Project 0 - DataLab mentorship program - 16/08/2023

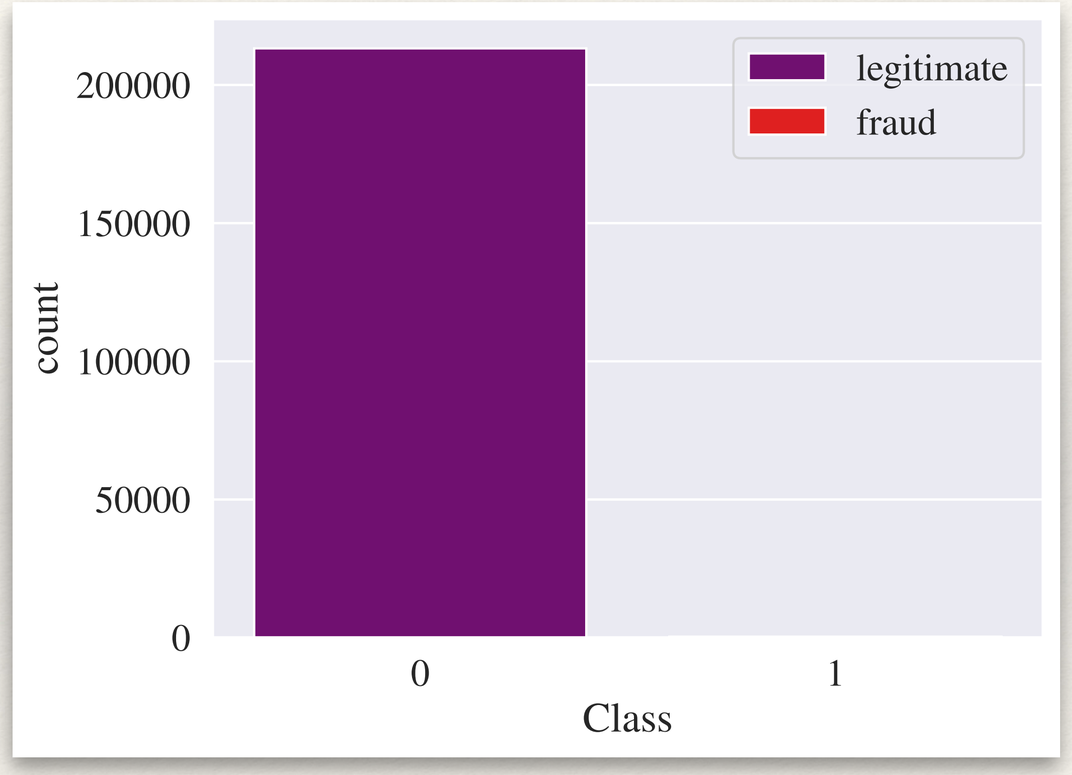
Credit card fraud detection

Bruno Ortega Goes
Mentor: Carlo(s)

serasa experian™

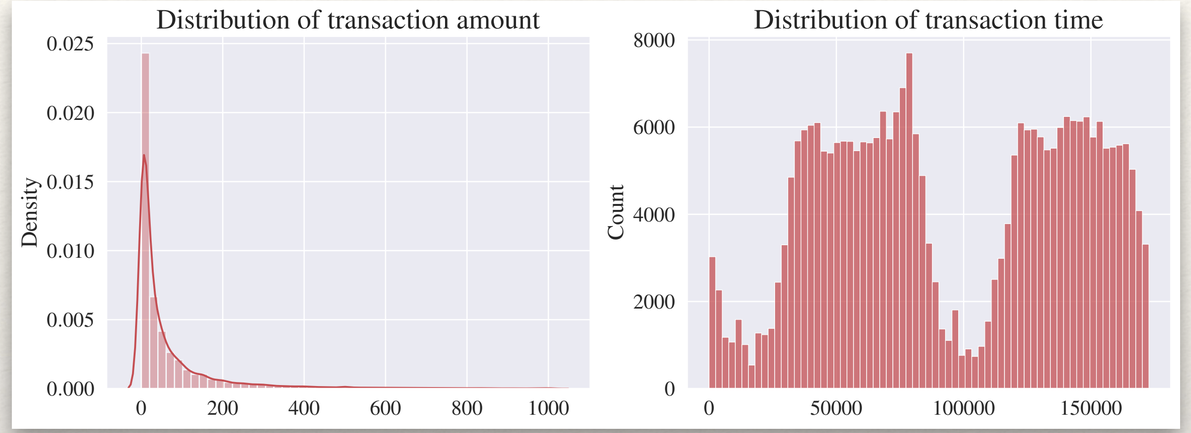
Highly imbalanced class: 99.82 % legitimate vs. 0.17 % frauds.

