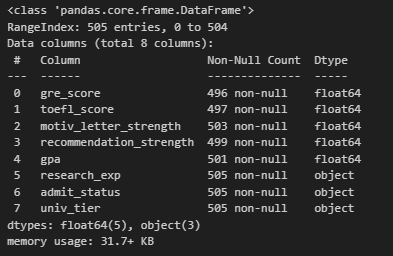


Prediksi admission pakai klasifikasi

Variabelnya ada yang object



Data ada duplikasi, tapi belum tentu mereka orang yang sama



Sehingga tidak perlu dihapus.

Cek missing: sangat kecil, didrop saja

gre\_score 0.017822

toefl\_score 0.015842

motiv\_letter\_strength 0.003960

recommendation\_strength 0.011881

gpa 0.007921

research\_exp 0.000000

admit\_status 0.000000

univ\_tier 0.000000

dtype: float64

data tinggal 476 data

<class 'pandas.core.frame.DataFrame'>

Index: 476 entries, 0 to 504

Data columns (total 8 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 gre\_score 476 non-null float64

1 toefl\_score 476 non-null float64

2 motiv\_letter\_strength 476 non-null float64

3 recommendation\_strength 476 non-null float64

4 gpa 476 non-null float64

5 research\_exp 476 non-null object

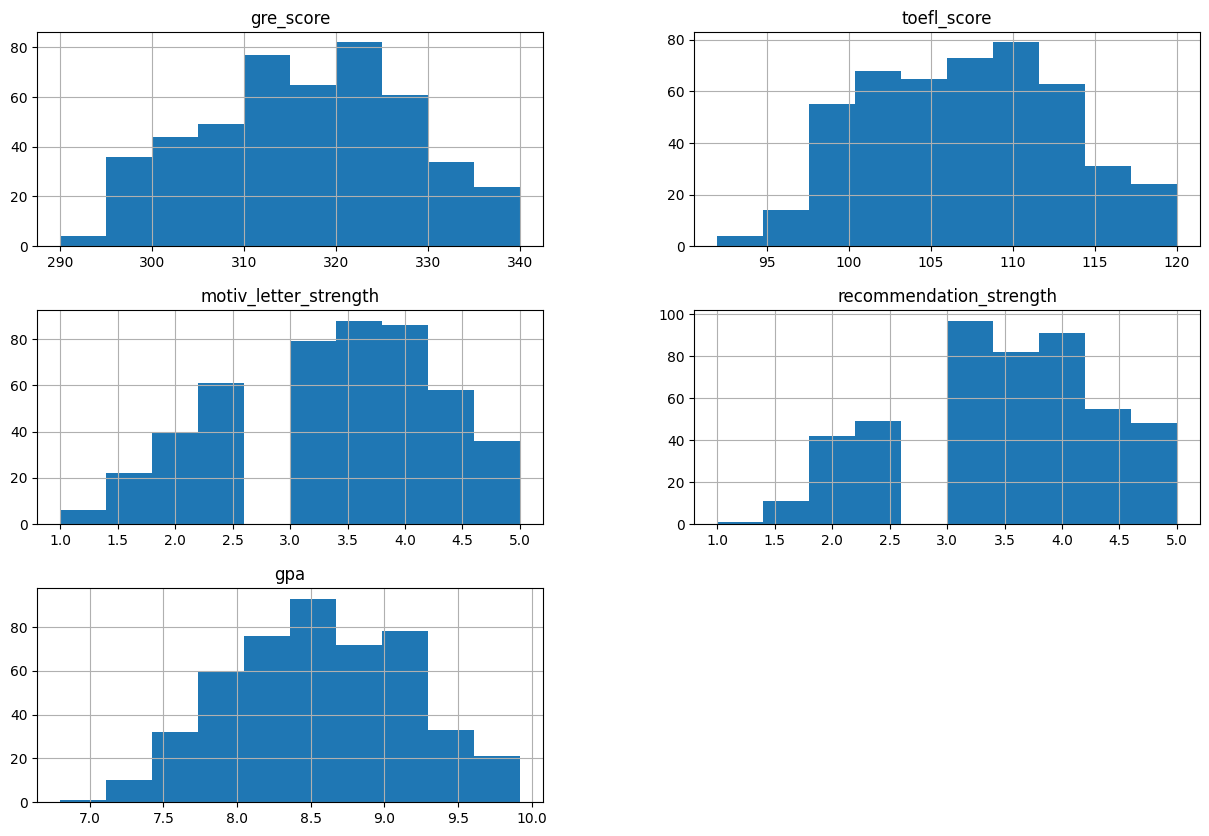
6 admit\_status 476 non-null object

7 univ\_tier 476 non-null object

dtypes: float64(5), object(3)

