LAPORAN TUGAS BESAR | MACHINE LEARNING

NAMA : Shindy Trimaria Laxmi
NIM : 1301170092
KELAS : IF4107
YOUTUBE : https://www.youtube.com/channel/UCmx4dyswuPjmEGmAzZm-vcw

SUMMARY

Pada tugas besar ini akan digunakan dataset ‘fifa20.csv’. Pada dataset ‘fifa20.csv’ terdapat 18278 data yang memiliki 104 atribut. Lalu telah dilakukan clustering dan classification terhadap data tersebut. Untuk kedua task tersebut, dipilih 7 atribut yaitu potential, pace, physic, power_stamina, passing, shooting dan defending.

Untuk task clustering dilakukan percobaan dengan algoritma K-Means. Digunakan algoritma K-Means karena implementasinya relatif mudah dan memiliki waktu komputasi yang lumayan cepat jika nilai K nya kecil. Digunakan beberapa kombinasi K dan atribut dalam task clustering. Adapun percobaan yang dilakukan adalah:

- K=2 dengan 2 atribut
- K=2 dengan 3 atribut
- K=3 dengan 2 atribut
- K=3 dengan 3 atribut

Kemudian dilakukan evaluasi dengan mencari silhouette score dari clustering yang dihasilkan. Digunakan silhouette score karena silhoueete score dapat mengukur kualitas dari klasterisasi yang dihasilkan.

Sedangkan untuk classification dilakukan percobaan dengan algoritma SVM. Dipilih algoritma SVM karena terdapat Margin yang dapat membuat pemisahan kelas lebih bagus. Untuk task classification, dilakukan dua percobaan yaitu :

- Classification dengan balanced data
- Classification dengan unbalanced data

Lalu dilakukan evaluasi dengan menggunakan accuracy karena accuracy cocok digunakan apabila semua kelas yang dihasilkan penting (tidak ingin memprioritaskan satu kelas saja).

Pada tugas besar ini juga digunakan beberapa teknik eksplorasi dan persiapan data. Untuk eksplorasi data digunakan boxplot untuk melihat pesebaran data dan membantu dalam mendeteksi outlier. Untuk persiapan data, telah dilakukan impute terhadap missing value, binning terhadap atribut potential dan scalling terhadap atribut pace, physic, power_stamina, passing, shooting dan defending.

TAHAPAN PROGRAM

1. Pertama pertama dilakukan read data ‘fifa20.csv’. Pada data tersebut, masih terdapat beberapa missing value, sehingga dilakukan impute terhadap missing value tersebut. Untuk missing value numerik diberikan nilai median dari atribut yang bersangkutan. Sedangkan untuk data selain numerik, diberikan data yang paling banyak muncul pada atribut yang bersangkutan.

	sofifa_id	player_url	short_name	long_name	age	dob	height_cm	weight_kg	nationality	club	...	lwb	ldm	cdm	rc
0	158023	https://sofifa.com/player/158023/lionel-messi/...	L. Messi	Lionel Andrés Messi Cuccittini	32	1987-06-24	170	72	Argentina	FC Barcelona	...	68+2	66+2	66+2	66
1	20801	https://sofifa.com/player/20801/c-ronaldo-dos-...	Cristiano Ronaldo	Cristiano Ronaldo dos Santos Aveiro	34	1985-02-05	187	83	Portugal	Juventus	...	65+3	61+3	61+3	61
2	190871	https://sofifa.com/player/190871/neymar-da-sil...	Neymar Jr	Neymar da Silva Santos Junior	27	1992-02-05	175	68	Brazil	Paris Saint-Germain	...	66+3	61+3	61+3	61
3	200389	https://sofifa.com/player/200389/jan-oblak/20/...	J. Oblak	Jan Oblak	26	1993-01-07	188	87	Slovenia	Atlético Madrid	...	NaN	NaN	NaN	N
4	183277	https://sofifa.com/player/183277/eden-hazard/2...	E. Hazard	Eden Hazard	28	1991-01-07	175	74	Belgium	Real Madrid	...	66+3	63+3	63+3	63