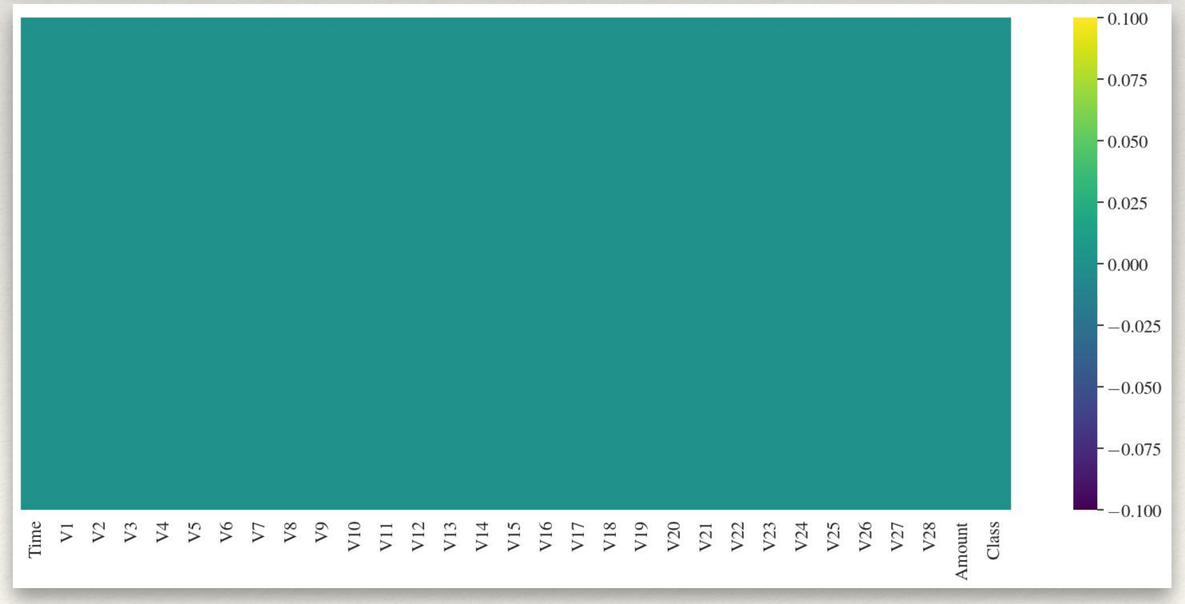
- **Goal:** classify if a transaction is a fraud (1) or not (0)
- **Challenge:** The frauds are less than 1% of the data set →Highly imbalanced dataset.
- **Commercial importance:**
  - Identifying a fraud is of interest for both the bank and the client to block the card.
  - We want a model that **minimizes** as much as possible the **confusion between real frauds and legitimate transactions**, so that the bank doesn't loose money and the client doesn't get his card blocked by misclassification.
    - Minimum less positives possible!



