, this column is not valid!**

In [22]:
```
plt.figure(figsize=(15,8))
n=1
for i in NewCustomer[[ 'past_3_years_bike_related_purchases','tenure' , 'Value' , 'Rank']].dropna().astype('float').columns:
    plt.subplot(2,3,n)
    n+=1
    sns.histplot(NewCustomer[i],palette='gist_earth' , kde=True)
    if len(NewCustomer[i].value_counts()) > 4:
        plt.xticks(rotation=90)
plt.tight_layout()
plt.show()
```



As we can see, the data for int type of the data is `not normal.`

💫Completeness (Data Fields with Values)

In [ ]:
```
print('Column Name\t\tNull Values Percentage')
print(NewCustomer.isnull().sum() / len(NewCustomer) * 100)
print("\nThis dataframe contains above mentioned null values of which 2 columns' values are not neglible and need to be properly imputed.")
```

💫Uniqueness (Records that are Duplicated.)

In [23]:
```
print(f'The duplicate data in the dataframes are {NewCustomer.duplicated().sum()}')
```

```
The duplicate data in the dataframes are 0
```

In [54]:
```
NewCustomer['DOB']=NewCustomer['DOB'].astype('datetime64')
```

In [61]:
```
NewCustomer['DOB'].
```

Out[61]:
```
Timestamp('1938-06-08 00:00:00')
```

Dataframe 3

In [24]:
```
CustDemographic = pd.read_excel('KPMG_VI_New_raw_data_update_final.xlsx',sheet_name='CustomerDemographic')
```

In [25]:
```
CustDemographic.rename(columns=CustDemographic.iloc[0],inplace=True)
```

In [26]:
```
CustDemographic.drop(index=0,inplace=True)
```

In [27]:
```
CustDemographic.head()
```

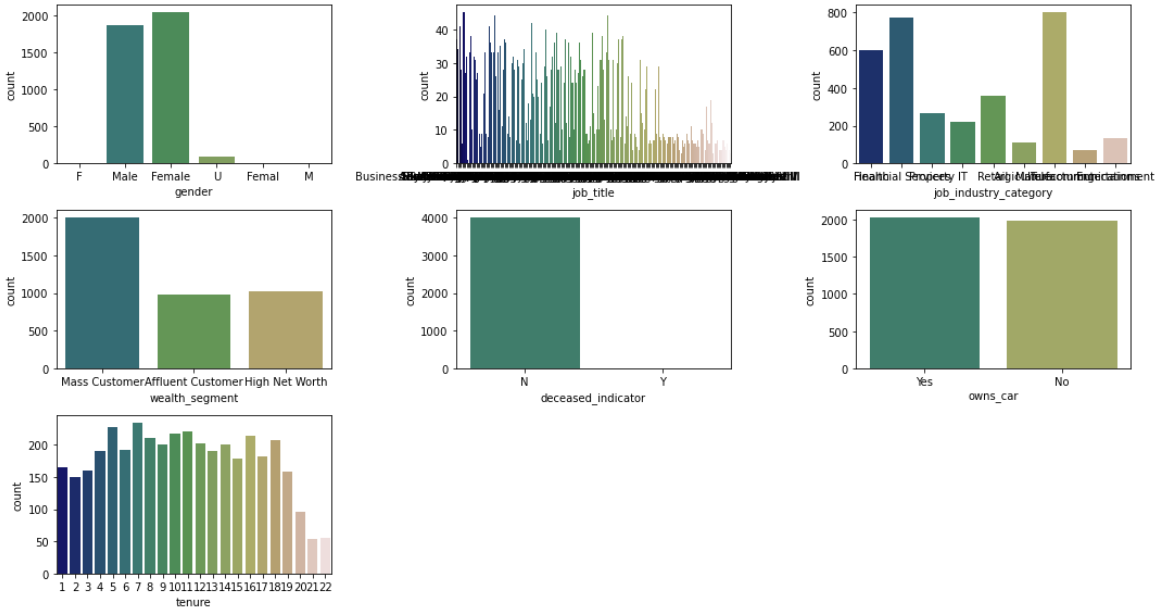
Out[27]:

