Hospital Readmission Analysis

Axel Houte, Thomas Hédan, Minji Kim - DIA3

Python For Data Analysis

→

Summary

- Dataset
- Variables
- Visualizations
- Link between readmission and variables

- Decision trees
- Bagging
- Gradient boosting
- Why re-sample ?
- Use again the same models
- Conclusion

Dataset : Diabetes 130 US hospitals for years 1999-2008

Readmitted : Days to inpatient readmission. Values: “<30” if the patient was readmitted in less than 30 days, “>30” if the patient was readmitted in more than 30 days, and “No” for no record of readmission

VARIABLES

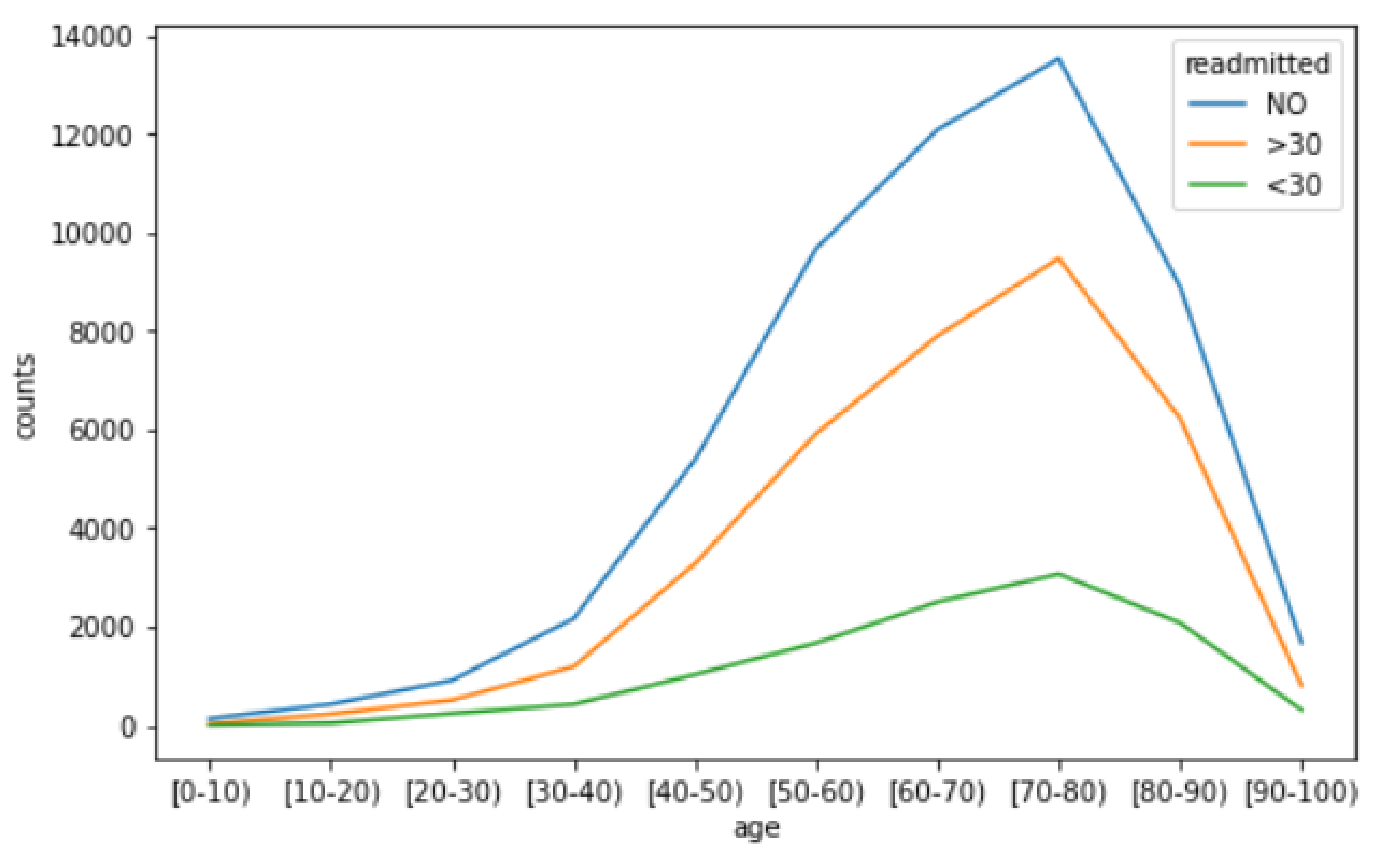
Encounter ID: Unique identifier of an encounter
Patient number: Unique identifier of a patient
Race Values: Caucasian, Asian, African American, Hispanic, and other
Gender Values: male, female, and unknown/invalid
Age Grouped in 10-year intervals: 0, 10), 10, 20), …, 90, 100)
Weight: Weight in pounds
Admission type: Integer identifier corresponding to 9 distinct values, for example, emergency, urgent, elective, newborn, and not available
Discharge disposition: Integer identifier corresponding to 29 distinct values, for example, discharged to home, expired, and not available
Admission source: Integer identifier corresponding to 21 distinct values, for example, physician referral, emergency room, and transfer from a hospital
Time in hospital: Integer number of days between admission and discharge
Payer code: Integer identifier corresponding to 23 distinct values, for example, Blue Cross/Blue Shield, Medicare, and self-pay Medical
Medical specialty: Integer identifier of a specialty of the admitting physician, corresponding to 84 distinct values, for example, cardiology, internal medicine, family/general practice, and surgeon
Number of lab procedures: Number of lab tests performed during the encounter
Number of procedures: Numeric Number of procedures (other than lab tests) performed during the encounter
Number of medications: Number of distinct generic names administered during the encounter