Data pre-processing

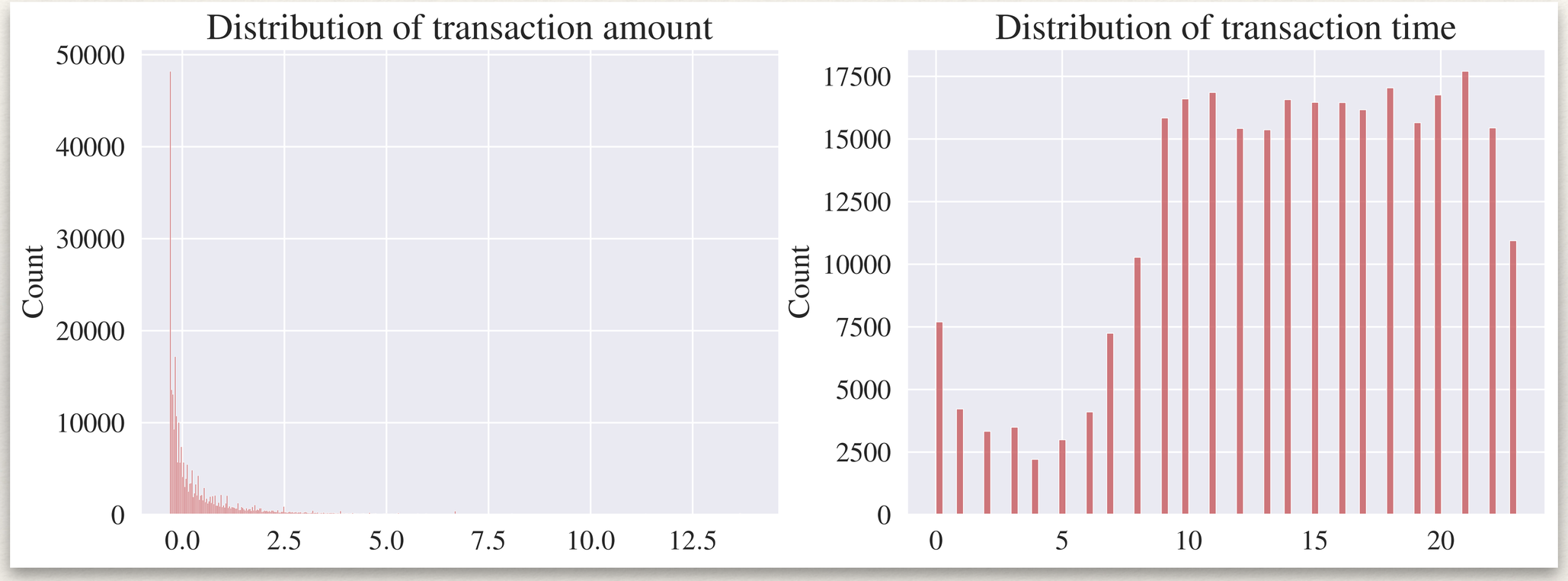
Good quality data set: treatment required in two features





- No missing values.
- Register of 2 days of transactions
  - Most of the transactions are low values.
    - Average of 100€.
  - Wave-ish behavior of the transactions.
    - night = the valleys, day = the peaks
- Amount and time must be treated before using training models.

Pre-processing: Rescaled transaction amount and time in hours.



What time does the frauds happen in average? The mean hour that the frauds occur is around 11.

Further preprocessing technical details.