5 rows × 104 columns

Figure 1 Before Impute

	sofifa_id	player_url	short_name	long_name	age	dob	height_cm	weight_kg	nationality	club	...	lwb	ldm	cdm	rc
0	158023	https://sofifa.com/player/158023/lionel-messi/...	L. Messi	Lionel Andrés Messi Cuccittini	32	1987-06-24	170	72	Argentina	FC Barcelona	...	68+2	66+2	66+2	66
1	20801	https://sofifa.com/player/20801/c-ronaldo-dos-...	Cristiano Ronaldo	Cristiano Ronaldo dos Santos Aveiro	34	1985-02-05	187	83	Portugal	Juventus	...	65+3	61+3	61+3	61
2	190871	https://sofifa.com/player/190871/neymar-da-sil...	Neymar Jr	Neymar da Silva Santos Junior	27	1992-02-05	175	68	Brazil	Paris Saint-Germain	...	66+3	61+3	61+3	61
3	200389	https://sofifa.com/player/200389/jan-oblak/20/...	J. Oblak	Jan Oblak	26	1993-01-07	188	87	Slovenia	Atlético Madrid	...	59+2	59+2	59+2	59
4	183277	https://sofifa.com/player/183277/eden-hazard/2...	E. Hazard	Eden Hazard	28	1991-01-07	175	74	Belgium	Real Madrid	...	66+3	63+3	63+3	63

5 rows × 104 columns

Figure 2 After Impute

2. Dilakukan pemilihan atribut yang akan digunakan untuk melakukan task clustering dan task classification. Atribut yang dipilih adalah atribut potential, pace, physic, power_stamina, passing, shooting dan defending. Atribut potential akan digunakan sebagai label untuk task classification.

	potential	pace	physic	power_stamina	passing	shooting	defending
0	94	87.0	66.0	75	92.0	92.0	39.0
1	93	90.0	78.0	85	82.0	93.0	35.0
2	92	91.0	58.0	81	87.0	85.0	32.0
3	93	69.0	66.0	41	58.0	54.0	56.0
4	91	91.0	66.0	84	86.0	83.0	35.0

Figure 3 Choosen Attributes

3. Setelah memilih atribut, dilakukkan binning terhadap atribut potential untuk mengubah data dari numerik menjadi kategori. Digunakan 3 kategori, yaitu high, average dan low.

	potential	pace	physic	power_stamina	passing	shooting	defending
0	high	87.0	66.0	75	92.0	92.0	39.0
1	high	90.0	78.0	85	82.0	93.0	35.0
2	high	91.0	58.0	81	87.0	85.0	32.0
3	high	69.0	66.0	41	58.0	54.0	56.0
4	high	91.0	66.0	84	86.0	83.0	35.0

4. Lalu gunakan boxplot pada atribut numerik untuk melihat persebaran data dan outlier. Namun pada program ini outlier tidak dihilangkan karena pada setiap atribut, nilai dari outlier tersebut masih masuk akal untuk atribut yang bersangkutan. Berikut merupakan hasil dari boxplot pada setiap atribut numerik.

```
count    18278.000000
mean        67.845607
std         10.657660
min         24.000000
25%         63.000000
50%         69.000000
75%         74.000000
max         96.000000
Name: pace, dtype: float64
```

```
count    18278.000000
mean        65.001805
std          9.207275
min         27.000000
25%         60.000000
50%         66.000000
75%         71.000000
max         90.000000
Name: physic, dtype: float64
```

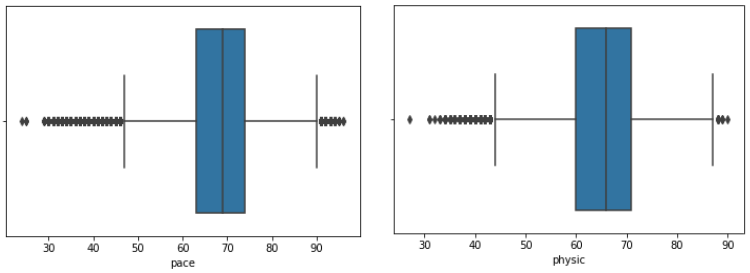

Figure 4 Atribut Pace dan Physic

```
count    18278.000000
mean        62.889485
std         16.017954
min         12.000000
25%         56.000000
50%         66.000000
75%         74.000000
max         97.000000
Name: power_stamina, dtype: float64
```

```
count    18278.000000
mean        57.319127
std          9.813993
min         24.000000
25%         52.000000
50%         58.000000
75%         63.000000
max         92.000000
Name: passing, dtype: float64
```

