

PROJECT JUPITER
KONSEP DAN APLIKASI DATA MINING



OLEH :
PROFUL BARIYAH
17.51.0004

STMIK PPKIA PRADNYA PARAMITA
MALANG
2020

1. Soal no1

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

```
In [2]: df=pd.read_csv('dataset_soal No.1.csv',delimiter=';')
```

```
In [3]: df.head()
```

```
Out[3]:
```

	Age	Income	Student	Credit_rating	Class (buy_computer)
0	<=30	High	No	Fair	No
1	<=30	High	No	Excellent	No
2	31..40	High	No	Fair	Yes
3	> 40	Medium	No	Fair	Yes
4	> 40	Low	Yes	Fair	Yes

```
In [4]: df.shape
```

```
Out[4]: (51, 5)
```

```
In [5]: #student
df['Student'].value_counts()
```

```
Out[5]: Yes    27
        No     24
```

```
In [6]: PYes = 27/51
PNo = 24/51
```

```
In [7]: print(PYes)
```

```
0.5294117647058824
```

```
In [8]: print(PNo)
```

```
0.47058823529411764
```

```
In [9]: #income with student
pd.crosstab(df['Income'], df['Student'])
```

```
Out[9]:
```

	Student No	Yes
Income		
High	9	2
Low	1	20
Medium	14	5

```
In [10]: PHighNo = 9/24
PLOWNo = 1/24
PMediumNo = 14/24

PHighYes = 2/27
PLOWYes = 20/27
PMediumYes = 5/27

PHigh = 11/51
PLOW = 21/51
PMedium = 19/51

print(PHighNo)
```

```
0.375
```

```
In [11]: print(PLOWNo)
```

```
0.041666666666666664
```

```
In [12]: print(PMediumNo)
```

```
0.5833333333333334
```

```
In [13]: print(PHighYes)
```

```
0.07407407407407407
```

Activate Windows
Go to Settings to activate

Activate Windows

```
In [14]: print(PLowYes)
0.7407407407407407
```

```
In [15]: print(PMediumYes)
0.18518518518518517
```

```
In [16]: print(PHigh)
0.21568627450980393
```

```
In [17]: print(PLow)
0.4117647058823529
```

```
In [18]: print(PMedium)
0.37254901960784315
```

```
In [19]: #credit rating with student
pd.crosstab(df['Credit_rating'], df['Student'])
```

```
Out[19]:
```

	Student	No	Yes
Credit_rating			
Excellent	8	12	
Fair	16	15	

```
In [20]: PExcellentNo = 8/24
PFairNo = 16/24
PExcellentYes = 12/27
PFairYes = 15/27
PExcellent = 20/51
PFair = 31/51
print(PExcellentNo)
0.3333333333333333
```

```
In [21]: print(PFairNo)
0.6666666666666666
```

Activate Windows

```
In [22]: print(PExcellentYes)
0.4444444444444444
```

```
In [23]: print(PFairYes)
0.5555555555555556
```

```
In [24]: print(PExcellent)
0.39215686274509803
```

```
In [25]: print(PFair)
0.6078431372549019
```

```
In [26]: #income with class(buy_computer)
pd.crosstab(df['Income'], df['Class (buy_computer)'])
```

```
Out[26]:
```

	Class (buy_computer)	No	Yes
Income			
High	6	5	
Low	11	10	
Medium	5	14	

Activate Windows
Go to Settings to activate

```
In [27]: PHighNo = 6/22
        PLowNo = 11/22
        PMediumNo = 5/22

        PHighYes = 5/29
        PLowYes = 10/29
        PMediumYes = 24/29

        PHigh = 11/51
        PLow = 21/51
        PMedium = 19/51

        print(PHighNo)
```

0.2727272727272727

```
In [28]: print(PLowNo)

0.5
```

```
In [29]: print(PMediumNo)

0.22727272727272727
```

```
In [30]: print(PHighYes)

0.1724137931034483
```

```
In [31]: print(PLowYes)

0.3448275862068966
```

```
In [32]: print(PMediumYes)

0.8275862068965517
```

```
In [33]: #credit rating with class(buy_computer)
        pd.crosstab(df['Credit_rating'], df['Class (buy_computer)'])
```

```
Out[33]:
```