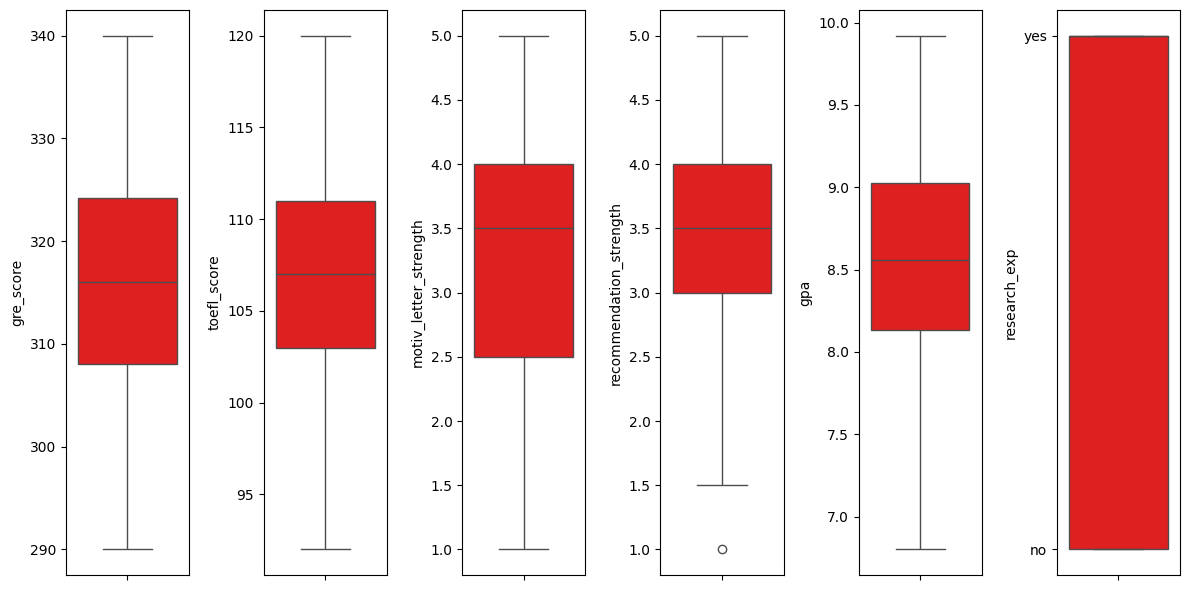
memory usage: 33.5+ KB

data sudah normal



Tidak perlu transformasi

Outlier handling



Cuma satu outlier, dibuang saja

Data tinggal 475

<class 'pandas.core.frame.DataFrame'>

Index: 475 entries, 0 to 504

Data columns (total 8 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 gre\_score 475 non-null float64

1 toefl\_score 475 non-null float64

2 motiv\_letter\_strength 475 non-null float64

3 recommendation\_strength 475 non-null float64

4 gpa 475 non-null float64

5 research\_exp 475 non-null object

6 admit\_status 475 non-null object

7 univ\_tier 475 non-null object

dtypes: float64(5), object(3)

memory usage: 33.4+ KB

Encoding

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Value Counts of research\_exp

research\_exp

yes 260

no 215

Name: count, dtype: int64

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Value Counts of admit\_status

admit\_status

yes 271

no 204

Name: count, dtype: int64

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Value Counts of univ\_tier

univ\_tier

high 304

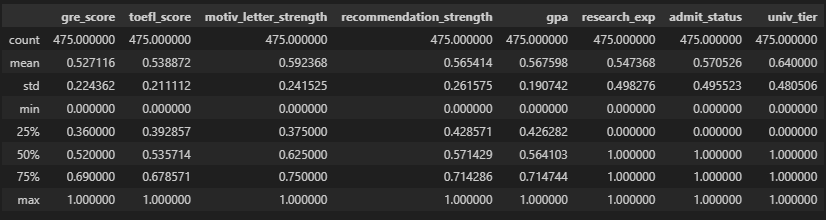
low 171

Name: count, dtype: int64

Setelah encoding menjadi begini



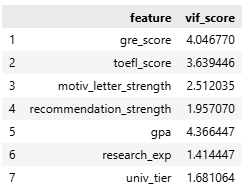
Dilakukan scaling dan hasil akhirnya data yang akan dimodelkan