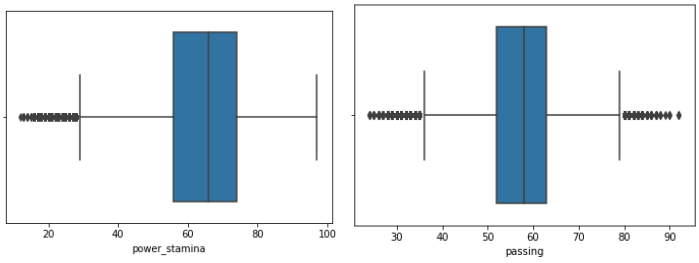

Figure 5 Atribut Power Stamina dan Passing

```
count    18278.000000
mean        52.487854
std         13.235769
min         15.000000
25%         44.000000
50%         54.000000
75%         62.000000
max         93.000000
Name: shooting, dtype: float64
```

```
count    18278.000000
mean        52.048802
std         15.541085
min         15.000000
25%         39.000000
50%         56.000000
75%         64.000000
max         90.000000
Name: defending, dtype: float64
```

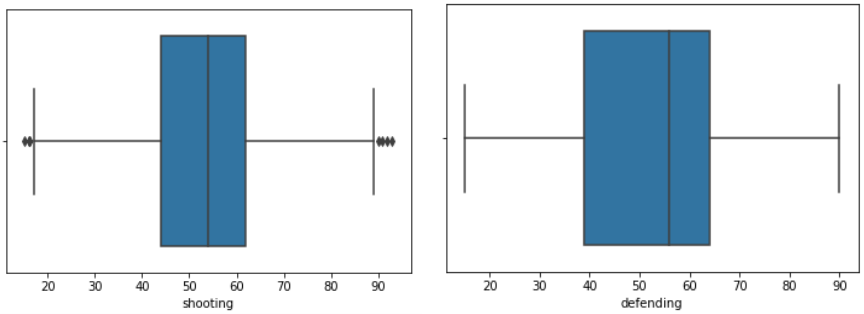

Figure 6 Atribut Shooting dan Stamina

5. Dilakukan scaling terhadap atribut pace, physic, power_stamina, passing, shooting dan defending. Digunakan scaling karena pada task clustering akan digunakan euclidean distance, lalu rentang dari setiap atribut berbeda-beda. Sehingga scaling dilakukan untuk membuat rentang nilai pada setiap atribut menjadi sama. Scaling yang digunakan adalah MinMax Scaling.

	potential	pace	physic	power_stamina	passing	shooting	defending
0	94	0.875000	0.619048	0.741176	1.000000	0.987179	0.320000
1	93	0.916667	0.809524	0.858824	0.852941	1.000000	0.266667
2	92	0.930556	0.492063	0.811765	0.926471	0.897436	0.226667
3	93	0.625000	0.619048	0.341176	0.500000	0.500000	0.546667
4	91	0.930556	0.619048	0.847059	0.911765	0.871795	0.266667

Figure 7 Scaling Data

6. Pada tahap ini dilakukan clustering dengan algoritma K-Means. Lalu dilakukan beberapa percobaan dengan algoritma tersebut, antara lain sebagai berikut.
   a. Percobaan dengan K=2 (membagi menjadi 2 cluster) dan 2 atribut yang berbeda. Atribut yang digunakan adalah Physic-Power Stamina dan Shooting-Defending. Hasil yang didapatkan adalah sebagai berikut.