		Class (buy_computer)	
		No	Yes
Credit_rating	Excellent	8	12
	Fair	14	17

```
In [34]: PExcellentNo = 8/22
        PFairNo = 14/22

        PExcellentYes = 12/29
        PFairYes = 17/29

        PExcellent = 20/51
        PFair = 31/51

        print(PExcellentNo)
```

0.36363636363636365

```
In [35]: print(PFairNo)

0.6363636363636364
```

```
In [36]: print(PExcellentYes)

0.41379310344827586
```

```
In [37]: print(PFairYes)

0.5862068965517241
```

Activate Windows

2. Soal no2

- a. Apabila Cuaca buruk dengan nilai = 1, Weekday, dan Game = 0, maka berapa roti yang harus dibuat?

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

```
In [2]: data=pd.read_excel('E:/KULIAH/SEMESTER 6/data mining/uas/data set2 ke 2/dataset_soal No. 2.xls')
```

```
In [3]: data
```

```
Out[3]:
```

	Category	weatherv-1	holidayv-2	gamev-3	Qty
0	A	5	1	0	250
1	B	3	1	1	200
2	C	1	1	0	75
3	D	4	1	1	400
4	E	4	0	0	150
5	F	2	0	0	50

```
In [4]: import math
dis = []
for i in range(6):
    dis.append(math.sqrt((float(data.iloc[i]['weatherv-1'])-1)**2+
                        (float(data.iloc[i]['holidayv-2'])- 1)**2+
                        (float(data.iloc[i]['gamev-3'])-0)**2))
```

```
In [5]: data['dis'] = dis
data
```

```
Out[5]:
```

	Category	weatherv-1	holidayv-2	gamev-3	Qty	dis
0	A	5	1	0	250	4.000000
1	B	3	1	1	200	2.236068
2	C	1	1	0	75	0.000000
3	D	4	1	1	400	3.162278
4	E	4	0	0	150	3.162278
5	F	2	0	0	50	1.414214

```
In [6]: data.to_excel('E:/KULIAH/SEMESTER 6/data mining/uas/project uas/soal_no2.xls')
```

```
In [7]: data.to_excel('E:/KULIAH/SEMESTER 6/data mining/uas/project uas/soal_no2a.xls')
```

Activate Windows

- b. Apabila Cuaca baik dengan nilai 4, Weekend, dan Game =1, maka berapa roti yang harus dibuat?

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

```
In [2]: data=pd.read_excel('E:/KULIAH/SEMESTER 6/data mining/uas/data set2 ke 2/dataset_soal No. 2.xls')
```

```
In [3]: data
```

```
Out[3]:
```

	Category	weatherv-1	holidayv-2	gamev-3	Qty
0	A	5	1	0	250
1	B	3	1	1	200
2	C	1	1	0	75
3	D	4	1	1	400
4	E	4	0	0	150
5	F	2	0	0	50

```
In [4]: import math
dis = []
for i in range(6):
    dis.append(math.sqrt((float(data.iloc[i]['weatherv-1'])-4)**2+
                        (float(data.iloc[i]['holidayv-2'])- 1)**2+
                        (float(data.iloc[i]['gamev-3'])-1)**2))
```

```
In [5]: data['dis'] = dis
data
```

```
Out[5]:
```

	Category	weatherv-1	holidayv-2	gamev-3	Qty	dis
0	A	5	1	0	250	1.414214
1	B	3	1	1	200	1.000000
2	C	1	1	0	75	3.162278
3	D	4	1	1	400	0.000000
4	E	4	0	0	150	1.414214
5	F	2	0	0	50	2.449490

```
In [6]: data.to_excel('E:/KULIAH/SEMESTER 6/data mining/uas/project uas/soal_no2b.xls')
```

3. Jawaban No 3

```
In [1]: import numpy as np
import pandas as pd
from apyori import apriori
```

```
In [2]: store_data = pd.read_excel('E:/KULIAH/SEMESTER 6/data mining/uas/dataset_soal No. 3.xls')
```

```
In [3]: store_data.head()
```

```
Out[3]:
```