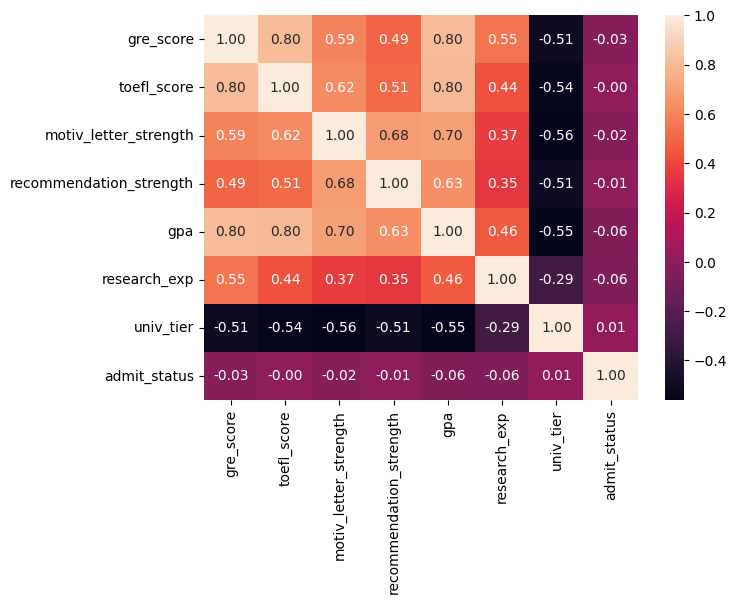
Data yang dimodelkan adalah : 'D:/Bootcamp ML & AI/misi\_mira/data\_clean.csv'

Spliting data into training and testing (80:20)

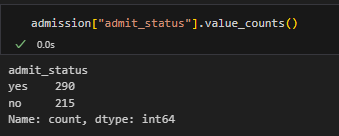
Cek multikolinieritas



Gre dan gpa kemungkinan besar memuat multikolinieritas.



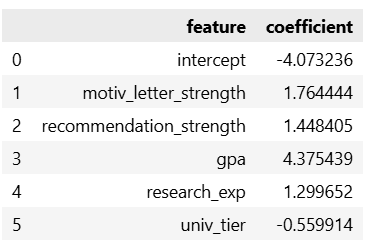
Gre dan toefl didrop. Sehingga fitur yang tersedia tinggal motiv letter, recommend, gpa, reserach exp, dan univ tier.



Seimbang

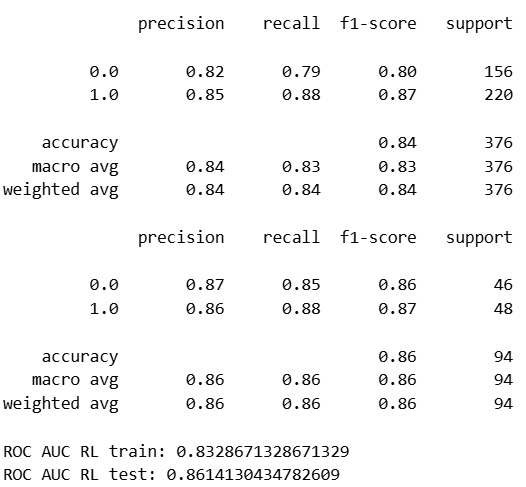
Tidak perlu undersampling atau oversampling.