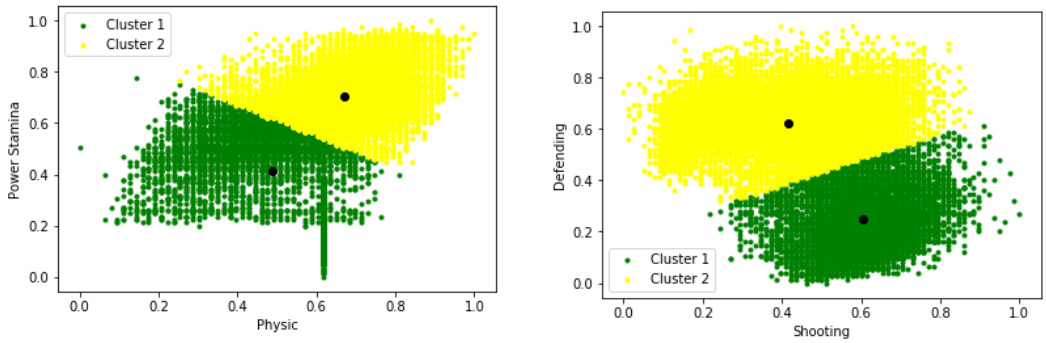

Figure 8 K=2 dan dua attributes. Kiri : Physic-Power Stamina. Kanan : Shooting-Defending.

b. Percobaan dengan K=2 (2 cluster) dan 3 atribut yang berbeda. Atribut yang digunakan adalah Pace-Physic-Power Stamina dan Passing-Shooting-Defending. Hasil yang didapatkan adalah sebagai berikut.

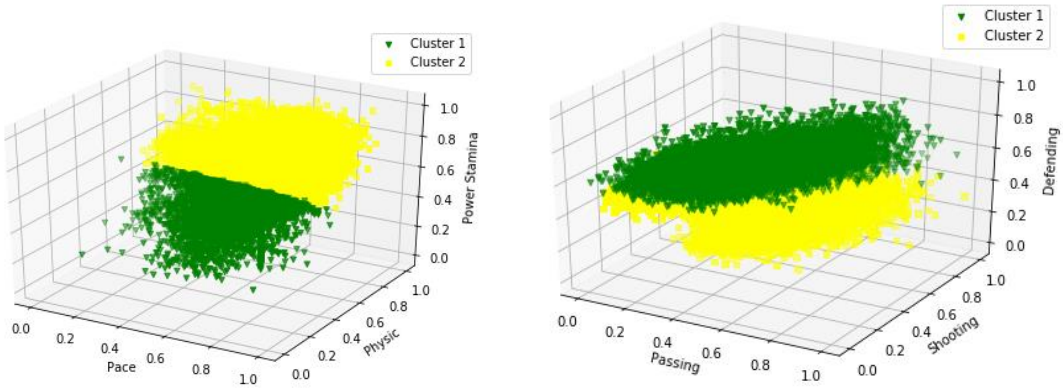

Figure 9 K=2 dan tiga attributes. Kiri : Pace-Physic-Power Stamina. Kanan : Passing-Shooting-Defending

c. Percobaan dengan K=3 (3 cluster) dan 2 atribut yang berbeda. Atribut yang digunakan adalah Physic-Power Stamina dan Shooting-Defending. Hasil yang didapatkan adalah sebagai berikut.

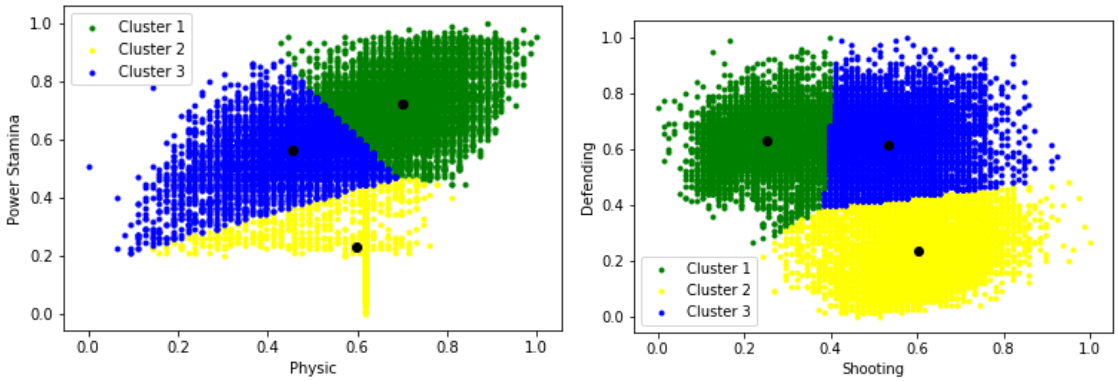

Figure 10 K=3 dan dua attributes. Kiri : Physic-Power Stamina. Kanan : Shooting-Defending.

d. Percobaan dengan K=3 (3 cluster) dan 3 atribut yang berbeda. Atribut yang digunakan adalah Pace-Physic-Power Stamina dan Passing-Shooting-Defending. Hasil yang didapatkan adalah sebagai berikut.

