Laporan Praktikum Pertemuan 7

Data Science Lanjut

Data Quality With Python For Beginner

Dibuat oleh

Nama	: Muhamad faisal halim
Nim	: 19.240.0163
Kelas	: -
Mata kuliah	: Data science lanjut

Mahasiswa pertukaran mahasiswa.

Universitas muhammadiyah kalimantan timur

~ stmik widya pratama pekalongan

Penting

Mohon maaf pak, pada materi sebelumnya masih terdapat beberapa tugas yang belum saya kumpulkan, yaitu pada module 3,4 dan 6 untuk laporan saya sertakapan pada link berikut link_1 atau https://bit.ly/laporan-tertinggal-halim0163.

Semoga dengan saya mencantum laporan sebelumnya ini bisa mengisi kekosongan tugas laporan saya dan menjadi bahan pertimbangan tambahan untuk hal yang diperlukan dalam penilaian akhir matakuliah Data Science lanjut

Materi dan praktikum

Title : Data quality with python for beginner

Data profiling

Data profiling adalah kegiatan merangkum dataset menggunakan statistik deskriptif. Yang bertujuan memiliki pemahaman yang kuat tentang data sehingga dapan menyusun framework analist dan memvisualisasikannya.

Dalam praktikum kali ini kita masih menggunakan pandas dan numpy, karena memang dari awal sudah dikatakan bahwa kedua package python tersebut sangay membantu dalam data science.

Importing data

Tahap awal pada praktikum ini adalah mengimport dataset kedalam python dan library atau package dala project kitan.

```
import pandas as pd
import numpy as np
import io
import pandas_profiling

data_raw = pd.read_csv('https://dqlab-dataset.s3-ap-southeast-1.amazonaws.com/retail_raw_reduced_data_quality.csv')
print(data_raw)
```

```
      order_id  order_date  customer_id            city     province  \
0      1703458  17/10/2019        14004  Jakarta Selatan  DKI Jakarta
1      1706815  24/10/2019        17220  Jakarta Selatan  DKI Jakarta
2      1710718  03/11/2019        16518    Jakarta Utara  DKI Jakarta
3      1683592  19/08/2019        16364    Jakarta Barat  DKI Jakarta
4      1702573  16/10/2019        15696    Jakarta Timur  DKI Jakarta
...        ...         ...          ...              ...          ...
4995   1724011  01/12/2019        12838        Tangerang       Banten
4996   1676302  28/07/2019        13833            Bogor   Jawa Barat
4997   1706071  23/10/2019        16332    Jakarta Timur  DKI Jakarta
4998   1703620  17/10/2019        13055    Jakarta Barat  DKI Jakarta
4999   1720036  24/11/2019        17609    Jakarta Pusat  DKI Jakarta
```

Didalam python khususnya menggunakan pandas kita dapat melakuan pengecekan datatype pada setiap kolom.

```
print(data_raw.dtypes)
```

```
order_id         int64
order_date      object
customer_id      int64
city            object
province        object
product_id      object
brand           object
quantity       float64
item_price     float64
dtype: object
```

Deskriptif statistik

Pada tahap ini kita akan mempelajari statistika deskriptif yang digunakan untuk dapat memberikan pemahaman lebih mengenai struktur data.

Lenght

Fungsi len digunakan untuk menghitung jumlah pengamatan dalam satu series atau column.

```
len_city = len(data_raw['city'])
print('Length kolom city:', len_city)

len_pid = len(data_raw['product_id'])
print('Length kolom product_id:',len_pid)
```

```
Length kolom city: 5000
Length kolom product_id: 5000
```

Count

Fungsi ini akan menghitung jumlah pengamatan dalam satu series atau column yang memiliki nilai. Nilai disini bisa diartikan sebagai non null dan bukan missing value.

```
count_city = data_raw['city'].count()
print('Count kolom count_city:', count_city)

count_pid = data_raw['product_id'].count()
print('Count kolom product_id:', count_pid)
```

```
Count kolom count_city: 4984
Count kolom product_id: 4989
```

Missing value

Pada perhitungan ini akan menghasilkan perbedaan antara lenght dan count.

```
number_mv_city = len_city - count_city
flovat_mv_city = float(number_mv_city/len_city)
pct_mv_city = '{0:.1f}%'.format(flovat_mv_city * 100)

print('missing value kolom city:', pct_mv_city)


number_mv_pid = len_pid - count_pid
float_mv_pid = float(number_mv_pid/len_pid)
pct_mv_pid = '{0:.1f}%'.format(float_mv_pid * 100)

print('Persentase missing value kolom product_id:', pct_mv_pid)
```

```
missing value kolom city: 0.3%
Persentase missing value kolom product_id: 0.2%
```

Maximum dan minimum

Fungsi max dan min digunakan untuk mengetahui element terbesar dan terkecil dari suatu kolom di dataframe.