	Item1	Item2	Item3	Item4	Item5	Item6	Item7	Item8	Item9	Item10
0	burgers	meatballs	eggs	low fat yogurt	NaN	mineral water	salmon	low fat yogurt	NaN	mineral water
1	chutney	low fat yogurt	NaN	whole wheat pasta	french fries	mineral water	salmon	whole wheat pasta	french fries	mineral water
2	turkey	whole wheat pasta	french fries	soup	light cream	shallot	NaN	soup	light cream	shallot
3	mineral water	soup	light cream	frozen vegetables	spaghetti	green tea	NaN	frozen vegetables	spaghetti	green tea
4	low fat yogurt	frozen vegetables	spaghetti	french fries	eggs	chocolate	frozen smoothie	french fries	eggs	chocolate

```
In [4]: store_data.tail()
```

```
Out[4]:
```

	Item1	Item2	Item3	Item4	Item5	Item6	Item7	Item8	Item9	Item10
2049	burgers	eggs	french fries	fresh tuna	spaghetti	olive oil	clothes accessories	turkey	eggs	french fries
2050	burgers	eggs	frozen smoothie	french wine	eggs	french fries	energy drink	french fries	NaN	chocolate
2051	whole wheat pasta	cake	melons	champagne	pancakes	light mayo	soup	chocolate	milk	herb & pepper
2052	ground beef	tomato sauce	spaghetti	red wine	honey	hot dogs	turkey	herb & pepper	whole wheat pasta	mineral water
2053	burgers	eggs	frozen smoothie	milk	bacon	eggs	french fries	mineral water	avocado	cookies

```
In [ ]:
```

```
In [5]: store_data.shape
```

```
Out[5]: (2054, 10)
```

```
In [6]: records = []
for i in range(0, 2054):
    records.append([str(store_data.values[i,j]) for j in range(0,10)])
```

```
In [7]: association_rules = apriori(records, min_support=0.2, min_confidence=0.2, min_lift=0.2, min_lenght=2)
association_result = list(association_rules)
```

```
In [8]: print(len(association_result))

61
```

```
In [9]: print(association_result[0])
```

```
RelationRecord(items=frozenset({'avocado'}), support=0.314508276533593, ordered_statistics=[OrderedStatistic(items_base=frozenset(), items_add=frozenset({'avocado'}), confidence=0.314508276533593, lift=1.0)])
```

```
In [13]: result = []
for item in association_result:
    pair = item[0]
    items = [x for x in pair]

    value0 = str(items[0])
    value1 = str(item[1])
    value2 = str(item[1])[:10]
    value3 = str(item[2][0][2])[:10]
    value4 = str(item[2][0][3])[:10]

    rows = (value0,value1,value2,value3,value4)

    result.append(rows)

    label = ['title1', 'title2', 'support','confidence','lift']

    store_suggestion = pd.DataFrame.from_records(result,columns=label)

    print(store_suggestion)
```

```
0 avocado 0.314508276533593 0.31450827 0.31450827 1.0
  title1 title2 support confidence lift
0 avocado 0.314508276533593 0.31450827 0.31450827 1.0
1 burgers 0.24294060370009737 0.24294060 0.24294060 1.0
  title1 title2 support confidence lift
0 avocado 0.314508276533593 0.31450827 0.31450827 1.0
1 burgers 0.24294060370009737 0.24294060 0.24294060 1.0
2 chocolate 0.4756572541382668 0.47565725 0.47565725 1.0
  title1 title2 support confidence lift
0 avocado 0.314508276533593 0.31450827 0.31450827 1.0
```

```

1 burgers 0.24294060370009737 0.24294060 0.24294060 1.0
  title1      title2      support  confidence lift
0 avocado 0.314508276533593 0.31450827 0.31450827 1.0
1 burgers 0.24294060370009737 0.24294060 0.24294060 1.0
2 chocolate 0.4756572541382668 0.47565725 0.47565725 1.0
  title1      title2      support  confidence lift
0 avocado 0.314508276533593 0.31450827 0.31450827 1.0
1 burgers 0.24294060370009737 0.24294060 0.24294060 1.0
2 chocolate 0.4756572541382668 0.47565725 0.47565725 1.0
3 clothes accessories 0.33982473222979553 0.33982473 0.33982473 1.0
  title1      title2      support  confidence lift
0 avocado 0.314508276533593 0.31450827 0.31450827 1.0
1 burgers 0.24294060370009737 0.24294060 0.24294060 1.0
2 chocolate 0.4756572541382668 0.47565725 0.47565725 1.0
3 clothes accessories 0.33982473222979553 0.33982473 0.33982473 1.0
4 cookies 0.3588120740019474 0.35881207 0.35881207 1.0
  title1      title2      support  confidence lift

