

Laporan Tugas Besar Pembelajaran Mesin

1. Data Preparation dan Data Exploration

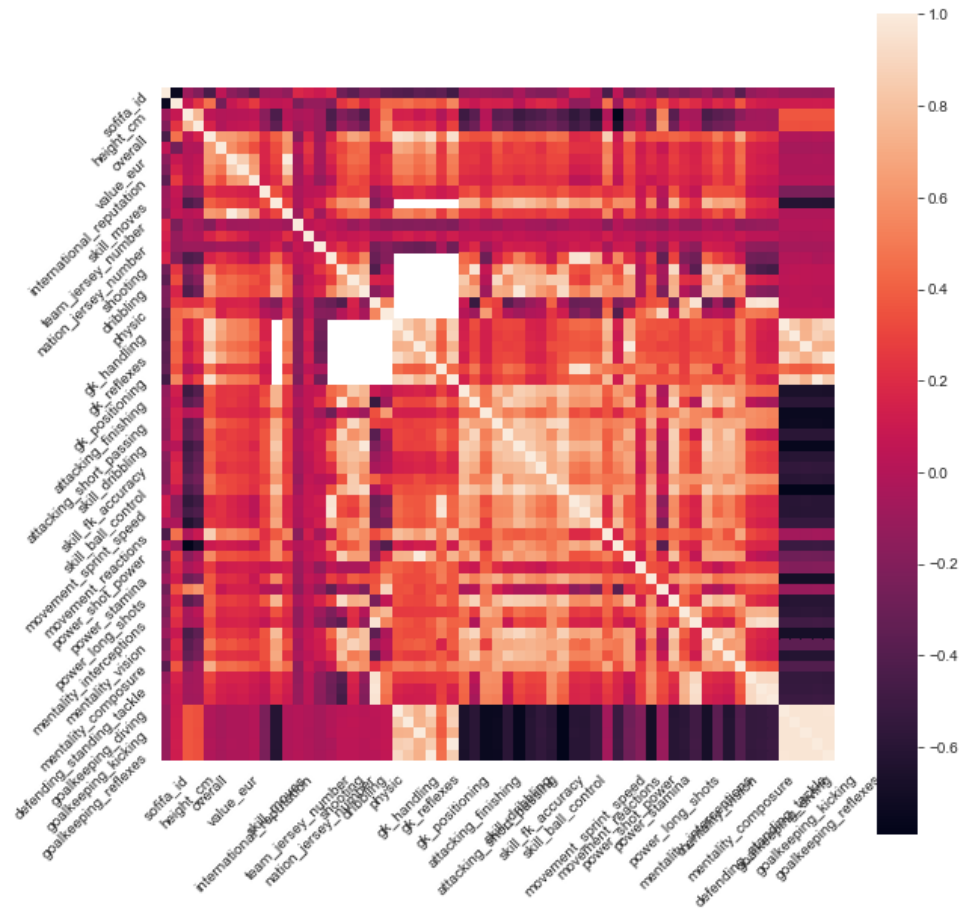
Dalam penyiapan data, saya menampilkan *summary* dari data, untuk melihat apakah ada *missing value* pada data tersebut, yaitu pada gambar berikut

[5]: data.describe()

	sofifa_id	age	height_cm	weight_kg	overall	potential	value_eur	wage_eur	international_reputation	weak_foot	...	mentality_penalties	mentality_composure	defending
count	18278.000000	18278.000000	18278.000000	18278.000000	18278.000000	18278.000000	1.827800e+04	18278.000000	18278.000000	18278.000000	...	18278.000000	18278.000000	18278.000000
mean	219738.864482	25.283291	181.362184	75.276343	66.244994	71.546887	2.484038e+06	9456.942773	1.103184	2.944250	...	48.383357	58.528778	48.383357
std	27960.200461	4.656964	6.756961	7.047744	6.949953	6.139669	5.585481e+06	21351.714095	0.378861	0.664656	...	15.708099	11.880840	15.708099
min	768.000000	16.000000	156.000000	50.000000	48.000000	49.000000	0.000000e+00	0.000000	1.000000	1.000000	...	7.000000	12.000000	7.000000
25%	204445.500000	22.000000	177.000000	70.000000	62.000000	67.000000	3.250000e+05	1000.000000	1.000000	3.000000	...	39.000000	51.000000	39.000000
50%	226165.000000	25.000000	181.000000	75.000000	66.000000	71.000000	7.000000e+05	3000.000000	1.000000	3.000000	...	49.000000	60.000000	49.000000
75%	240795.750000	29.000000	186.000000	80.000000	71.000000	75.000000	2.100000e+06	8000.000000	1.000000	3.000000	...	60.000000	67.000000	60.000000
max	252905.000000	42.000000	205.000000	110.000000	94.000000	95.000000	1.055000e+08	565000.000000	5.000000	5.000000	...	92.000000	96.000000	92.000000