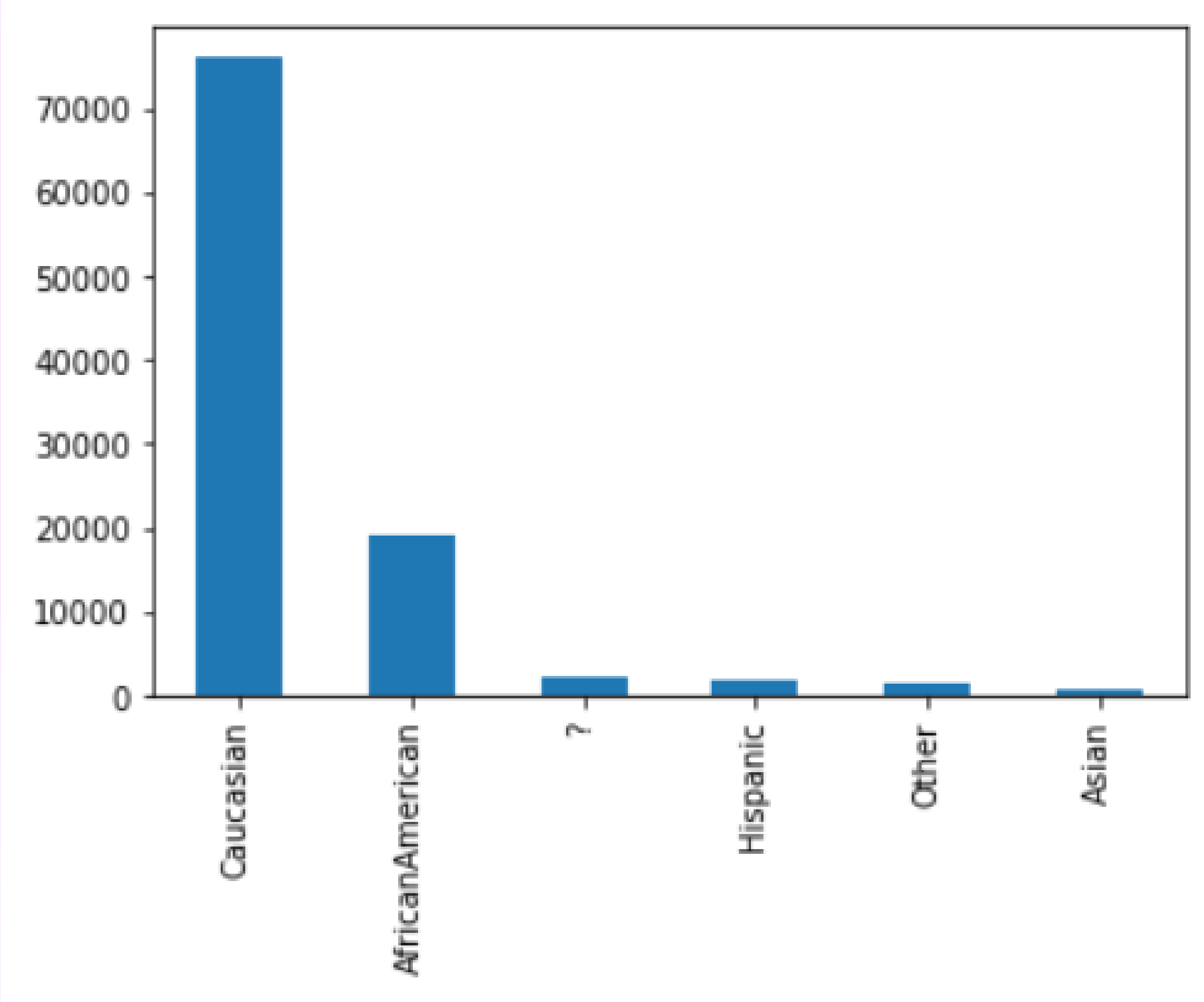
Number of outpatient: visits Number of outpatient visits of the patient in the year preceding the encounter
Number of emergency: visits Number of emergency visits of the patient in the year preceding the encounter
Number of inpatient visits: Number of inpatient visits of the patient in the year preceding the encounter
Number of diagnoses: Number of diagnoses entered to the system 0%
Glucose serum test result Indicates the range of the result or if the test was not taken. Values: “>200,” “>300,” “normal,” and “none” if not measured
A1c test result Indicates the range of the result or if the test was not taken. Values: “>8” if the result was greater than 8%, “>7” if the result was greater than 7% but less than 8%, “normal” if the result was less than 7%, and “none” if not measured.
Change of medications: Indicates if there was a change in diabetic medications (either dosage or generic name). Values: “change” and “no change”
Diabetes medications: Indicates if there was any diabetic medication prescribed. Values: “yes” and “no”
24 features for medications Values: “up” if the dosage was increased during the encounter, “down” if the dosage was decreased, “steady” if the dosage did not change, and “no” if the drug was not prescribed

Created and dropped variables

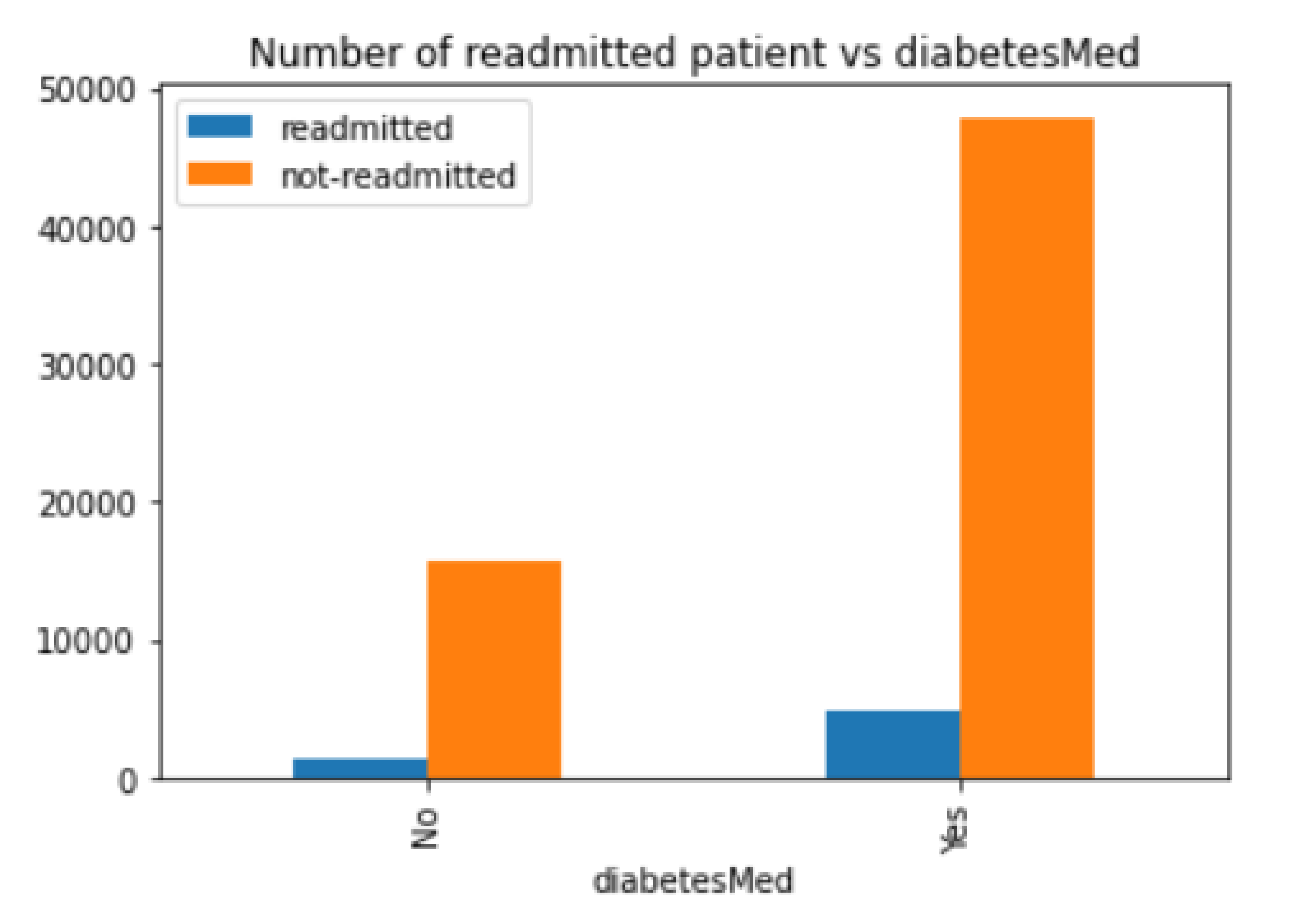
Created	Dropped
• readmitted : yes values in the readmitted column • non-readmitted : no values and >30 values in the readmitted column	• encounter_code • weight, payer_code, medical_specialty (columns with high null values) • diag_1, diag_2, diag_3 • examide, citoglipton (always the same value) • useless_drugs