```

In [14]: store_suggestion.describe()

Out[14]:

	title1	title2	support	confidence	lift
count	61	61	61	61	61
unique	15	53	53	53	1
top	nan	0.24294060370009737	0.24294060	0.24294060	1.0
freq	19	4	4	4	61

In [15]: store_suggestion.to_excel('E:/KULIAH/SEMESTER 6/data mining/uas/project uas/output_soal3.xls')

Activate Windows
Go to Settings to activate

4. Jawaban No 4

- a. Berdasarkan data tersebut bagaimana perlakuan dengan kondisi Ibu hamil dengan Usia 30 Tahun, yang merupakan Kelahiran ke -1, dengan Waktu kelahiran sesuai dengan HPL, Memiliki tekanan darah Normal? Carilah KNN dengan menggunakan Key = 5

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

In [2]: pd.__version__
Out[2]: '1.0.1'

In [3]: data = pd.read_csv('E:/KULIAH/SEMESTER 6/data mining/uas/datamining-master/Uas/dataset_soal No. 4.txt', delimiter=',')

In [4]: data
Out[4]:
```

	Usia	Kelahiran_ke-	Waktu_Kelahiran	Tekanan_darah	Kelainan_jantung	Caesarian
0	22	1	0	2	0	0
1	28	2	0	1	0	1
2	26	2	1	1	0	0
3	28	1	0	2	0	0
4	22	2	0	1	0	1
...
75	27	2	1	1	0	0
76	33	4	0	1	0	1
77	29	2	1	2	0	1
78	25	1	2	0	0	1
79	24	2	2	1	0	0

80 rows x 6 columns

```
In [5]: import math
dis = []
for i in range(80):
    dis.append(math.sqrt((float(data.iloc[i]['Usia'])-30)**2+
        (float(data.iloc[i]['Kelahiran_ke-']- 1)**2+
        (float(data.iloc[i]['Waktu_Kelahiran']-0)**2+
        (float(data.iloc[i]['Tekanan_darah']-1)**2)))

In [6]: data['dis'] = dis
data
Out[6]:
```

	Usia	Kelahiran_ke-	Waktu_Kelahiran	Tekanan_darah	Kelainan_jantung	Caesarian	dis
0	22	1	0	2	0	0	8.062258
1	28	2	0	1	0	1	4.123106
2	26	2	1	1	0	0	4.242641
3	28	1	0	2	0	0	2.236068
4	22	2	0	1	0	1	8.062258
...
75	27	2	1	1	0	0	3.316625
76	33	4	0	1	0	1	4.242641
77	29	2	1	2	0	1	2.000000
78	25	1	2	0	0	1	5.477226
79	24	2	2	1	0	0	6.403124

80 rows x 7 columns

```
In [7]: data.sort_values('dis')
Out[7]:
```

	Usia	Kelahiran_ke-	Waktu_Kelahiran	Tekanan_darah	Kelainan_jantung	Caesarian	dis
27	30	1	0	1	0	0	0.000000
38	31	1	0	1	0	0	1.000000
67	29	2	0	1	1	0	1.414214
54	29	2	0	1	1	1	1.414214
59	30	2	1	2	1	1	1.732051
...
41	19	1	0	1	0	1	11.000000
61	19	1	0	1	0	1	11.000000
25	18	1	0	1	0	0	12.000000
26	18	1	1	2	1	1	12.083046
70	17	1	0	0	0	1	13.038405

80 rows x 7 columns

```
In [8]: y = data.sort_values('dis').head(5)
y
Out[8]:
```

	Usia	Kelahiran_ke-	Waktu_Kelahiran	Tekanan_darah	Kelainan_jantung	Caesarian	dis
27	30	1	0	1	0	0	0.000000
38	31	1	0	1	0	0	1.000000
67	29	2	0	1	1	0	1.414214
54	29	2	0	1	1	1	1.414214
59	30	2	1	2	1	1	1.732051

