02_Categorical_Variables

September 4, 2021

0.1 What is a Variable?

A variable is any characteristic, number, or quantity that can be measured or counted. They are called 'variables' because the value they take may vary, and it usually does. The following are examples of variables:

- Age (21, 35, 62, …)
- Gender (male, female)
- Income (GBP 20000, GBP 35000, GBP 45000, …)
- House price (GBP 350000, GBP 570000, …)
- Country of birth (China, Russia, Costa Rica, …)
- Eye colour (brown, green, blue, …)
- Vehicle make (Ford, Volkswagen, …)

Most variables in a data set can be classified into one of two major types:

- **Numerical variables**
- **Categorical variables**

==========

0.2 Categorical Variables

The values of a categorical variable are selected from a group of **categories**, also called **labels**. Examples are gender (male or female) and marital status (never married, married, divorced or widowed). Other examples of categorical variables include:

- Intended use of loan (debt-consolidation, car purchase, wedding expenses, …)
- Mobile network provider (Vodafone, Orange, …)
- Postcode

Categorical variables can be further categorised into:

- **Ordinal Variables**
- **Nominal variables**

0.2.1 Ordinal Variable

Ordinal variables are categorical variable in which the categories can be meaningfully ordered. For example:

- Student's grade in an exam (A, B, C or Fail).
- Days of the week, where Monday = 1 and Sunday = 7.

- Educational level, with the categories Elementary school, High school, College graduate and PhD ranked from 1 to 4.

0.2.2 Nominal Variable

For nominal variables, there isn't an intrinsic order in the labels. For example, country of birth, with values Argentina, England, Germany, etc., is nominal. Other examples of nominal variables include:

- Car colour (blue, grey, silver, …)
- Vehicle make (Citroen, Peugeot, …)
- City (Manchester, London, Chester, …)

There is nothing that indicates an intrinsic order of the labels, and in principle, they are all equal.

To be considered:

Sometimes categorical variables are coded as numbers when the data are recorded (e.g. gender may be coded as 0 for males and 1 for females). The variable is still categorical, despite the use of numbers.

In a similar way, individuals in a survey may be coded with a number that uniquely identifies them (for example to avoid storing personal information for confidentiality). This number is really a label, and the variable then categorical. The number has no meaning other than making it possible to uniquely identify the observation (in this case the interviewed subject).

Ideally, when we work with a dataset in a business scenario, the data will come with a dictionary that indicates if the numbers in the variables are to be considered as categories or if they are numerical. And if the numbers are categories, the dictionary would explain what each value in the variable represents.

==========

0.3 In this demo: Peer to peer lending (Finance)

In this demo, we will use a toy data set which simulates data from a peer-o-peer finance company to inspect discrete and continuous numerical variables.

- You should have downloaded the **Datasets** together with the Jupyter notebooks in **Section 1**.

[1]:
```
import pandas as pd

import matplotlib.pyplot as plt
```

[2]:
```
# let's load the dataset

# Variable definitions:
#-------------------------
# loan_purpose: intended use of the loan
# market: the risk market assigned to the borrower (based in their financial␣
 ↪situation)
```

```
# householder: whether the borrower owns or rents their property

data = pd.read_csv('./loan.csv')

data.head()
```

[2]:

```
   customer_id  disbursed_amount  interest market    employment time_employed  \
0            0           23201.5   15.4840      C       Teacher     <=5 years
1            1            7425.0   11.2032      B    Accountant     <=5 years
2            2           11150.0    8.5100      A  Statistician     <=5 years
3            3            7600.0    5.8656      A         Other     <=5 years
4            4           31960.0   18.7392      E    Bus driver      >5 years

  householder    income date_issued  target        loan_purpose  \
0        RENT   84600.0  2013-06-11       0  Debt consolidation
1       OWNER  102000.0  2014-05-08       0        Car purchase
2        RENT   69840.0  2013-10-26       0  Debt consolidation
3        RENT  100386.0  2015-08-20       0  Debt consolidation
4        RENT   95040.0  2014-07-22       0  Debt consolidation

   number_open_accounts date_last_payment  number_credit_lines_12
0                     4        2016-01-14                     NaN
1                    13        2016-01-25                     NaN
2                     8        2014-09-26                     NaN
3                    20        2016-01-26                     NaN
4                    14        2016-01-11                     NaN
```