Hasil regresi logistik

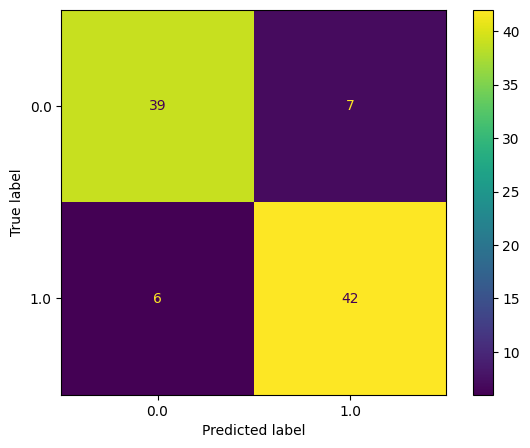


Tujuan disini untuk prediksi, sehingga tidak perlu cek signifikansi.

Presisi akurasi

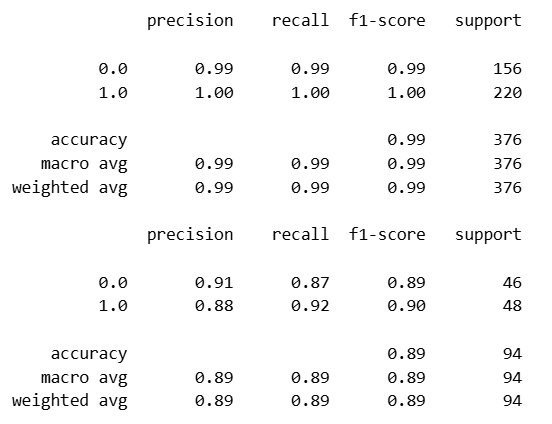


Confusion matrix



Random forest:

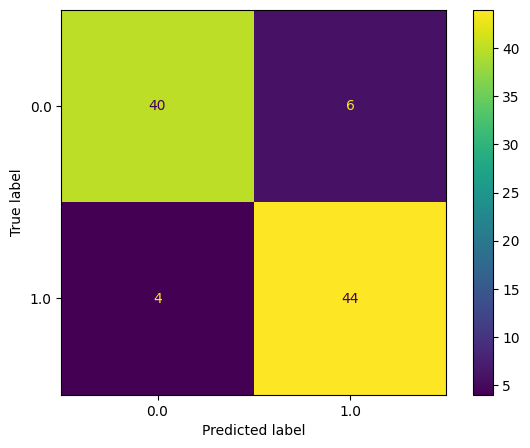
Training



Testingh

ROC AUC RF train: 1.0

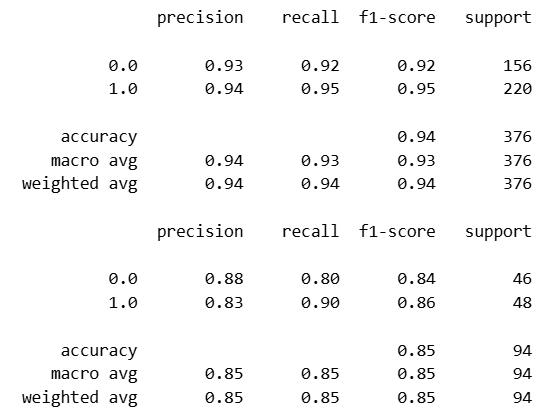
ROC AUC RF test: 0.8718297101449276



Gradient Boosted Trees

Training

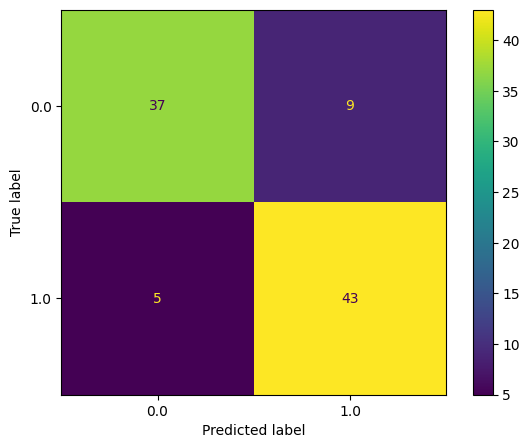
Testing



ROC AUC GBT train: 0.9588578088578089

ROC AUC GBT test: 0.8614130434782609

Confusion matrix



Berdasarkan perbandingna nilai ukuran kebaikan, regresi logistik yang paling baik karena paling stabil (model yang lain overvittiing dan performanya menurun saat testing). Sehingga model yang akan digunakan adalah model regresi logistik.

Model logreg begini yang digunakan