Visalulizations

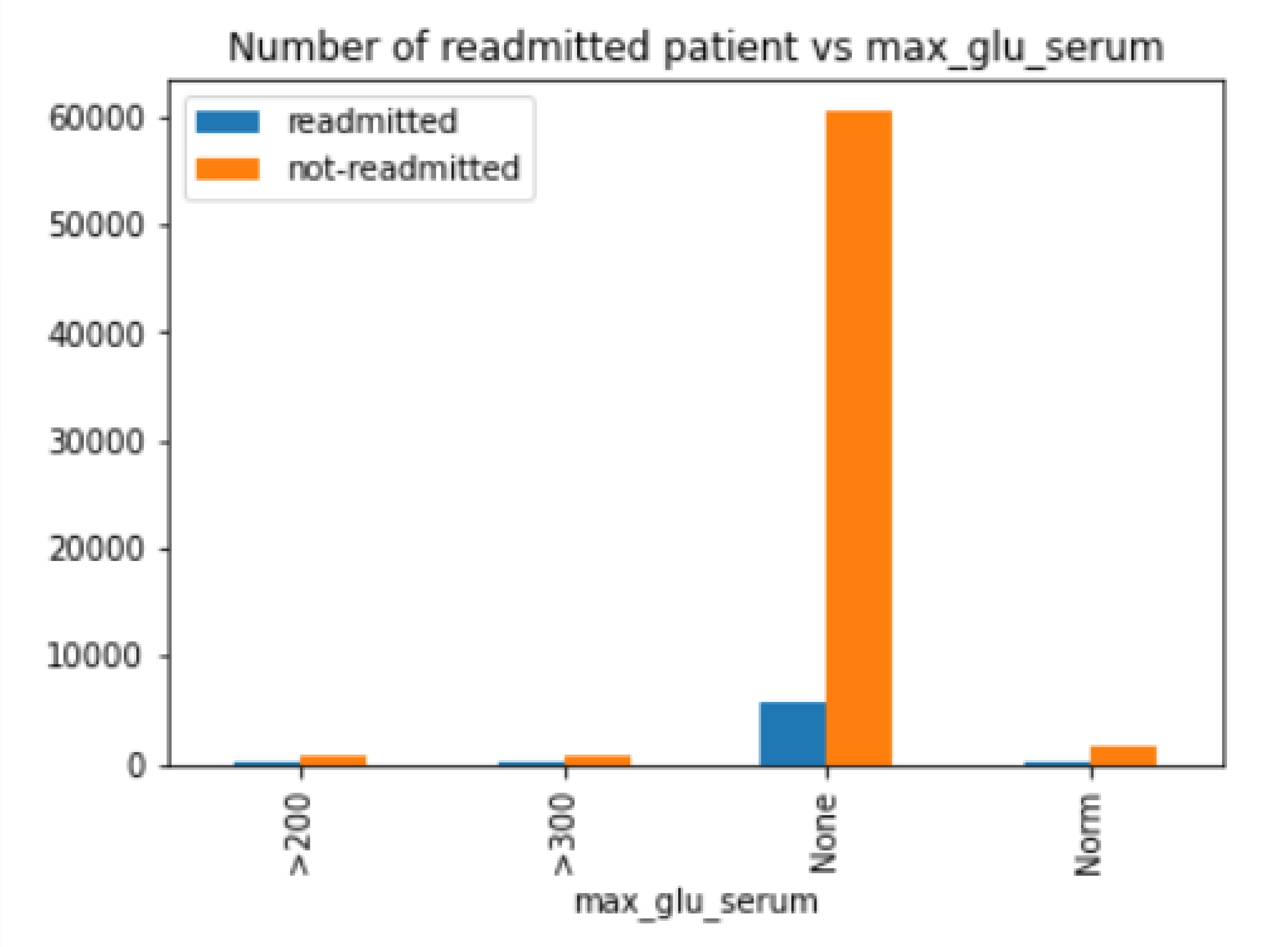




Link between readmission and variables

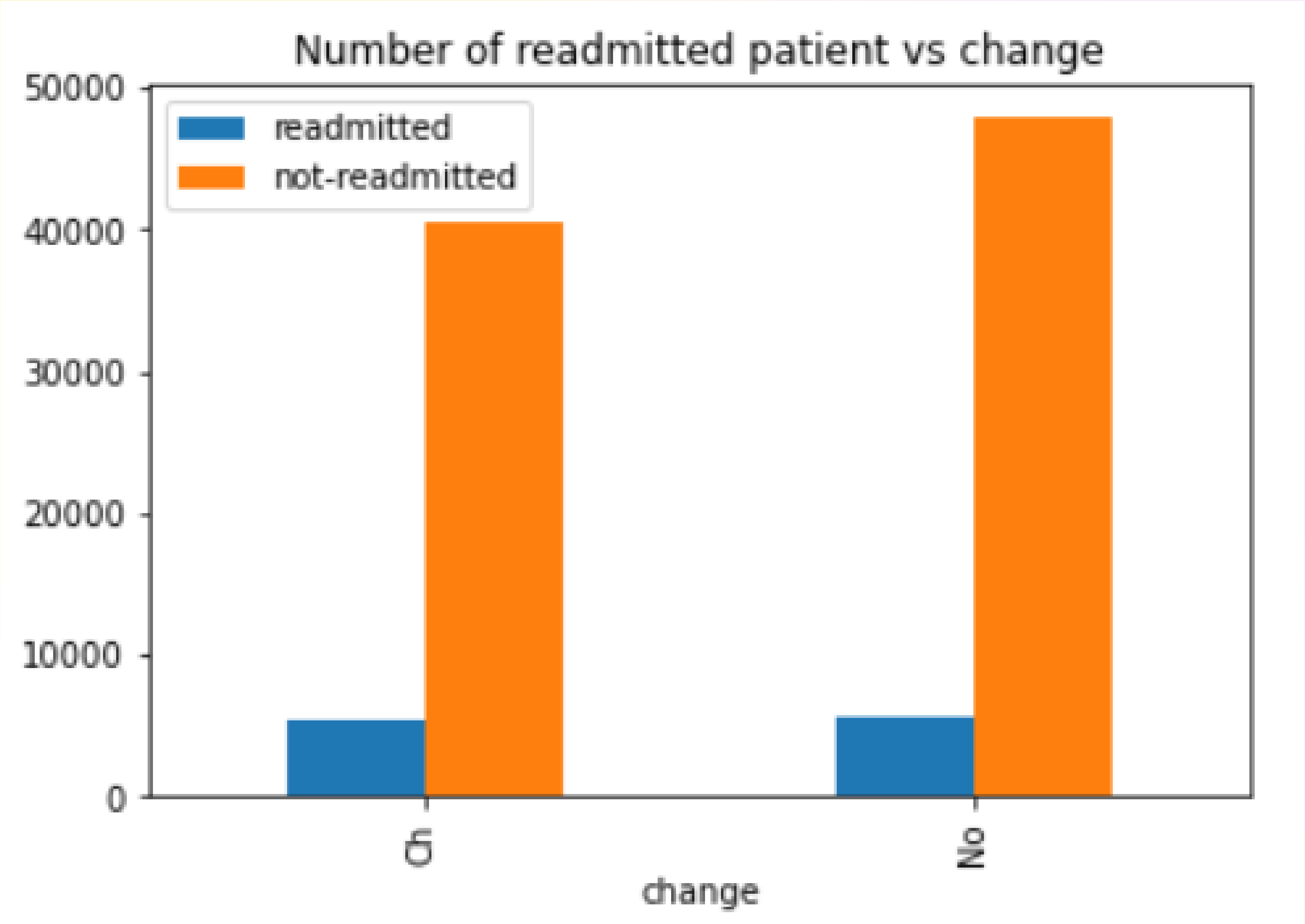


Indicates if there was any diabetic medication prescribed. Values: “yes” and “no”