8 rows x 61 columns

Kemudian setelah itu saya melihat korelasi antar data dan melakukan *plotting* untuk melihat korelasi antar data tersebut kemudain menentukan fitur yang saya gunakan untuk melakukan klasifikasi maupun klastering.



Saya melakukan pemilihan fitur berdasarkan plotingan tersebut, akan tetapi saya tidak melakukan *encoding* sehingga untuk klasifikasi saya melakukan *framing problem* secara manual dengan melihat pada data CSV. Kemudian saya mendapatkan ide untuk

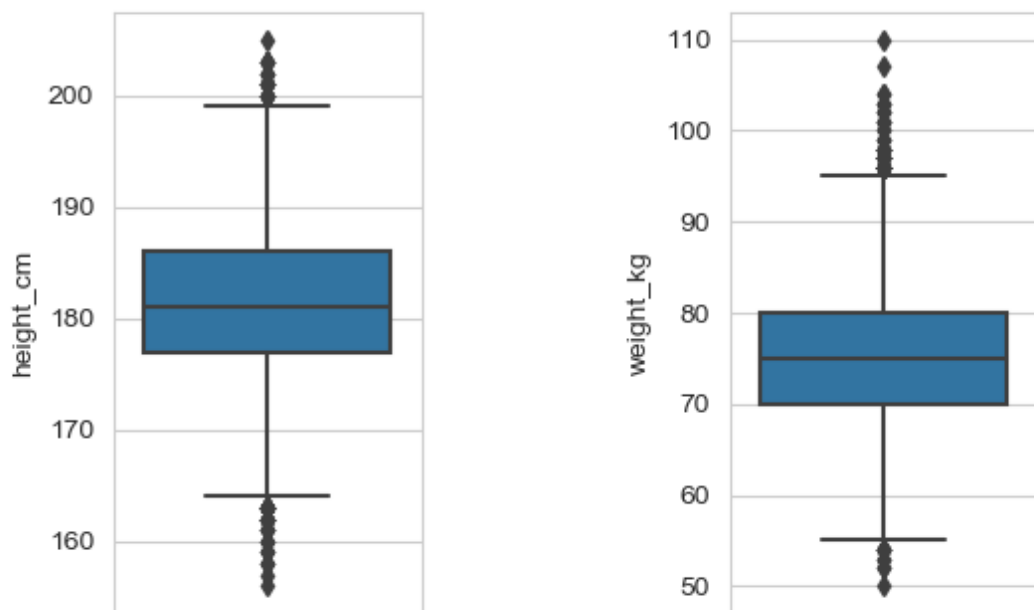
melakukan klasifikasi bentuk tubuh seorang pemain sepakbola. Kemudian baru dengan hasil plot korelasi pada gambar saya menentukan untuk menggunakan tinggi badan dan berat badan, selain karena memang cocok untuk kasusnya, saya juga melihat korelasi antar dua data tersebut lumayan bagus. Sehingga dari situ saya melakukan pemotongan dataset dengan mengambil kolom tinggi badan, berat badan, dan tipe badan untuk dijadikan dataset klasifikasi.

Kemudian untuk kasus klustering, saya murni melihat korelasi data dari hasil plot, kemudian saya memutuskan untuk melakukan klasterisasi pemain berdasar potensial dan *overall* juga melakukan klasterisasi pemain berdasar *overall* dan harga pemain tsb dengan harga eropa. Sehingga untuk klasterisasi saya akan melakukan eksperimen dengan 2 data yang berbeda. Sama dengan klasifikasi saya juga melakukan pemotongan dataset sesuai yang saya butuhkan.