```
In [9]: z = y["Caesarian"]
z
Out[9]:
```

27	0
38	0
67	0
54	1
59	1

Name: Caesarian, dtype: int64

```
In [10]: np.mean(z)
Out[10]: 0.4
```

```
In [12]: data.to_excel('E:/KULIAH/SEMESTER 6/data mining/uas/project uas/output_soal_no4a.xls')
```

- b. Bagaimana Apabila Ibu hamil dengan Usia 29 Tahun, yang merupakan Kelahiran ke - 2, dengan Waktu kelahiran sesuai dengan HPL, Memiliki tekanan darah Tinggi? Carilah KNN dengan menggunakan Key =5

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

In [2]: pd.__version__
Out[2]: '1.0.1'

In [3]: data = pd.read_csv('E:/KULIAH/SEMESTER 6/data mining/uas/datamining-master/Uas/dataset_soal No. 4.txt', delimiter=',')

In [4]: data
Out[4]:
```

	Usia	Kelahiran_ke-	Waktu_Kelahiran	Tekanan_darah	Kelainan_jantung	Caesarian
0	22	1	0	2	0	0
1	26	2	0	1	0	1
2	26	2	1	1	0	0
3	28	1	0	2	0	0
4	22	2	0	1	0	1
...
75	27	2	1	1	0	0
76	33	4	0	1	0	1
77	29	2	1	2	0	1
78	25	1	2	0	0	1
79	24	2	2	1	0	0

80 rows x 6 columns

```
In [5]: import math
dis = []
for i in range(80):
    dis.append(math.sqrt((float(data.iloc[i]['Usia'])-29)**2+
                        (float(data.iloc[i]['Kelahiran_ke-'])-2)**2+
                        (float(data.iloc[i]['Waktu_kelahiran'])-0)**2+
                        (float(data.iloc[i]['Tekanan_darah'])-2)**2))
```

```
In [6]: data['dis'] = dis
data
```

```
Out[6]:
```

	Usia	Kelahiran_ke-	Waktu_kelahiran	Tekanan_darah	Kelainan_jantung	Caesarian	dis
0	22	1	0	2	0	0	7.071068
1	26	2	0	1	0	1	3.162278
2	26	2	1	1	0	0	3.316625
3	28	1	0	2	0	0	1.414214
4	22	2	0	1	0	1	7.071068
...
75	27	2	1	1	0	0	2.449490
76	33	4	0	1	0	1	4.582576
77	29	2	1	2	0	1	1.000000
78	25	1	2	0	0	1	5.000000
79	24	2	2	1	0	0	5.477228

80 rows x 7 columns

```
In [7]: data.sort_values('dis')
```

```
Out[7]:
```

	Usia	Kelahiran_ke-	Waktu_kelahiran	Tekanan_darah	Kelainan_jantung	Caesarian	dis
54	29	2	0	1	1	1	1.000000
77	29	2	1	2	0	1	1.000000
67	29	2	0	1	1	0	1.000000
59	30	2	1	2	1	1	1.414214
3	28	1	0	2	0	0	1.414214
...
61	19	1	0	1	0	1	10.099505
26	18	1	1	2	1	1	11.090537
31	40	1	0	1	1	1	11.090537
25	18	1	0	1	0	0	11.090537
70	17	1	0	0	0	1	12.208556

80 rows x 7 columns

```
In [8]: y = data.sort_values('dis').head(5)
y
```

```
Out[8]:
```

	Usia	Kelahiran_ke-	Waktu_kelahiran	Tekanan_darah	Kelainan_jantung	Caesarian	dis
54	29	2	0	1	1	1	1.000000
77	29	2	1	2	0	1	1.000000
67	29	2	0	1	1	0	1.000000
59	30	2	1	2	1	1	1.414214
3	28	1	0	2	0	0	1.414214

```
In [9]: z = y["Caesarian"]
z
```

```
Out[9]: 54    1
77     1
67     0
59     1
3       0
Name: Caesarian, dtype: int64
```

```
In [10]: np.mean(z)
```

```
Out[10]: 0.6
```

```
In [11]: data.to_excel('E:/KULIAH/SEMESTER 6/data mining/uas/project uas/output_soal_no4b.xls')
```

https://github.com/irafulbariyah/project_uas