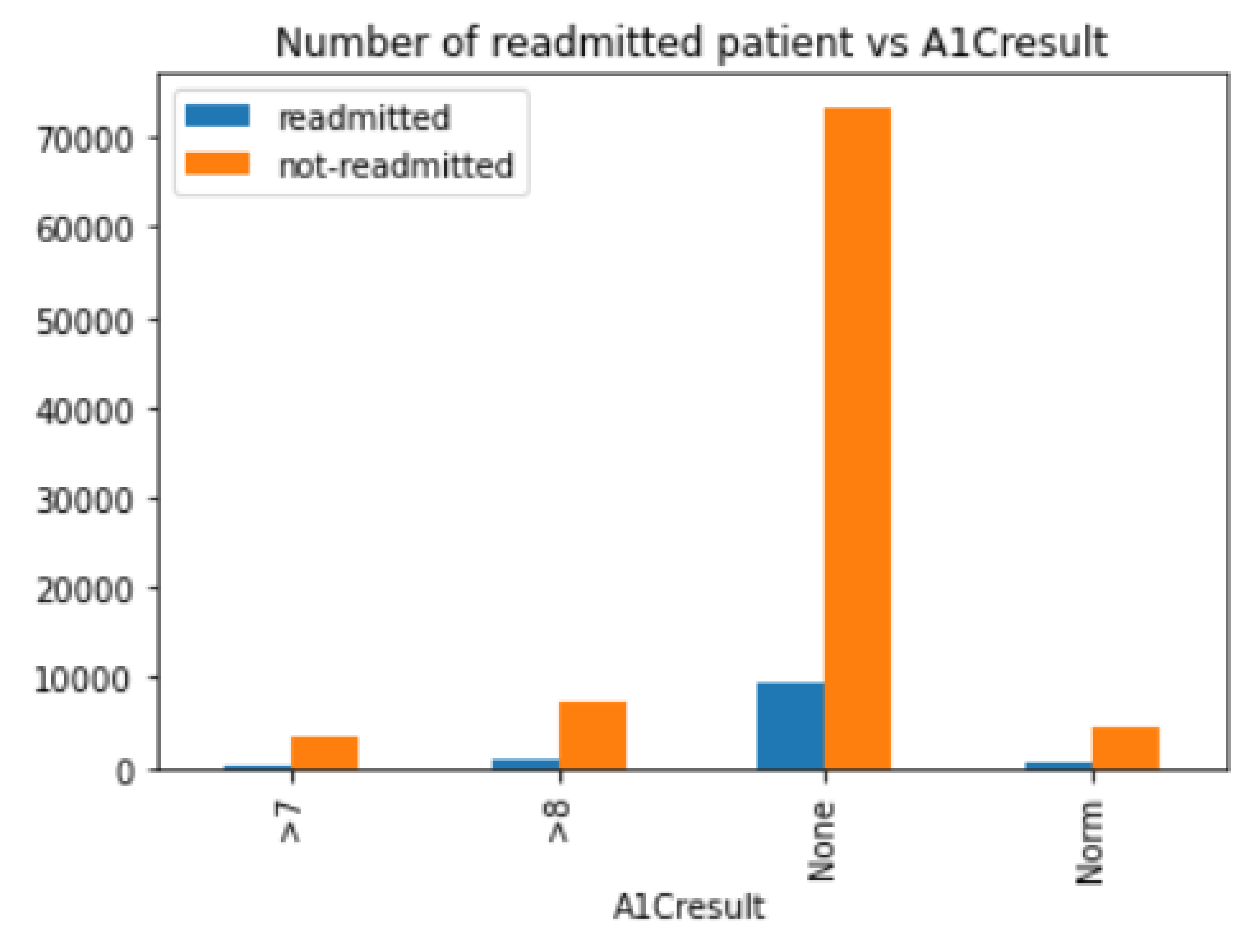


Indicates the range of the result or if the test was not taken. Values: “>200,” “>300,” “normal,” and “none” if not measured

Link between readmission and variables



Indicates if there was a change in diabetic medications (either dosage or generic name). Values: “change” and “no change”



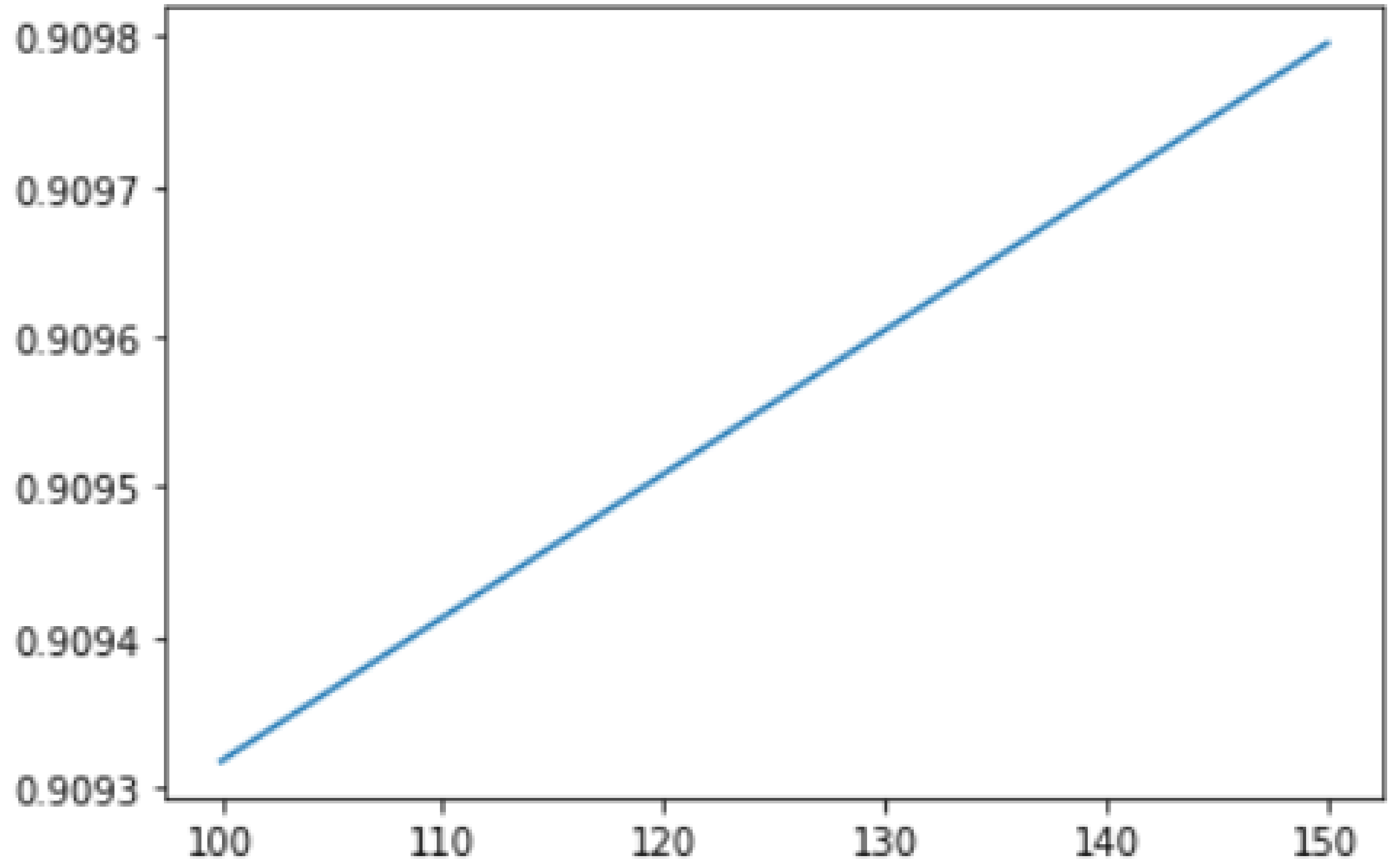
Indicates the range of the result or if the test was not taken. Values: “>8” if the result was greater than 8%, “>7” if the result was greater than 7% but less than 8%, “normal” if the result was less than 7%, and “none” if not measured.

