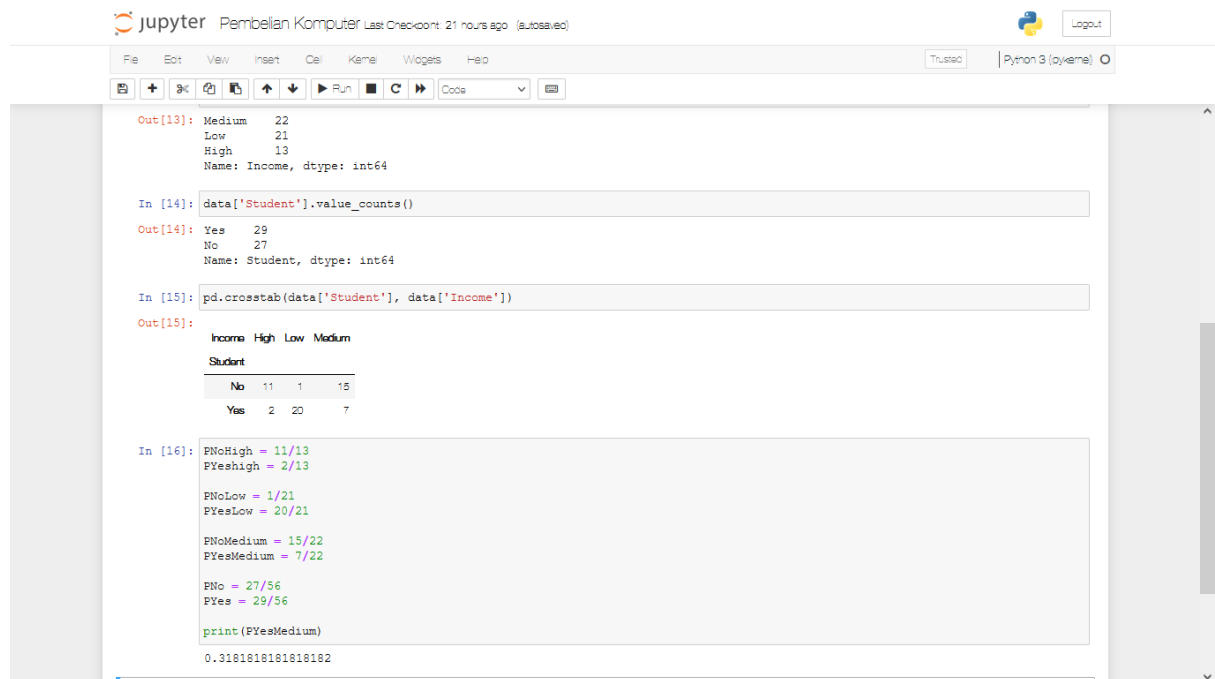


Geulis Juan Ishmah Andini

21.51.0016

Kisi-kisi UAS

1. tingkat pembelian komputer dari Student = Yes, dengan tingkat income = Medium adalah 0.3181818181818182



The screenshot shows a Jupyter Notebook interface with the following code and output:

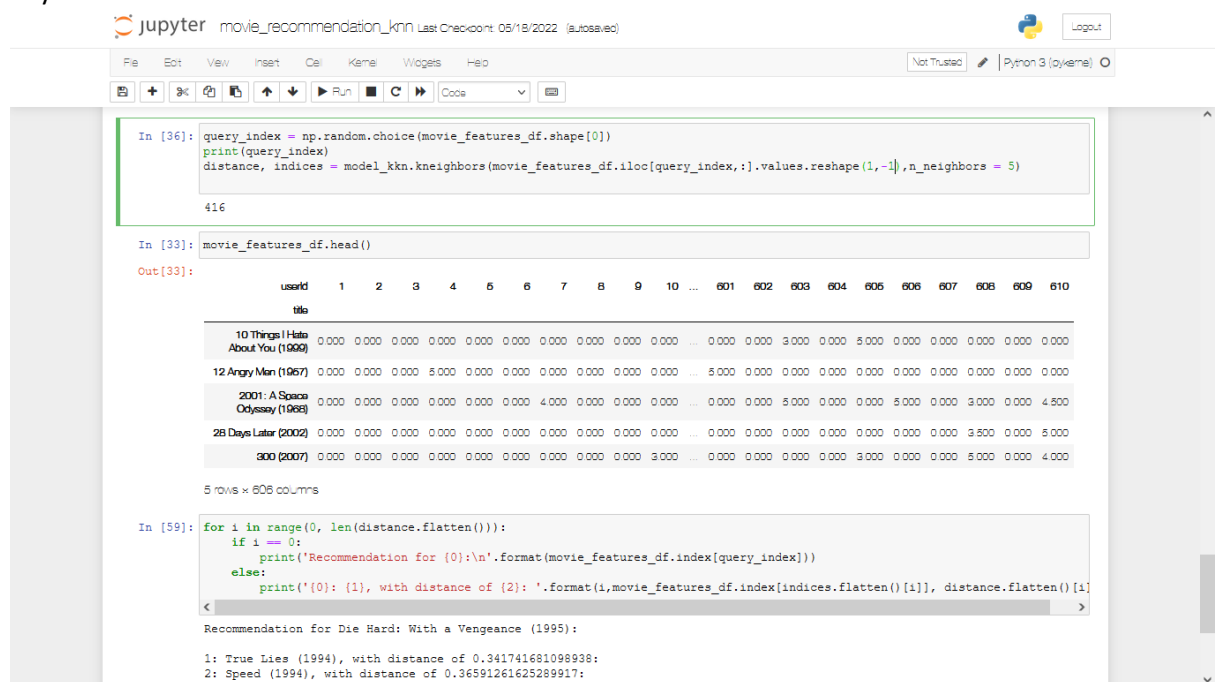
```
Out[13]: Medium    22
Low             21
High            13
Name: Income, dtype: int64

In [14]: data['Student'].value_counts()
Out[14]: Yes      29
No         27
Name: Student, dtype: int64

In [15]: pd.crosstab(data['Student'], data['Income'])
Out[15]:
Income Student
High      No    11     1    15
         Yes     2    20     7
Low       No     9     1    10
         Yes     5    16    21
Medium    No     7     1     8
         Yes     5    19    24

In [16]: PNoHigh = 11/13
PYesHigh = 2/13
PNoLow = 9/21
PYesLow = 12/21
PNoMedium = 7/22
PYesMedium = 17/22
PNo = 27/56
PYes = 29/56
print(PYesMedium)
0.3181818181818182
```

2. key = 5



The screenshot shows a Jupyter Notebook interface with the following code and output:

```
In [36]: query_index = np.random.choice(movie_features_df.shape[0])
print(query_index)
distance, indices = model_knn.kneighbors(movie_features_df.iloc[query_index,:].values.reshape(1,-1), n_neighbors = 5)
416

In [33]: movie_features_df.head()
Out[33]:
   userid  1  2  3  4  5  6  7  8  9  10  ...  601  602  603  604  605  606  607  608  609  610
title
10 Things I Hate About You (1999)  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  ...  0.000  0.000  3.000  0.000  5.000  0.000  0.000  0.000  0.000  0.000
12 Angry Men (1957)                0.000  0.000  5.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  ...  5.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
2001: A Space Odyssey (1968)        0.000  0.000  0.000  0.000  0.000  0.000  4.000  0.000  0.000  0.000  ...  0.000  0.000  5.000  0.000  0.000  5.000  0.000  3.000  0.000  4.500
28 Days Later (2002)                0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  ...  0.000  0.000  0.000  0.000  0.000  0.000  0.000  3.500  0.000  5.000
300 (2007)                          0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  3.000  ...  0.000  0.000  0.000  0.000  3.000  0.000  0.000  5.000  0.000  4.000

5 rows x 608 columns

In [59]: for i in range(0, len(distance.flatten())):
if i == 0:
print('Recommendation for {}:\n'.format(movie_features_df.index[query_index]))
else:
print('{}[0]: {}, with distance of {}: '.format(i, movie_features_df.index[indices.flatten()[i]], distance.flatten()[i]))

Recommendation for Die Hard: With a Vengeance (1995):
1: True Lies (1994), with distance of 0.341741681098938:
2: Speed (1994), with distance of 0.36591261625289917:
```

jupyter movie_recommendation_knn Last Checkpoint: 05/18/2022 (autosaved)