Mean, medium, modus dan standard deviasi

Fungsi mean, medium, modus dan standard deviasi digunakan untuk mengetahui pemusatan data dan persebarannya.

```
print('Kolom quantity')
print('Minimum value: ', data_raw['quantity'].min())
print('Maximum value: ', data_raw['quantity'].max())
print('Mean value: ', data_raw['quantity'].mean())
print('Mode value: ', data_raw['quantity'].mode())
print('Median value: ', data_raw['quantity'].median())
print('Standard Deviation value: ', data_raw['quantity'].std())
```

```
Kolom quantity
Minimum value:  1.0
Maximum value:  720.0
Mean value:  11.423987164059366
Mode value:  0    1.0
dtype: float64
Median value:  5.0
Standard Deviation value:  29.44202501081146
```

Quantile statistics

Quantiles adalah titik potong yang membagi distribusi dalam ukuran yang sama. Jika akan membagi distribusi menjadi empat grup yang sama, kuantil yang dibuat dinamai quartile. Jika dibagi kedalam 10 sepuluh group yang sama dinamakan percentile. Dalam kasus di bawah ini, ingin membagi distribusi menjadi empat grup atau quartile.

```
print('Kolom quantity:')
print(data_raw['quantity'].quantile([0.25, 0.5, 0.75]))

print()

print('Kolom item_price:')
print(data_raw['item_price'].quantile([0.25, 0.5, 0.75]))
```

```
Kolom quantity:
0.25     2.0
0.50     5.0
0.75    12.0
Name: quantity, dtype: float64

Kolom item_price:
0.25     450000.0
0.50     604000.0
0.75    1045000.0
Name: item_price, dtype: float64
```

Correlation

Korelasi adalah cara yang tepat untuk menemukan hubungan antara variabel numerik. Koefisien korelasi berkisar antara -1 hingga 1. Korelasi 1 adalah korelasi positif total, korelasi -1 adalah korelasi negatif total dan korelasi 0 adalah korelasi non-linear.

```
print('Korelasi quantity dengan item_price')
print(data_raw[['quantity', 'item_price']].corr())
```

```
Korelasi quantity dengan item_price
            quantity  item_price
quantity    1.000000   -0.133936
item_price -0.133936    1.000000
```

Penggunaan profilling library

Pada praktik sebelumnya kita tahu bahwa engumpulkan statistik deskriptif dapat menjaro proses yang panjang, pandas profiling library dapat mempersingkat proses tersebut secara otomatis.

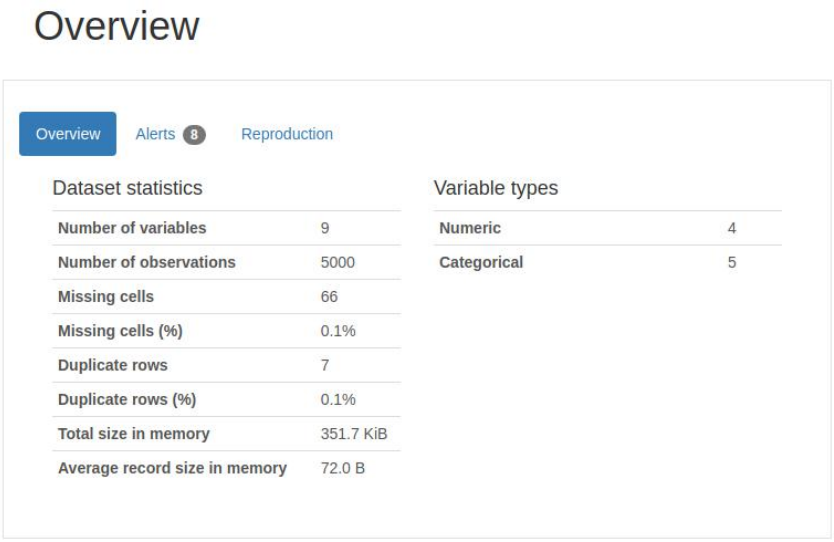
Pertama kita install dulu library yang kita butuhskan dengan command berikut.

```
Pip install pandas_profiling atau
Pip3 install pandas_profiling
```