Decision tree before re-sampling

```
Random Forest :
Best training accuracy =  0.910848450801398
Best parameters :  {'criterion': 'entropy',
'max_depth': 4}
Validation accuracy =  0.9117327372285228
test repartition :
 0.0    44455
1.0     4306
Name: readmitted, dtype: int64
Confusion matrix :
 [[44454  4303]
 [    1     3]]
```

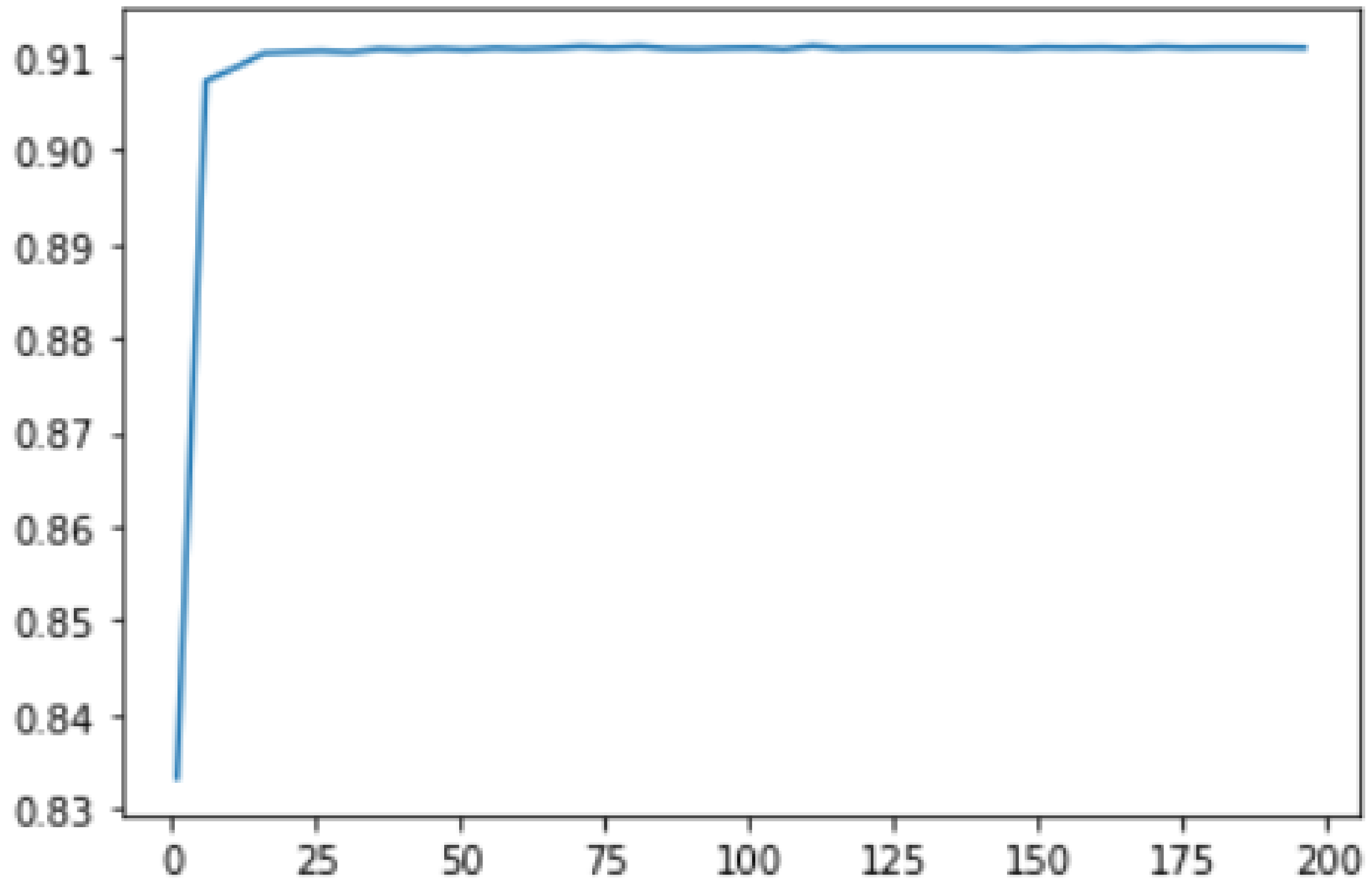
Bagging

```
Random Forest :
Best training accuracy =  0.9097956517607404
Best parameters :  {'max_features': 25, 'n_estimators': 150}
Validation accuracy =  0.9103586882959742
test repartition :
 0.0    44455
1.0     4306
Name: readmitted, dtype: int64
Confusion matrix :
 [[44303  4219]
 [  152    87]]
```