	customer_id	first_name	last_name	gender	past_3_years_bike_related_purchases	DOB	job_title	job_industry_category	wealth_segment	deceased_indicator	default	owns_car	te
1	1	Laraine	Medendorp	F	93	1953-10-12 00:00:00	Executive Secretary	Health	Mass Customer	N	""	Yes	
2	2	Eli	Bockman	Male	81	1980-12-16 00:00:00	Administrative Officer	Financial Services	Mass Customer	N	<script>alert('hi') </script>	Yes	
3	3	Arlin	Dearle	Male	61	1954-01-20 00:00:00	Recruiting Manager	Property	Mass Customer	N	2018-02-01 00:00:00	Yes	
4	4	Talbot	NaN	Male	33	1961-10-03 00:00:00	NaN	IT	Mass Customer	N	() { _; } >_[$($())] { touch /tmp/blns.shellsh...	No	
5	5	Sheila-kathryn	Calton	Female	56	1977-05-13 00:00:00	Senior Editor	NaN	Affluent Customer	N	NIL	Yes	

The datatypes of the dataframe are incorrectly classified as `object.`

💫Accuracy (Correct Values)

In [28]:
```
plt.figure(figsize=(15,8))
n=1
for i in CustDemographic.drop(columns=['customer_id' , 'past_3_years_bike_related_purchases' , 'DOB' , 'first_name','last_name' , 'default']).columns:
    plt.subplot(3,3,n)
    n+=1
    sns.countplot(CustDemographic[i],palette='gist_earth')
plt.tight_layout()
plt.show()
```

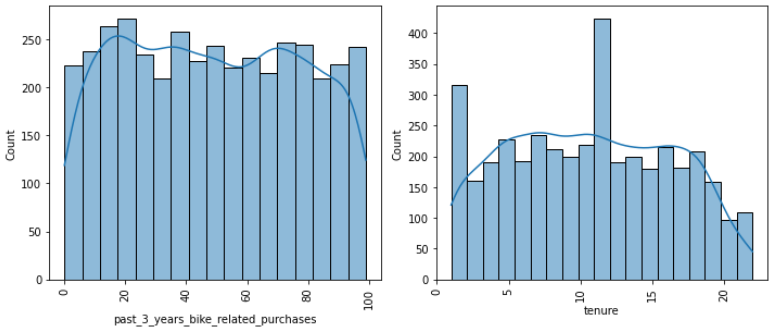


As we can see, there is `class imbalance` in the data.
The gender column has multipe type of entries for `male` and `female` genders also some of them have spelling mistakes. So we need to clean the data.
The column deceased_indicator have only 1 type of data i.e., `N.`

In [29]:
```
CustDemographic.dtypes
```

Out[29]:
```
customer_id                          object
first_name                           object
last_name                            object
gender                               object
past_3_years_bike_related_purchases  object
DOB                                  object
job_title                            object
job_industry_category                object
wealth_segment                       object
deceased_indicator                   object
default                              object
owns_car                             object
tenure                               object
dtype: object
```

In [30]:
```
plt.figure(figsize=(15,8))
n=1
for i in CustDemographic[[ 'past_3_years_bike_related_purchases','tenure']].dropna().astype('int').columns:
    plt.subplot(2,3,n)
    n+=1
    sns.histplot(CustDemographic[i],palette='gist_earth' , kde=True)
    if len(CustDemographic[i].value_counts()) > 4:
        plt.xticks(rotation=90)
plt.tight_layout()
plt.show()
```



As we can see, the data for int type of the data is `not normal.`

💫Completeness (Data Fields with Values)

In [31]:
```
print('Column Name\t\tNull Values Percentage')
print(CustDemographic.isnull().sum() / len(CustDemographic) * 100)
print("\nThis dataframe contains above mentioned null values of which 2 columns' values are not neglible and need to be properly imputed.")
```

```
Column Name             Null Values Percentage
customer_id                              0.000
first_name                               0.000
last_name                                3.125
gender                                   0.000
past_3_years_bike_related_purchases      0.000
DOB                                      2.175
job_title                               12.650
job_industry_category                   16.400
wealth_segment                           0.000
deceased_indicator                       0.000
default                                  7.550
                                         0.000
```

```
tenure                                  2.175
dtype: float64
```

This dataframe contains above mentioned null values of which 2 columns' values are not neglible and need to be properly imputed.

💫Uniqueness (Records that are Duplicated.)