[3]:

```
# let's inspect the variable householder,
# which indicates whether the borrowers own their home
# or if they are renting, among other things.

data['householder'].unique()
```

[3]:

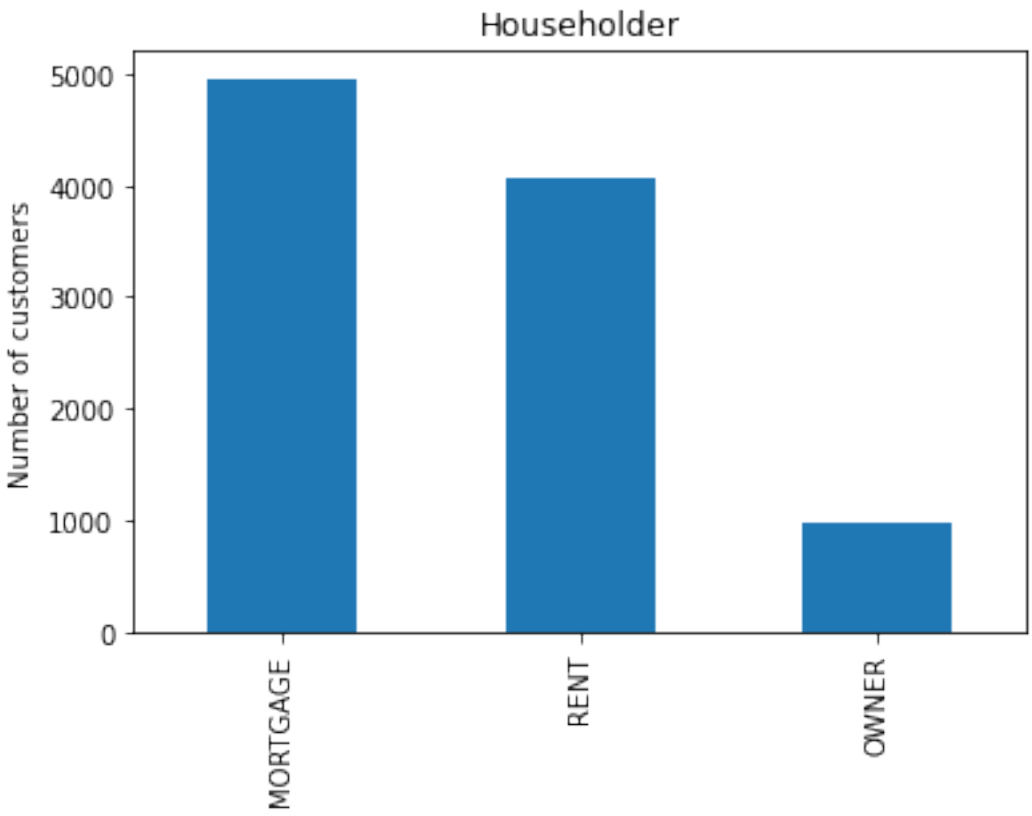
```
array(['RENT', 'OWNER', 'MORTGAGE'], dtype=object)
```

[4]:

```
# let's make a bar plot, with the number of loans
# for each category of home ownership

# the code below counts the number of observations (borrowers)
# within each category and then makes a bar plot

fig = data['householder'].value_counts().plot.bar()
fig.set_title('Householder')
fig.set_ylabel('Number of customers')
```

[4]:

```
Text(0, 0.5, 'Number of customers')
```



The majority of the borrowers either own their house on a mortgage or rent their property. A few borrowers own their home completely.