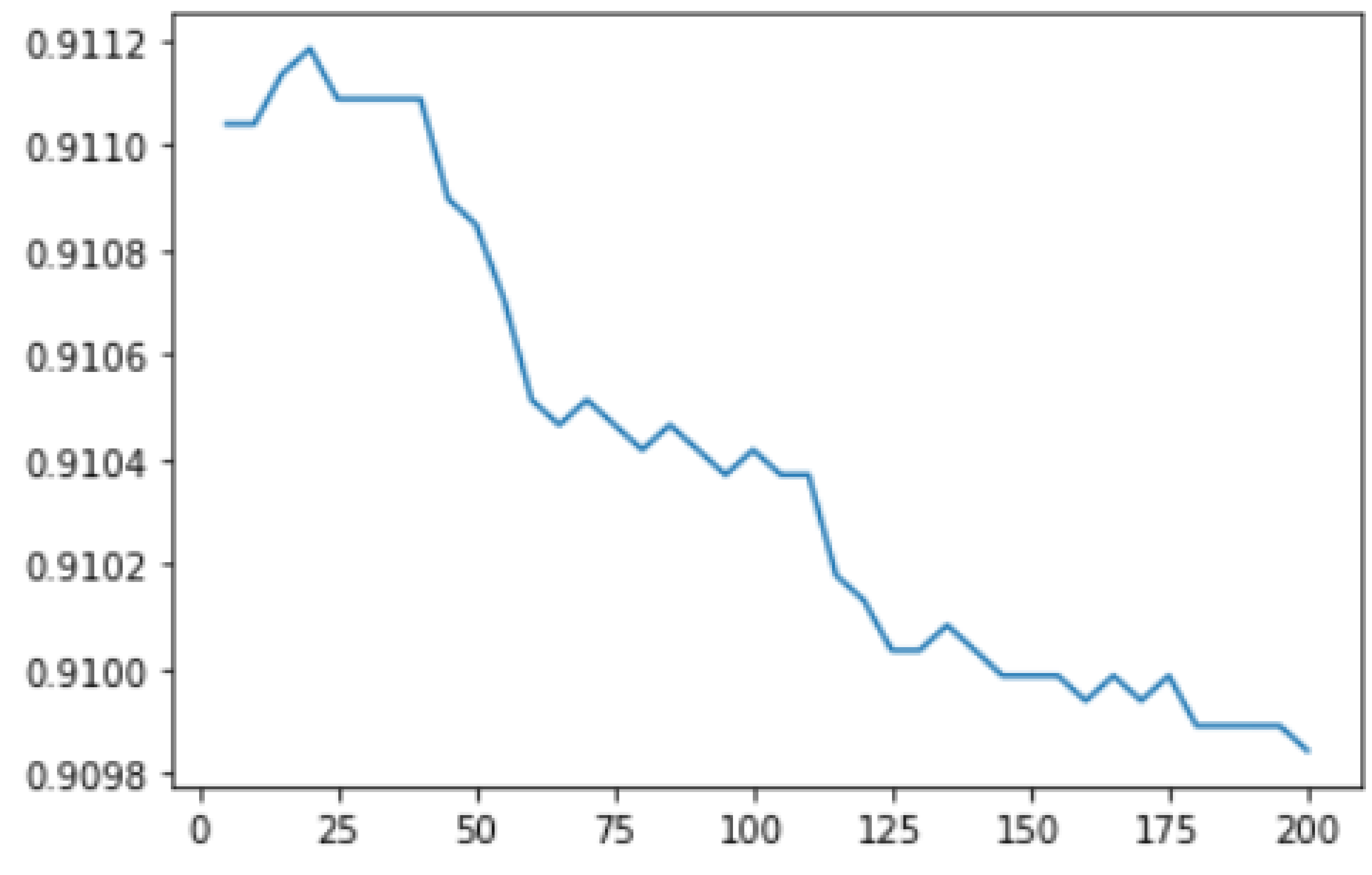
Random Forest before re-sampling

```
Random Forest :
Best training accuracy =  0.9110398655386754
Best parameters :  {'max_features': 'sqrt', 'n_estimators': 111}
Validation accuracy =  0.9116507044564304
test repartition :
 0.0    44455
1.0     4306
Name: readmitted, dtype: int64
Confusion matrix :
 [[44444  4297]
 [   11     9]]
```



Gradient Boosting before re-sampling

```
Gradiant boosting :
Best training accuracy =  0.9111834176578952
Best parameters :  {'learning_rate': 0.1, 'max_depth': 3, 'n_estimators': 20, 'random_state': 1, 'subsample': 1}
Validation accuracy =  0.911773753614569
test repartition :
 0.0    44455
1.0     4306
Name: readmitted, dtype: int64
Confusion matrix :
 [[44454  4301]
 [    1     5]]
```