In [32]:
```
print(f'The duplicate data in the dataframes are {CustDemographic.duplicated().sum()}')
```

```
The duplicate data in the dataframes are 0
```

Dataframe 4

In [33]:
```
CustAddress = pd.read_excel('KPMG_VI_New_raw_data_update_final.xlsx',sheet_name='CustomerAddress')
```

In [34]:
```
CustAddress.rename(columns=CustAddress.iloc[0],inplace=True)
```

In [35]:
```
CustAddress.drop(index=0,inplace=True)
```

In [36]:
```
CustAddress.head()
```

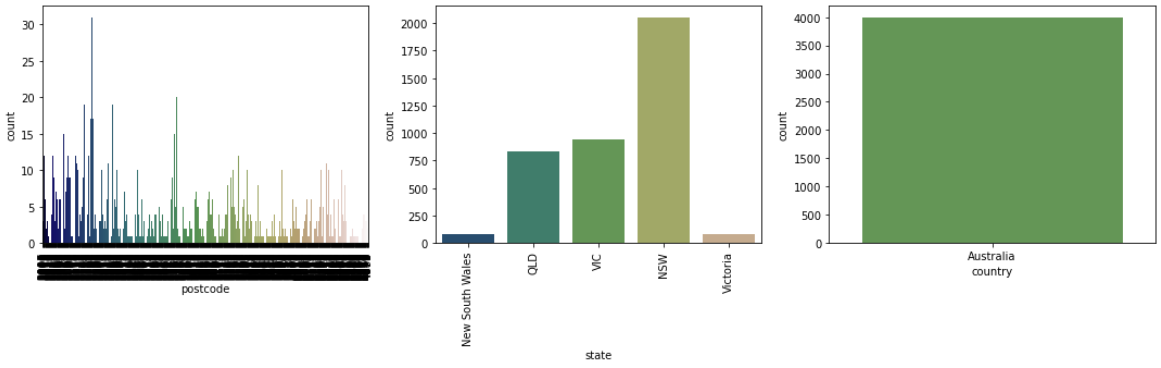
Out[36]:

	customer_id	address	postcode	state	country	property_valuation
1	1	060 Morning Avenue	2016	New South Wales	Australia	10
2	2	6 Meadow Vale Court	2153	New South Wales	Australia	10
3	4	0 Holy Cross Court	4211	QLD	Australia	9
4	5	17979 Del Mar Point	2448	New South Wales	Australia	4
5	6	9 Oakridge Court	3216	VIC	Australia	9

The datatypes of some of the dataframe columns are incorrectly classified as `object.`

💫Accuracy (Correct Values)

In [37]:
```
plt.figure(figsize=(15,8))
n=1
for i in [ 'postcode' , 'state' , 'country']:
    plt.subplot(2,3,n)
    n+=1
    sns.countplot(CustAddress[i],palette='gist_earth')
    if len(CustAddress[i].value_counts()) > 4:
        plt.xticks(rotation=90)
plt.tight_layout()
plt.show()
```

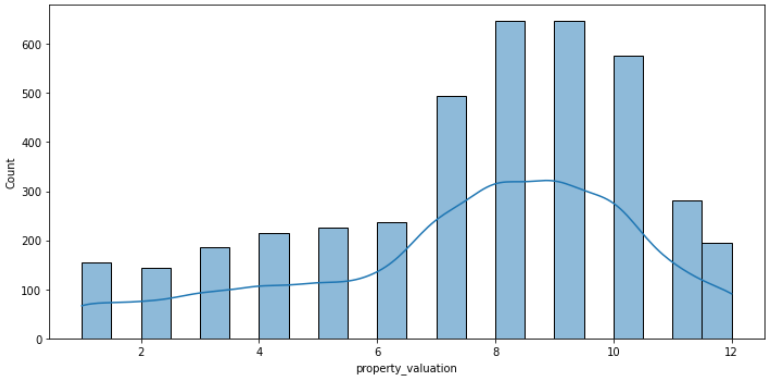


As we can see, there is `class imbalance` in the data.
The column `country` has only 1 value i.e., `Australia.`

In [38]:
```
CustAddress.dtypes
```

Out[38]:
```
customer_id           object
address               object
postcode              object
state                 object
country               object
property_valuation    object
dtype: object
```

In [39]:
```
plt.figure(figsize=(10,5))
sns.histplot(CustAddress['property_valuation'],palette='gist_earth' , kde=True)
if len(CustAddress[i].value_counts()) > 4:
    plt.xticks(rotation=90)
plt.tight_layout()
plt.show()
```



As we can see, the data for int type of the data is `not normal.`

💫Completeness (Data Fields with Values)

In [40]:
```
print('Column Name\tNull Values Percentage')
print(CustAddress.isnull().sum() / len(CustAddress) * 100)
print('\nThis dataframe does not contain any null values.')
```

```
Column Name     Null Values Percentage
customer_id           0.0
address               0.0
postcode              0.0
state                 0.0
country               0.0
property_valuation    0.0
dtype: float64

This dataframe does not contain any null values.
```

💫Uniqueness (Records that are Duplicated.)