Kemudian data-data yang sudah saya ambil tadi masing-masing saya cek kembali apakah ada *missing value* pada data tersebut dan melakukan pencarian nilai *outliers* pada data tersebut menggunakan boxplot seperti pada gambar berikut

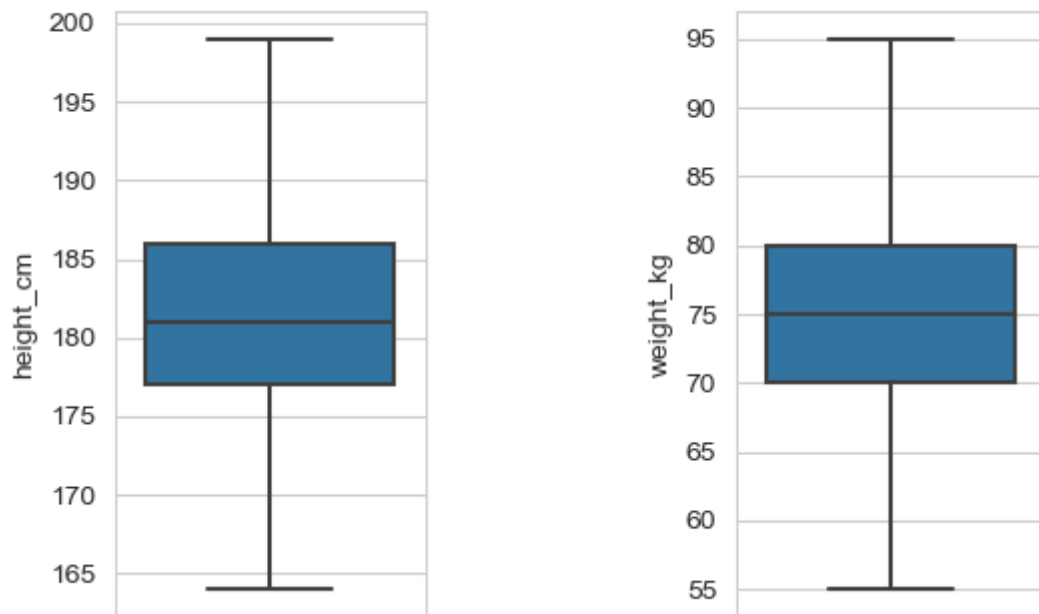
```
[8]: boxPlot(data["height_cm"], data["weight_kg"])
```

<Figure size 6000x6000 with 0 Axes>



Dari hasil tersebut, saya harus melakukan penghilangan data *outliers* tersebut. Dalam percobaan ini, karena masing-masing data outliersnya hampir sama, maka saya melakukan outliers hingga data outliers tersebut benar-benar hilang, hasilnya adalah sebagai berikut

<Figure size 6000x6000 with 0 Axes>



Saya melakukan teknik ini untuk kedua data untuk klastering maupun klasifikasi. Setelah itu, saya melakukan pengkonversian data yang semula masih dalam bentuk dataframe menjadi bentuk data *array*. Untuk kasus klasifikasi, saya melakukan split data menjadi 20% untuk *datatest* dan 80% untuk *datatrain*, sedangkan untuk kasus klastering sendiri saya tidak melakukan *split data*.

2. Klasifikasi

Pada metode klasifikasi, setelah melakukan split data, saya memutuskan untuk melakukan beberapa eksperimen, yaitu dengan menggunakan *SVM*, *SGD*, *Naïve Bayes*, dan *Neural Network*. Dan dari eksperimen tersebut menghasilkan kesimpulan sebagai

Conclusion

From all experiment that we did we got conclusion that using :

1. SVM model give us accuracy 63%
2. Neural Network model give us 62%
3. Naive Bayes model give 60%
4. SGD model give 37 %

So we can conclude that SVM give the best model for this experiment, and SGD model is not suitable for this data and experiment
berikut

Jadi didapatkan kesimpulan bahwa SVM merupakan model terbaik dan SGD adalah model terburuk untuk eksperimen ini. Hal ini didapatkan karena telah dilakukan evaluasi dengan mengecek akurasi dari model yang telah dibangun dengan cara menggunakan *datatest* dan didapatkan akurasi untuk masing-masing model. Saya menggunakan evaluasi berdasarkan akurasi karena ini merupakan teknik yang paling sederhana

dan cocok untuk kasus *supervised* karena sudah ada patokan untuk menguji kebenaran data.