Pastikan bahwa yang kita install adalah versi terbarunya. Untuk penggunaan kodenya bisa kita lihat seperti berikut. Data_raw diambil dari dataset yang kita import sebelumnya.

```
import pandas_profiling
from pandas_profiling import ProfileReport

pandas_profiling.ProfileReport(data_raw)
```

Untuk hasil run dari kode diatas terlihat cuplikanya seperti berikut.



Data cleansing

Adalah proses mendeteksi dan memperbaiki catatan yang rusak atau tidak akurat dari kumpulan catatan, tabel, atau basis data dan mengacu pada pengidentifikasian bagian data yang tidak lengkap, tidak benar, tidak akurat, atau tidak relevan dan kemudian mengganti, memodifikasi, atau menghapus datanya.

Missing data

Pada sekarang ini dengan banyaknya data yang ditemukan di dunia pastinya terdapay banyak juga missing value dari data tersebut. Oleh karena itu treatment missing value sangatlah penting, karena missing value dapat mempengaruhi analisis dan mechine learning model.

Ada beberapa cara untuk mengatasi ini,
1. Dibiarkan
2. Imputasi
3. Menghapus row yang mengandung missing value

Imputasi adalah cara yang dilakukan untuk mengisi kekosongan data menggunakan teknik tertentu, umumnya menggunakan mean, modus atau mengguankan predictiv modeling.

Mengecek kolom yan memiliki missing value

```
print(data_raw.isnull().any())
```

```
order_id       False
order_date     False
customer_id    False
city            True
province        True
product_id      True
brand          False
quantity        True
item_price      True
dtype: bool
```

Mengisi missing value pada kolom quantity dengan mean

```
print(data_raw['quantity'].fillna(data_raw['quantity'].mean()))
```

```
0       10.0
1        2.0
2        8.0
3        4.0
4        2.0
        ...
4995     2.0
4996     3.0
4997     4.0
4998     8.0
4999     1.0
Name: quantity, Length: 5000, dtype: float64
```

Menghapus missing value pada kolom quantity

```
print(data_raw['quantity'].dropna())
```

```
0       10.0
1        2.0
2        8.0
3        4.0
4        2.0
        ...
4995     2.0
4996     3.0
4997     4.0
4998     8.0
4999     1.0
Name: quantity, Length: 4986, dtype: float64
```

Outlier

Adalah observasi yang muncul dengan nilai yang sama sekali berbeda dengan sebagian besar nilai lain dalam kelompoknya, biasanya disebut juga dengan nilai ekstrim.

Cara treatment outlier antara lain
1. Dihapus
2. Imputasi
3. Capping
4. Prediction

Umumnya outlier dapat ditentukan dengan metric iqr (interquartile range) rumusnya q3 - q1, data suatu observasi dapat dikatakan outlier jika memenuhi syarat-syarat berikut.

< q1 - 1.5 * iqr
> q3 + 1.5 * iqr

```
# Q1, Q3, dan IQR
Q1 = data_raw['quantity'].quantile(0.25)
Q3 = data_raw['quantity'].quantile(0.75)
IQR = Q3 - Q1

# Check ukuran (baris dan kolom) sebelum data yang outliers dibuang
print('Shape awal: ', data_raw.shape)

# Removing outliers
data_raw = data_raw[~((data_raw['quantity'] < (Q1 - 1.5 * IQR)) | (data_raw['quantity'] > (Q3 + 1.5 * IQR)))]

# Check ukuran (baris dan kolom) setelah data yang outliers dibuang
print('Shape akhir: ', data_raw.shape)
```

```
Shape awal:  (5000, 9)
Shape akhir:  (4699, 9)
```