[5]:
```
data['householder'].value_counts()
```

[5]:
```
MORTGAGE    4957
RENT        4055
OWNER        988
Name: householder, dtype: int64
```

[6]:
```
# the "loan_purpose" variable is another categorical variable
# that indicates how the borrowers intend to use the
# money they are borrowing, for example to improve their
# house, or to cancel previous debt.

data['loan_purpose'].unique()
```

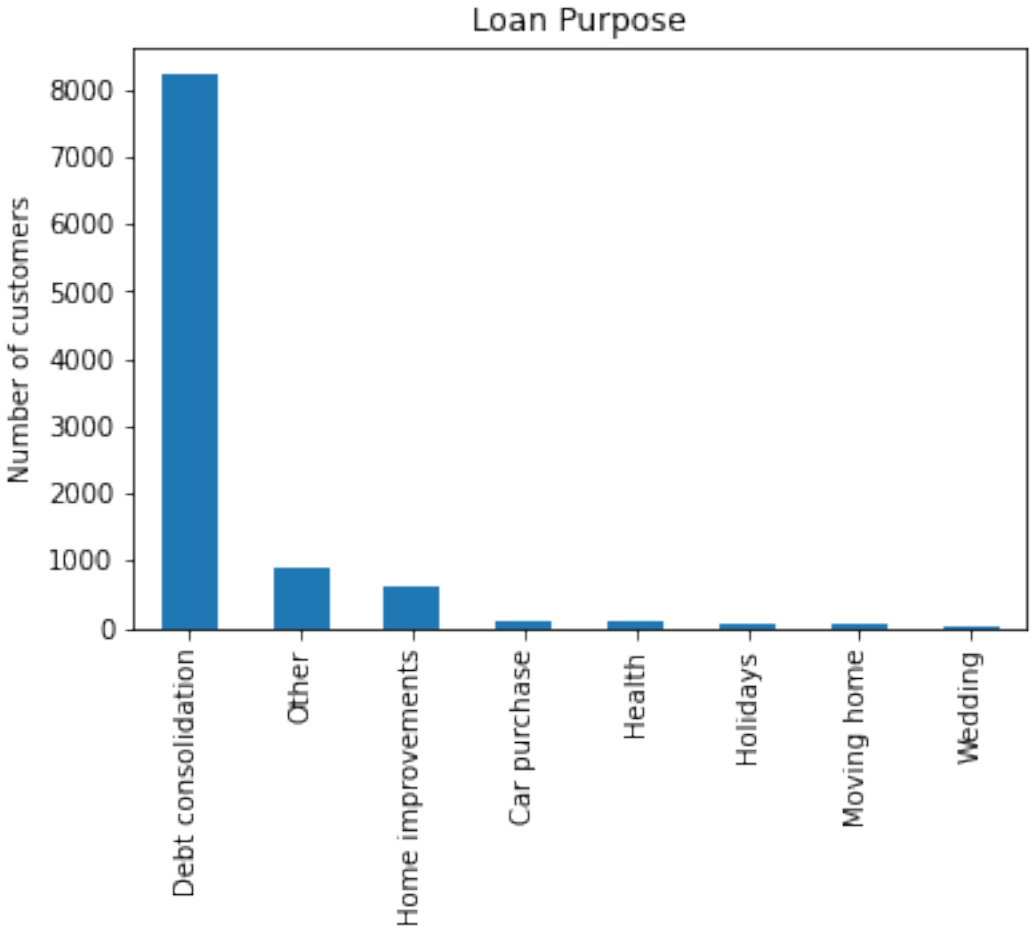
[6]:
```
array(['Debt consolidation', 'Car purchase', 'Other', 'Home improvements',
       'Moving home', 'Health', 'Holidays', 'Wedding'], dtype=object)
```

Debt consolidation means that the borrower would like a loan to cancel previous debts, Car purchase means that the borrower is borrowing the money to buy a car, and so on. It gives an idea of the

intended use of the loan.