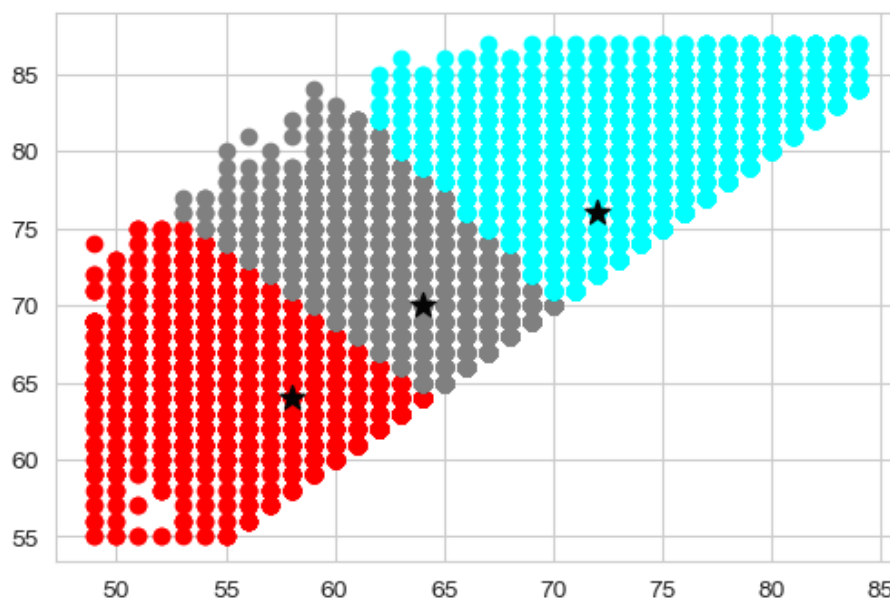
3. Klastering

Pada klastering, saya menggunakan satu metode yaitu *K-Means*. Kemudian saya melakukan dua eksperimen pada metode ini dengan menggunakan 2 fitur berbeda. Yaitu menggunakan nilai *potential* dan *overall*, satunya lagi menggunakan nilai *value europe* dan *overall*. Sehingga jika dilihat dari data yang saya pilih, saya ingin melakukan klasterisasi berdasarkan :

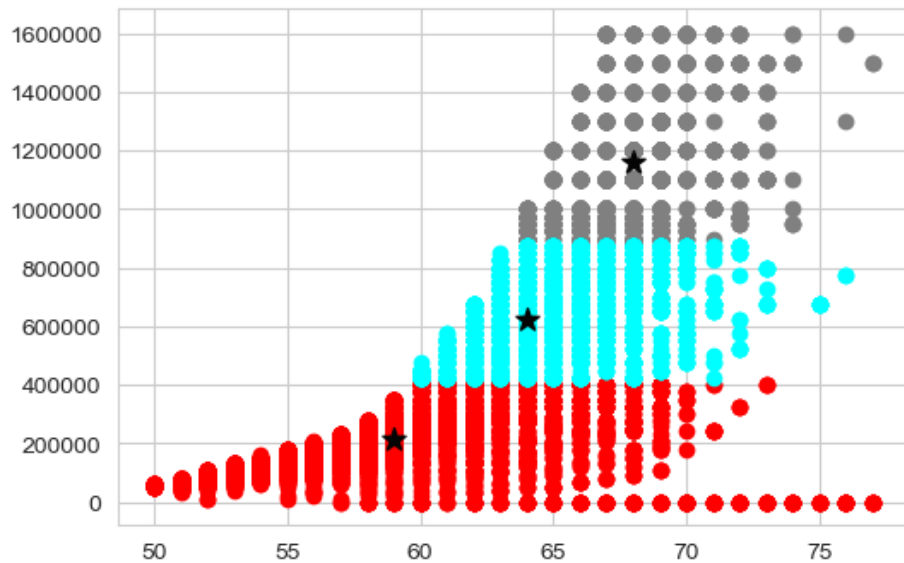
1. Nilai kemampuan mereka dalam sepakbola
2. Nilai dan harga mereka berdasar harga eropa