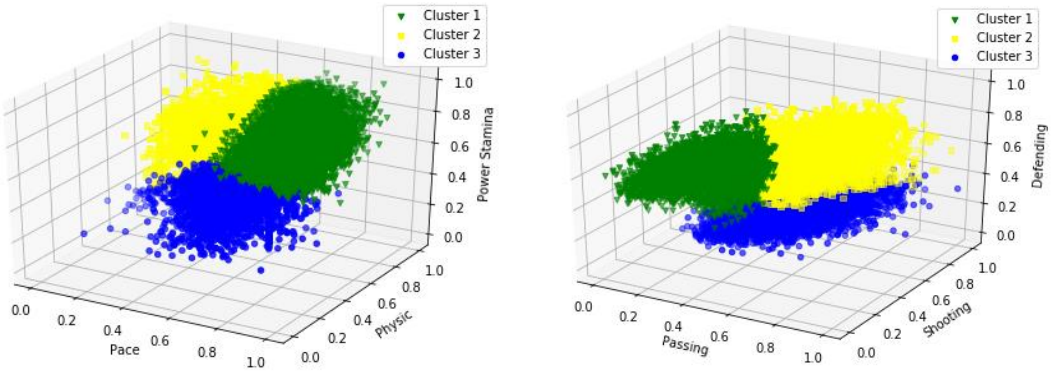

Figure 11 K=3 dan tiga attributes. Kiri : Pace-Physic-Power Stamina. Kanan : Passing-Shooting-Defending

Dari semua percobaan tersebut didapatkan hasil bahwa clustering dengan K=2 dan atribut Shooting-Defending memiliki nilai silhouette score terbaik dengan nilai 0.5087. Untuk perbandingan silhouette score dari semua percobaan dapan dilihat pada gambar berikut.

	K	Attributes	Silhouette Score
0	2	Physic, Power Stamina	0.453454
1	2	Shooting, Defending	0.508723
2	2	Pace, Physic, Power Stamina	0.345351
3	2	Passing, Shooting, Defending	0.380312
4	3	Physic, Power Stamina	0.458189
5	3	Shooting, Defending	0.474132
6	3	Pace, Physic, Power Stamina	0.303864
7	3	Passing, Shooting, Defending	0.423471

Figure 12 Silhouette score

7. Pada tahap ini akan dilakukan task classification dengan algoritma SVM. Pada task ini digunakan 7 atribut ayitu potential, pace, physic, power_stamina, passing, shooting dan defending. Atribut yang digunakan sebagai label adalah atribut potential, sedangkan atribut lainnya digunakan sebagai fitur. Untuk label dilakukan mapping ke bentuk angka, dimana:
   - High = 3
   - Average = 2
   - Low = 1

   Sehingga data yang digunakan akan terlihat sebagai berikut.

	potential	pace	physic	power_stamina	passing	shooting	defending
0	3	0.875000	0.619048	0.741176	1.000000	0.987179	0.320000
1	3	0.916667	0.809524	0.858824	0.852941	1.000000	0.266667
2	3	0.930556	0.492063	0.811765	0.926471	0.897436	0.226667
3	3	0.625000	0.619048	0.341176	0.500000	0.500000	0.546667
4	3	0.930556	0.619048	0.847059	0.911765	0.871795	0.266667

Figure 13 Data classification

Lalu akan dilakukan dua percobaan dengan menggunakan algoritma SVM ini, yaitu:

a. Classification dengan unbalanced data. Dimana data yang memiliki label 2 jauh lebih banyak dari pada data yang memiliki label 1 dan 3. Pada percobaan ini data akan dibagi menjadi dua yaitu 85% data train dan 15% data testing. Setelah membagi data, akan dilakukan classification dengan menggunakan kernel linear.