In [41]:
```
print(f'The duplicate data in the dataframes are {CustAddress.duplicated().sum()}')
```

```
The duplicate data in the dataframes are 0
```

Data Snippets

In [42]:
```
transaction.head()
```

Out[42]:

	transaction_id	product_id	customer_id	transaction_date	online_order	order_status	brand	product_line	product_class	product_size	list_price	standard_cost	product_first_sold_date
1	1	2	2950	2017-02-25 00:00:00	False	Approved	Solex	Standard	medium	medium	71.49	53.62	2012-12-02 00:00:00
2	2	3	3120	2017-05-21 00:00:00	True	Approved	Trek Bicycles	Standard	medium	large	2091.47	388.92	2014-03-03 00:00:00
3	3	37	402	2017-10-16 00:00:00	False	Approved	OHM Cycles	Standard	low	medium	1793.43	248.82	1999-07-20 00:00:00
4	4	88	3135	2017-08-31 00:00:00	False	Approved	Norco Bicycles	Standard	medium	medium	1198.46	381.10	1998-12-16 00:00:00
5	5	78	787	2017-10-01 00:00:00	True	Approved	Giant Bicycles	Standard	medium	large	1765.30	709.48	2015-08-10 00:00:00

In [43]:
```
NewCustomer.head()
```

Out[43]:

	first_name	last_name	gender	past_3_years_bike_related_purchases	DOB	job_title	job_industry_category	wealth_segment	deceased_indicator	owns_car	...	state	country	property_valu
1	Chickie	Brister	Male	86	1957-07-12	General Manager	Manufacturing	Mass Customer	N	Yes	...	QLD	Australia	
2	Morly	Genery	Male	69	1970-03-22	Structural Engineer	Property	Mass Customer	N	No	...	NSW	Australia	
3	Ardelis	Forrester	Female	10	1974-08-28 00:00:00	Senior Cost Accountant	Financial Services	Affluent Customer	N	No	...	VIC	Australia	
4	Lucine	Stutt	Female	64	1979-01-28	Account Representative III	Manufacturing	Affluent Customer	N	Yes	...	QLD	Australia	
5	Melinda	Hadlee	Female	34	1965-09-21	Financial Analyst	Financial Services	Affluent Customer	N	No	...	NSW	Australia	

5 rows × 23 columns

In [44]:
```
CustDemographic.head()
```

Out[44]:

	customer_id	first_name	last_name	gender	past_3_years_bike_related_purchases	DOB	job_title	job_industry_category	wealth_segment	deceased_indicator	default	owns_car	t
1	1	Laraine	Medendorp	F	93	1953-10-12 00:00:00	Executive Secretary	Health	Mass Customer	N	"'	Yes	
2	2	Eli	Bockman	Male	81	1980-12-16 00:00:00	Administrative Officer	Financial Services	Mass Customer	N	<script>alert('hi')</script>	Yes	
3	3	Arlin	Dearle	Male	61	1954-01-20 00:00:00	Recruiting Manager	Property	Mass Customer	N	2018-02-01 00:00:00	Yes	
4	4	Talbot	NaN	Male	33	1961-10-03 00:00:00	NaN	IT	Mass Customer	N	() { _; } >_[$($())] { touch /tmp/blns.shellsh...	No	
		Sheila-				1977-							

In [46]:
```
CustAddress.head()
```

Out[46]:

	customer_id	address	postcode	state	country	property_valuation
1	1	060 Morning Avenue	2016	New South Wales	Australia	10
2	2	6 Meadow Vale Court	2153	New South Wales	Australia	10
3	4	0 Holy Cross Court	4211	QLD	Australia	9
4	5	17979 Del Mar Point	2448	New South Wales	Australia	4
5	6	9 Oakridge Court	3216	VIC	Australia	9

----------End of Report----------