Jadi masalah yang saya angkat dalam kasus klastering adalah dua hal tersebut, kemudian saya mendapatkan hasil untuk masing data tersebut untuk $k = 3$ seperti berikut

1. Berdasar *Overall* (Y) dan *Potential* (X)



2. Berdasar *Value Europe (Y)* dan *Overall (X)*



Kemudian dalam pengevaluasian hasil klustering tersebut saya menggunakan *Silhouette Coefficient* dimana metode ini menghitung nilai kecocokan suatu data terhadap kluster tersebut berdasarkan nilai *euclidean distance* antara data dengan titik *centroid* nya. Kemudian saya menghitung rata-rata keseluruhan nilai tersebut dan mendapatkan nilai sebagai berikut :

Evaluatin model using Silhouette Coefficient

```
[49]: print('Value of the cluster : ')
print(silhouette_samples(npData, clusters, metric='euclidean')*100)
print()
print('Mean of the cluster value : ')
print(np.mean(silhouette_samples(npData, clusters, metric='euclidean')*100))
```

Value of the cluster :
[40.50278831 40.50278831 41.40902274 ... 41.10068622 39.5698761
39.5698761]

Mean of the cluster value :
36.19506464795887

```
[50]: print('Value of the cluster : ')
print(silhouette_samples(npData2, clusters2, metric='euclidean')*100)
print()
print('Mean of the cluster value : ')
print(np.mean(silhouette_samples(npData2, clusters2, metric='euclidean')*100))
```

Value of the cluster :
[65.56263651 59.99921043 65.56263651 ... 70.51851004 70.51851004
70.51851004]

Mean of the cluster value :
57.83928938383108

Conclusion

From this experiment, we can conclude that **experiment 1** that evaluate using *silhouette coefficient* show how good the cluster value for each data to those centroid is 35 %, while **experiment 2** give value that I think it's big enough that is 57%

1. Untuk eksperimen pertama menghasilkan nilai sekitar 36%, hal ini menunjukkan data tersebut tidak terlalu bagus saat dikelompokkan, dapat dipengaruhi oleh korelasi antar data, algoritma, maupun banyak kluster

optimal untuk data tersebut. Menurut saya pada eksperimen satu dipengaruhi oleh banyak kluster optimal dan algoritma yang digunakan.

2. eksperimen kedua menghasilkan nilai sekitar 57%, hal ini menunjukkan bahwa eksperimen kedua lebih bagus dalam mengelompokkan datanya daripada eksperimen satu. Untuk eksperimen kedua menurut saya, nilai yang didapatkan hanya sebesar ini karena algoritmanya kurang tepat untuk data tersebut.

4. Kesimpulan

Dari semua proses yang dijalankan didapatkan kesimpulan untuk masing-masing tugas, yaitu klasifikasi dan klastering sebagai berikut :

1. Klasifikasi

Dari proses dan eksperimen yang telah dilakukan, telah dibangun sistem yang dapat mengklasifikasikan bentuk tubuh pemain sepakbola berdasar tinggi badan dan berat badan. Akurasi tertinggi yang bisa didapatkan adalah 63% menggunakan metode SVM.

2. Klastering

Menggunakan metode K-Means telah berhasil dibangun untuk melakukan klastering data menggunakan data dua dimensi dan bisa dilihat hasilnya pada hasil plot akhir.

Saran

Untuk kedepannya pada **klasifikasi** disarankan untuk lebih menambah fitur yang digunakan, karena pada eksperiment ini masih menggunakan variasi fitur yang sangat minimum, sehingga diharapkan dengan fitur yang lebih banyak, akan menghasilkan hasil yang lebih baik.

Sedangkan untuk **klastering**, diharapkan kedepannya bisa memilih dan membangun algoritma klastering yang cocok untuk masing-masing data, karena tidak semua data cocok dengan satu metode klastering.