File Edit View Insert Cell Kernel Widgets Help

Not Trusted Python 3 (pykernel)

```

2001: A Space Odyssey (1968) 0.000 0.000 0.000 0.000 0.000 0.000 4.000 0.000 0.000 0.000 ... 0.000 0.000 5.000 0.000 0.000 5.000 0.000 0.000 4.500
28 Days Later (2002) 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 ... 0.000 0.000 0.000 0.000 0.000 3.500 0.000 5.000
300 (2007) 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 3.000 ... 0.000 0.000 0.000 0.000 3.000 0.000 5.000 0.000 4.000
5 rows x 606 columns

```

```

In [59]: for i in range(0, len(distance.flatten())):
          if i == 0:
              print('Recommendation for [0]:\n'.format(movie_features_df.index[query_index]))
          else:
              print('[0]: [1], with distance of [2]: '.format(i, movie_features_df.index[indices.flatten()[i]], distance.flatten()[i]))

```

```

Recommendation for Die Hard: With a Vengeance (1995):
1: True Lies (1994), with distance of 0.341741681098938:
2: Speed (1994), with distance of 0.36591261625289917:
3: Cliffhanger (1993), with distance of 0.3779313564300537:
4: Ace Ventura: Pet Detective (1994), with distance of 0.38145697116851807:
5: GoldenEye (1995), with distance of 0.3842063546180725:
6: Clear and Present Danger (1994), with distance of 0.39991122484207153:
7: Fugitive The (1993), with distance of 0.40829694271087646:
8: Batman (1989), with distance of 0.4132840037345886:
9: Outbreak (1995), with distance of 0.4230532646179199:

```

In []:

3. kelahiran

jupyter Kelahiran Last Checkpoint: 19 hours ago (autosaved)

File Edit View Insert Cell Kernel Widgets Help

Trusted Python 3 (pykernel)

```

In [31]: import pandas as pd
          import numpy as np

In [32]: pd.read_csv("E:/datamining-master/dataset_goal No. 3.csv", usecols=["Usia", "Kelahiran_ke-", "Waktu_Kelahiran", "Tekanan_darah"],
          sep=";", dtype={"Usia": 'int32', 'Kelahiran_ke-': 'int32', 'Waktu_Kelahiran': 'int32', 'Tekanan_darah': 'int32'})

```

```

In [33]: kelahiran_df.head()

```

```

Out[33]:
   Usia  Kelahiran_ke-  Waktu_Kelahiran  Tekanan_darah
0    22             1             0             2
1    26             2             0             1
2    26             2             1             1
3    26             1             0             2
4    22             2             0             1

```

```

In [34]: kelahiran_df.shape

```

```

Out[34]: (80, 4)

```

```

In [48]: pd.set_option('display.float.format', lambda x: '%.3f' % x)
          print(kelahiran_df["Usia"].describe())

```

```

count    80.000
mean     27.688
std       5.018
min      17.000
25%      25.000
50%      27.000
75%      32.000
max      33.000

```

```
In [34]: kelahiran_df.shape
Out[34]: (80, 4)

In [48]: pd.set_option('display.float.format', lambda x: '%.3f' % x)
print(kelahiran_df['Usia'].describe())

count    80.000
mean     27.688
std       5.018
min      17.000
25%      25.000
50%      27.000
75%      32.000
max      40.000
Name: Usia, dtype: float64

In [ ]: popularity_treshold = 30
umur_df = dataset_soa1_with_Usia.query('Usia >= @popularity_treshold')
umur_df.head()

In [ ]: query_index = np.random.choice(kelahiran_df.shape[0])
print(query_index)
distance, indices = model_kkn.kneighbors(kelahiran_df.iloc[query_index,:].values.reshape(1,-1), n_neighbors = 3)

In [ ]:
```