Why resample ? →

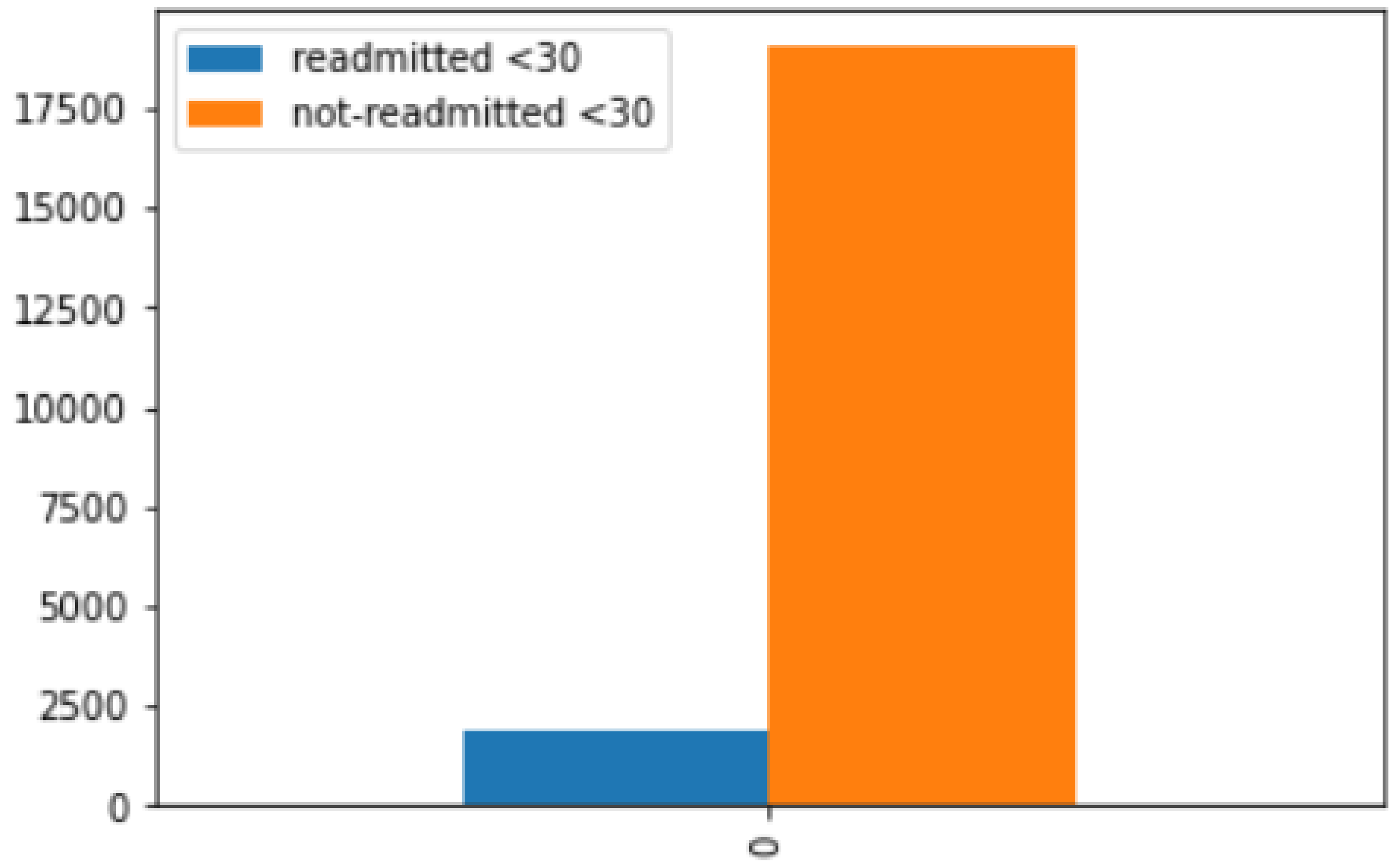
91.10%

- Non-readmitted patients of the training sample

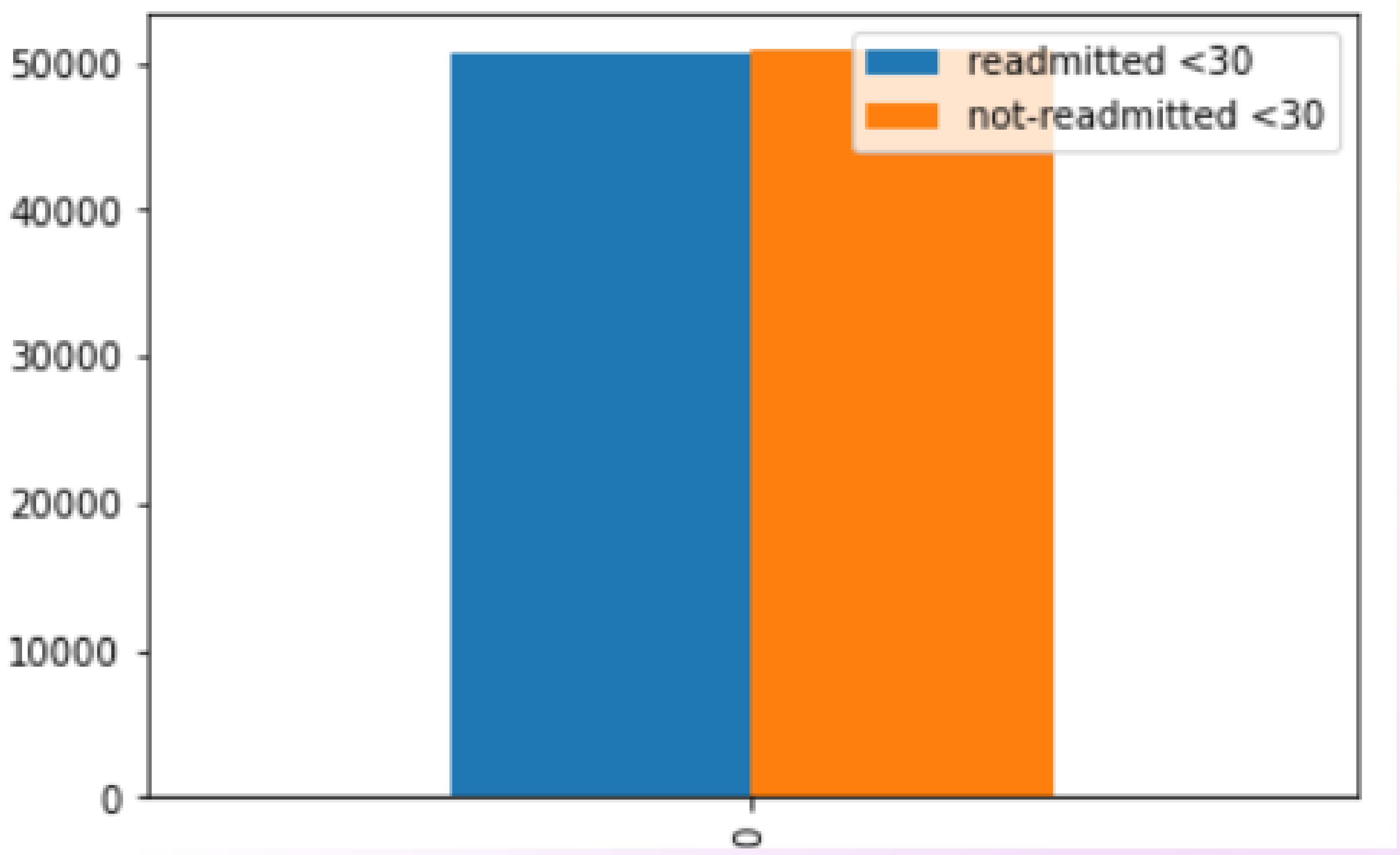
Only 8.89%

- Readmitted patients of the training sample

Before re-sampling



After re-sampling



Decision tree after re-sampling

```
Random Forest :
Best training accuracy =  0.9197740515070603
Best parameters :  {'criterion': 'entropy', 'max_depth': 16}
Validation accuracy =  0.9181037876998189
test repartition :
 1.0    12741
0.0    12657
Name: readmitted, dtype: int64
Confusion matrix :
 [[12260  1683]
 [  397 11058]]
```

```
pred = model_tree.predict(X)
print('Confusion matrix : \n', confusion_matrix(pred, y))
```