[7]:
```
# let's make a bar plot with the number of borrowers
# within each category

# the code below counts the number of observations (borrowers)
# within each category and then makes a plot

fig = data['loan_purpose'].value_counts().plot.bar()
fig.set_title('Loan Purpose')
fig.set_ylabel('Number of customers')
```

[7]:
```
Text(0, 0.5, 'Number of customers')
```



The majority of the borrowers intend to use the loan for 'debt consolidation'. This is quite common. What the borrowers intend to do is, to consolidate all the debt that they have on different financial items, in one single debt, the new loan that they will take from the peer to peer company. This loan will usually provide an advantage to the borrower, either in the form of lower interest rates

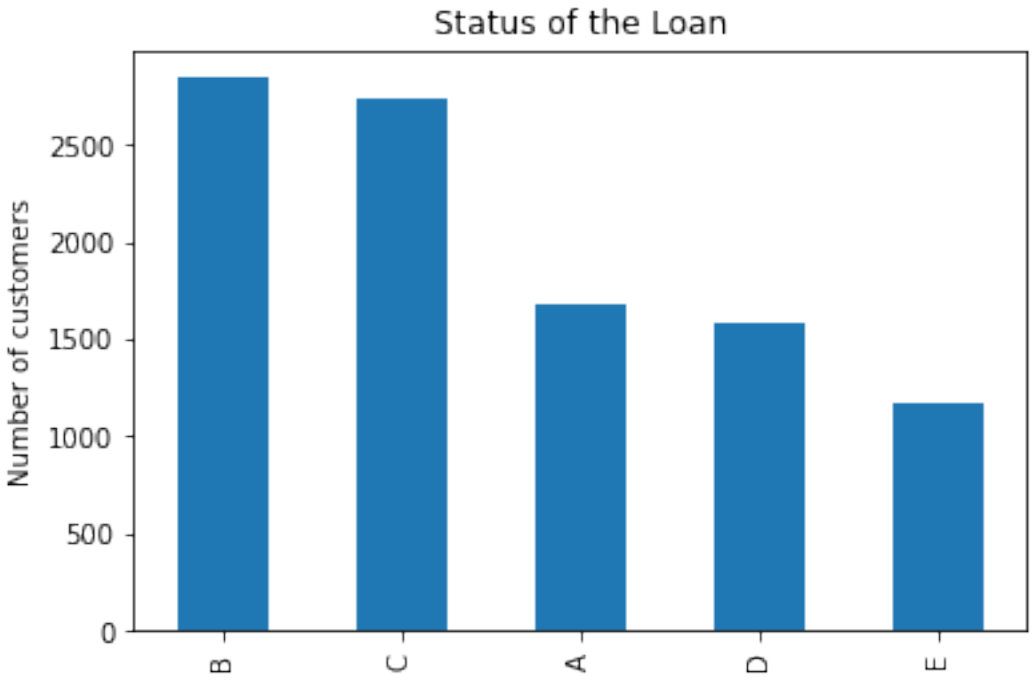
than a credit card, for example, or longer repayment period.

[8]:
```
# let's look at one additional categorical variable,
# "market", which represents the risk market or risk band
# assigned to the borrower

data['market'].unique()
```

[8]:
```
array(['C', 'B', 'A', 'E', 'D'], dtype=object)
```

[9]:
```
# let's make a bar plot with the number of borrowers
# within each category

fig = data['market'].value_counts().plot.bar()
fig.set_title('Status of the Loan')
fig.set_ylabel('Number of customers')
```

[9]:
```
Text(0, 0.5, 'Number of customers')
```



Most customers are assigned to markets B and C. A are lower risk custoemrs, and E the highest risk customers. The higher the risk, the more likely the customer is to default, thus the finance companies charge higher interest rates on those loans.

[10]:
```
# finally, let's look at a variable that is numerical,
# but its numbers have no real meaning
```

```
# their values are more "labels" than real numbers

data['customer_id'].head()
```

[10]:
```
0    0
1    1
2    2
3    3
4    4
Name: customer_id, dtype: int64
```

Each id represents one customer. This number is assigned to identify the customer if needed, while maintaining confidentiality and ensuring data protection.

[11]:
```
# The variable has as many different id values as customers,
# in this case 10000,

len(data['customer_id'].unique())
```

[11]:
```
10000
```

That is all for this demonstration. I hope you enjoyed the notebook, and see you in the next one.