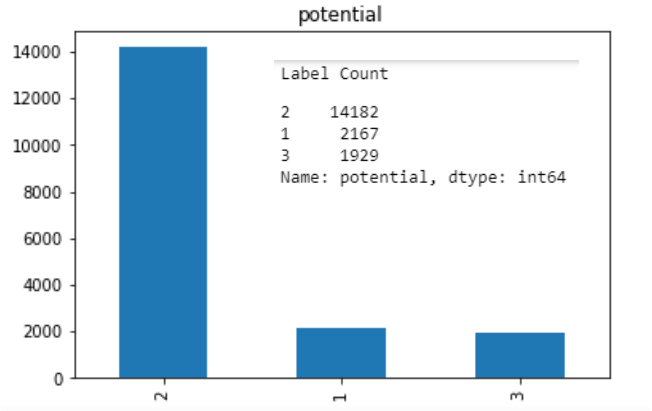

Figure 14 Label Count

```
SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto_deprecated',
  kernel='linear', max_iter=-1, probability=False, random_state=None,
  shrinking=True, tol=0.001, verbose=False)
```

Figure 15 Parameter yang digunakan dalam classification

a. Classification dengan balanced data. Dimana data telah didownsampled sehingga jumlah data pada setiap label sama. Pada percobaan ini data akan dibagi menjadi dua yaitu 85% data train dan 15% data testing. Setelah membagi data, akan dilakukan classification dengan menggunakan kernel linear.

	potential	pace	physic	power_stamina	passing	shooting	defending
12917	1	0.555556	0.587302	0.658824	0.529412	0.512821	0.613333
12144	1	0.500000	0.460317	0.623529	0.367647	0.141026	0.653333
11861	1	0.666667	0.682540	0.764706	0.588235	0.589744	0.466667
16688	1	0.625000	0.619048	0.152941	0.500000	0.500000	0.546667
17569	1	0.555556	0.285714	0.482353	0.411765	0.448718	0.306667

Figure 16 Downsampled data

```
Label Count

3    1929
2    1929
1    1929
Name: potential, dtype: int64
```

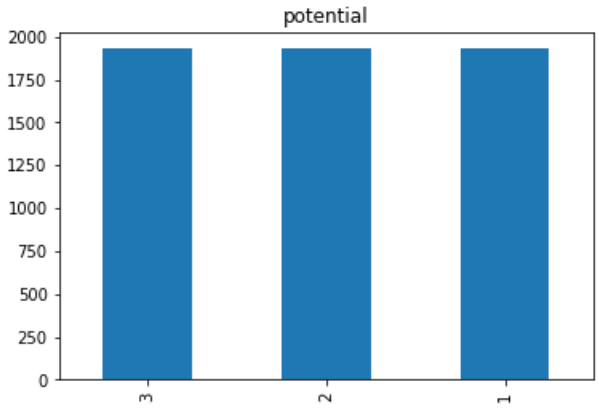

Figure 17 Count Label

```
SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto_deprecated',
  kernel='linear', max_iter=-1, probability=False, random_state=None,
  shrinking=True, tol=0.001, verbose=False)
```

Figure 18 Parameter yang digunakan

Dari dua percobaan diatas, didapatkan hasil akurasi sebagai berikut.

	Model	Kernel	Balanced Data	Training Set Accuracy	Test Set Accuracy
0	Support Vector Machine	Linear	No	0.776905	0.770241
1	Support Vector Machine	Linear	Yes	0.538024	0.524741

Dapat dilihat bahwa classification dengan unbaanced data memiliki nilai yang lebih baik dari classification dengan balanced data. Hal ini disebabkan karena pada balanced data dilakukan downsampling data, yaitu memilih data secara random sebanyak data dengan label paling sedikit yaitu label 3. Tetapi, label 3 hanya berjumlah 1929 data, sedangkan label 2 berjumlah 14182 data. Sehingga banyak data yang terbuang sehingga task classification tidak dapat dilakukan dengan optimal.