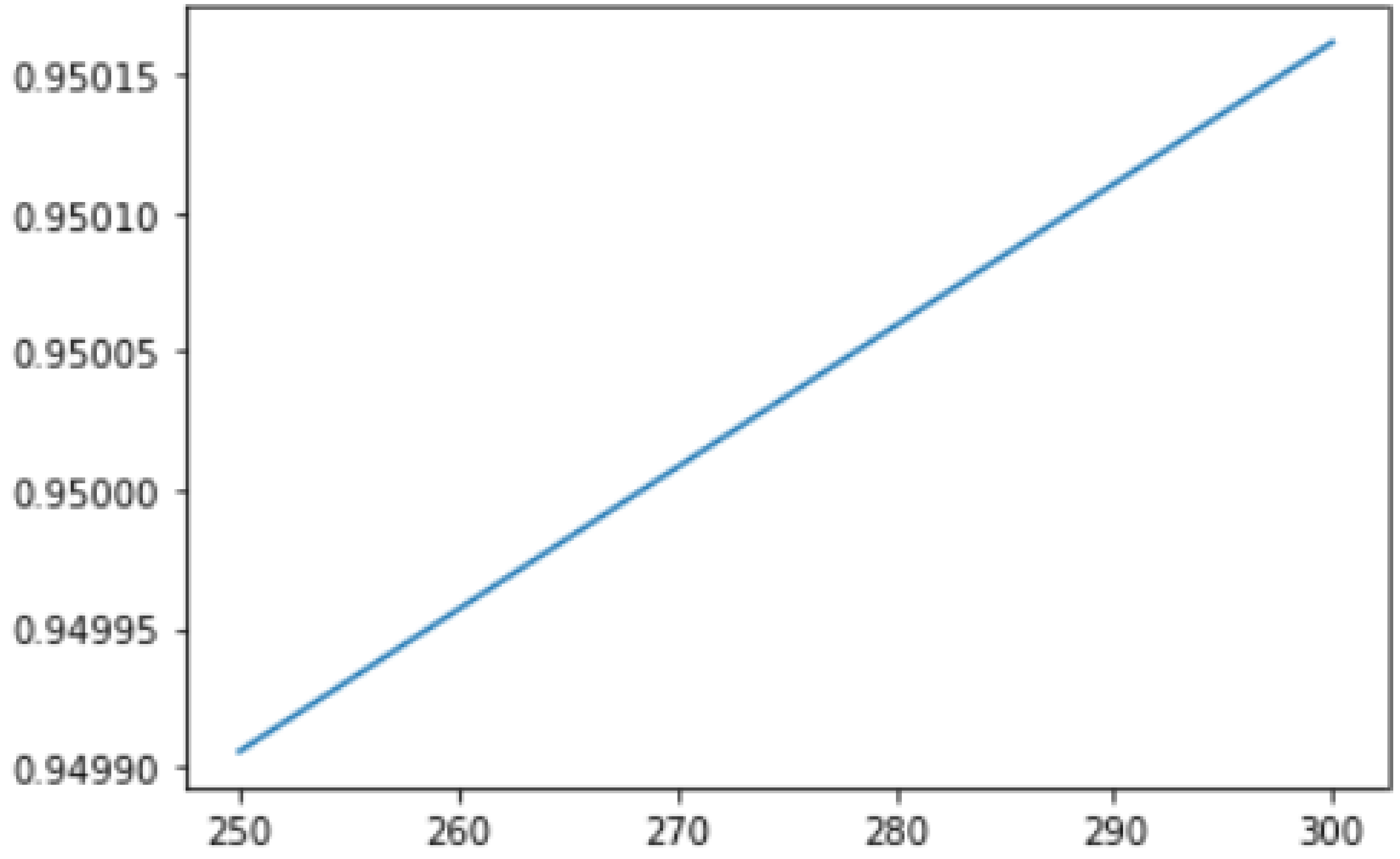
```
Confusion matrix :
 [[62791  4899]
 [  702  1266]]
```

Random Forest after re-sampling

```
pred = rf.predict(X)
print('Confusion matr
```

```
Confusion matrix :
 [[63480  1222]
 [   13  4943]]
```

```
Random Forest :
Best training accuracy =  0.9501614490549226
Best parameters :  {'criterion': 'entropy', 'max_features':
'sqrt', 'n_estimators': 300}
Validation accuracy =  0.9499960626821009
test repartition :
 1.0    12741
0.0    12657
Name: readmitted, dtype: int64
Confusion matrix :
 [[12644  1257]
 [   13 11484]]
```

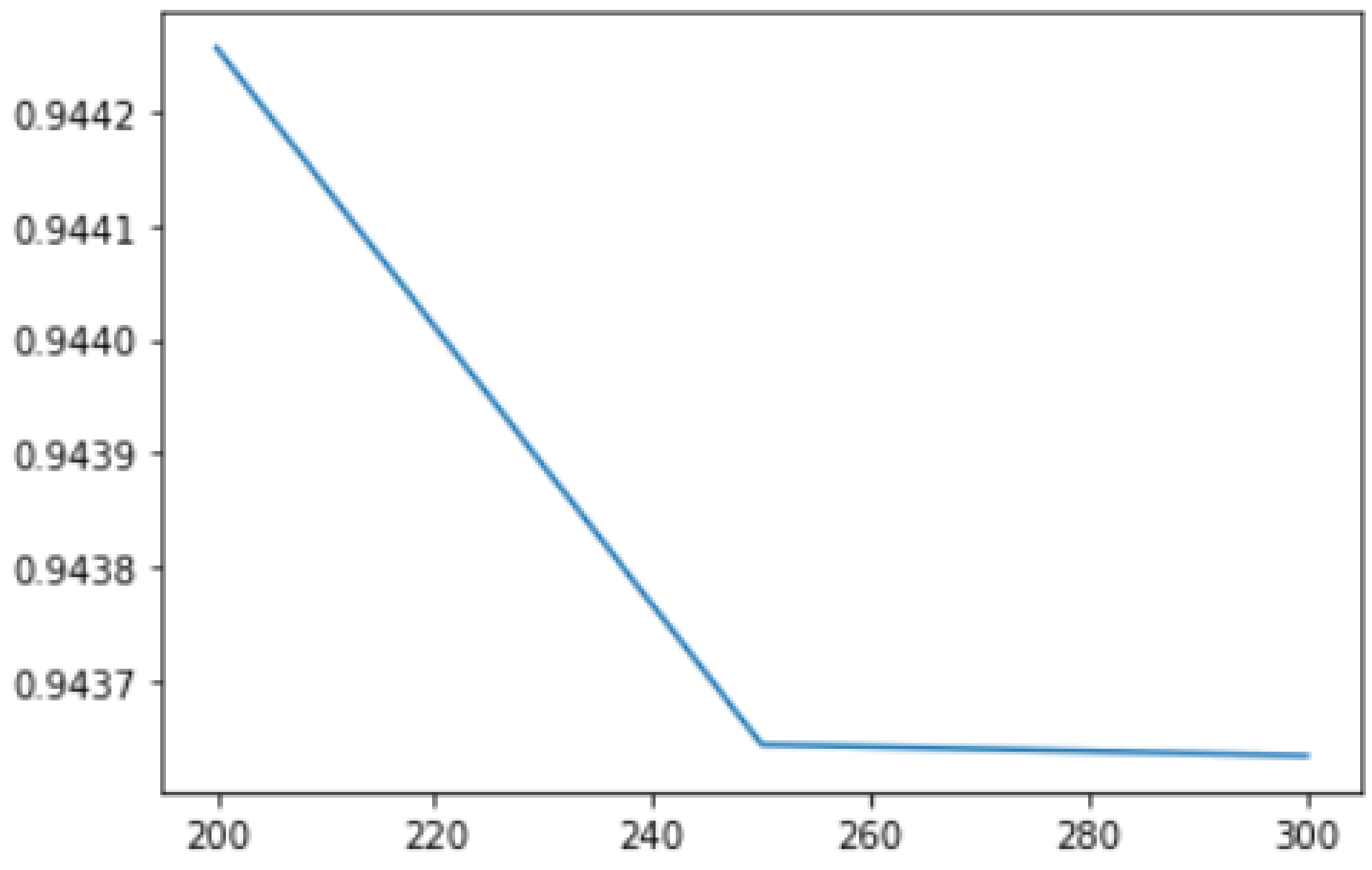


Gradient Boosting after re-sampling

```
Gradiant boosting :
Best training accuracy =  0.9442552406256688
Best parameters :  {'learning_rate': 1, 'loss': 'deviance', 'max_depth': 3, 'n_estimators': 200, 'random_state': 1, 'subsample': 1}
Validation accuracy =  0.943066383179778
test repartition :
 1.0    12741
0.0    12657
Name: readmitted, dtype: int64
Confusion matrix :
 [[12535  1324]
 [  122 11417]]
```

```
pred = gb.predict(X)
print('Confusion matr
```

```
Confusion matrix :
 [[63229  5668]
 [  264   497]]
```



Thank you !

Do you have any questions ?