Deduplikasi data

Merupakan data dengan kondisi pada row-row tertentu memiliki kesamaan data diseluruh kolomnya.

```
# cek sebelum data di duplikasi
print('Shape awal: ', data_raw.shape)

# hapus data yang terduplikasi
data_raw.drop_duplicates(inplace=True)

# cek setelah data diduplikasi
print('Shape akhir: ', data_raw.shape)
```

```
Shape awal:  (4692, 9)
Shape akhir:  (4692, 9)
```

Case studi

Profiling data

1. Import data kedalam variable uncleaned_csv
2. Inspeksi data
3. Cek kolom yang mengandung missing value, jika ada berapa persen missing value tersebut.
4. Mengisi missing value dengan mean

Hasil pengerjaan

```
import pandas as pd
import numpy as np
import io
import pandas_profiling

# membaca dataset uncleaned_raw.csv
uncleaned_raw = pd.read_csv('https://dqlab-dataset.s3-ap-southeast-1.amazonaws.com/uncleaned_raw.csv')

#inspeksi dataframe uncleaned_raw
print('Lima teratas:')
print(uncleaned_raw.head())

# cek kolom yang ada missing value
print('\nKolom dengan missing value:')
print(uncleaned_raw.isnull().any())

# persentase missing value
len_qty = len(uncleaned_raw['Quantity'])
count_qty = uncleaned_raw['Quantity'].count()
mv_qty = len_qty - count_qty
float_mv_qty = float(mv_qty / len_qty)

# persentase missing value
print("\nPersentase Missing Value Kolom Quantity : ", '{0:.1f}%'.format(float_mv_qty*100))

# mengisinya dengan mean
uncleaned_raw['Quantity'] = uncleaned_raw['Quantity'].fillna(uncleaned_raw['Quantity'].mean())
print('\nCek Kolom dengan missing value Setelah diisi dengan mean:')
print(uncleaned_raw.isnull().any())
```

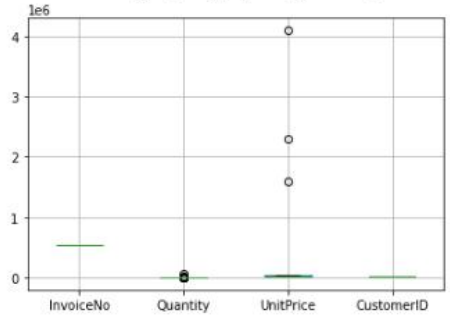
Hasil output

```
   InvoiceNo                          Description  Quantity     InvoiceDate  \
0     536365   WHITE HANGING HEART T-LIGHT HOLDER       6.0  12/01/10 08.26
1     536366                  WHITE METAL LANTERN       6.0  12/01/10 08.26
2     536367       CREAM CUPID HEARTS COAT HANGER       8.0  12/01/10 08.26
3     536368  KNITTED UNION FLAG HOT WATER BOTTLE       6.0  12/01/10 08.26
4     536369       RED WOOLLY HOTTIE WHITE HEART.       6.0  12/01/10 08.26

   UnitPrice  CustomerID      City
0      29000       17850  Surabaya
1      41000       17850  Surabaya
2      18000       17850  Surabaya
3      38000       17850   Jakarta
4      27000       17850     Medan

Kolom dengan missing value:
InvoiceNo      False
Description    False
Quantity        True
InvoiceDate    False
UnitPrice      False
CustomerID     False
City           False
dtype: bool

Persentase Missing Value Kolom Quantity :  4.0%

Cek Kolom dengan missing value Setelah diisi dengan mean:
InvoiceNo      False
Description    False
Quantity       False
InvoiceDate    False
UnitPrice      False
CustomerID     False
City           False
dtype: bool
```

5. Mengetahui kolom yang memiliki outlier, gunakan visualisasi dengan boxplot pada dataframe

```
import pandas as pd
import matplotlib.pyplot as plt

uncleaned_raw = pd.read_csv('https://dqlab-dataset.s3-ap-southeast-1.amazonaws.com/uncleaned_raw.csv')
uncleaned_raw.boxplot()
plt.show()
```

```
/usr/local/lib/python3.7/dist-packages/numpy/core/_asarray.py:83: VisibleDeprecationWarning: Creating an
  return array(a, dtype, copy=False, order=order)
```



6. Melakukan removing outliers pada kolom unitprice
7. Ceking duplikasi dan lakukan deduplikasi dataset tersebut