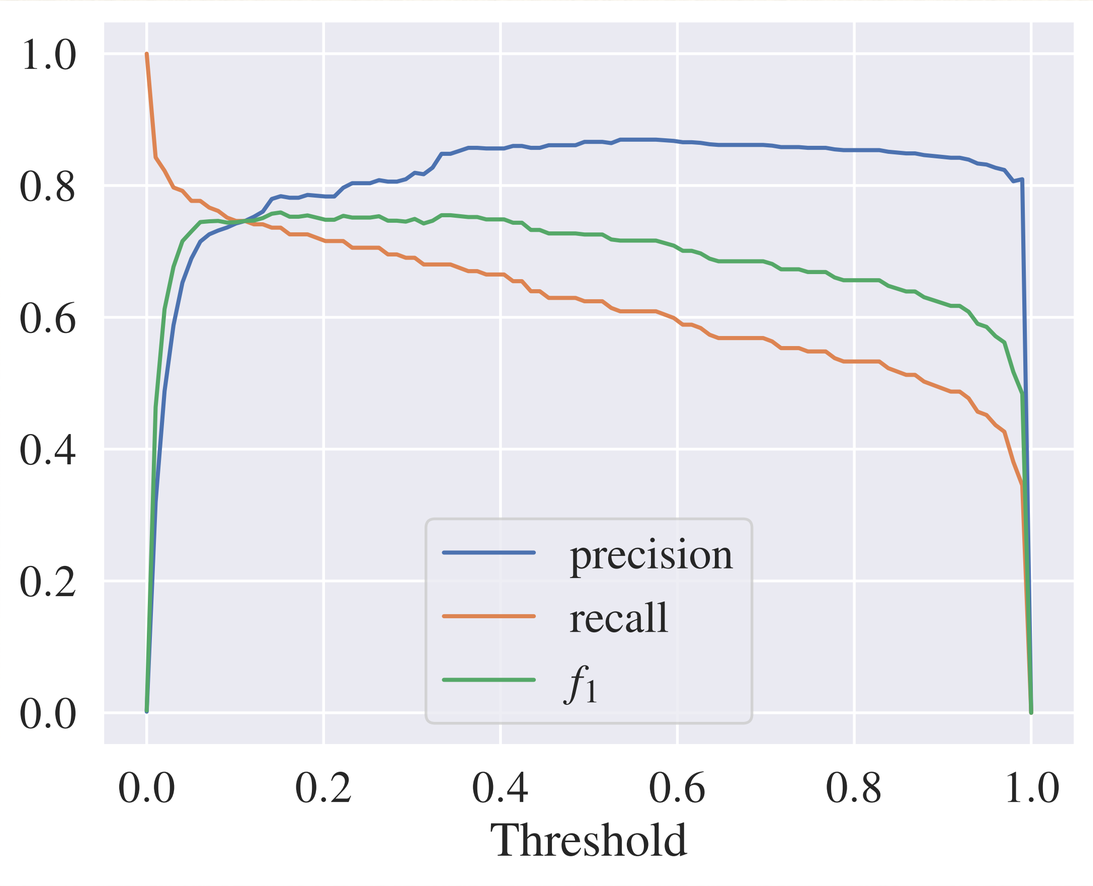
- Dataset is split into three sets:
  - Test set: to test the model
  - Training set: to train the model
  - Dev set: to study the threshold depend of important metrics
- We use **stratify** to make sure that all sets have the **same statistical distributions**.

Further preprocessing technical details.

- Models to be trained: **Logistic regression** & **Random Forest**
- Strategies to deal with the class imbalance:
  - **Do not treat it:** just apply the models to the original data set and check how it performs
  - Use **class weight** on the original data set: not aggressive balancing strategy
- **Resampling:**
  - **Oversample the minority class:** randomly create new instances of the minority class. Drawback: might overfit.
  - **Downsample the majority class:** randomly exclude the instances of the majority class until both are balanced. Drawback: We loose information.
  - **SMOTE:** Create synthetic instances of the minority class. Drawback: the created synthetic classes might not represent reality.

Brute data without balancing

Logistic regression threshold verification



- This is applied to the model trained in the dev set.
- We observe that for the brute data without balancing the best threshold is about **0.15**

Logistic regression: thr = 0.5 vs thr=0.16

Model	Precision	Recall	F1 score	ROC-AUC
0.5	0.79	0.65	0.72	0.96
0.16	0.77	0.74	0.75	0.96

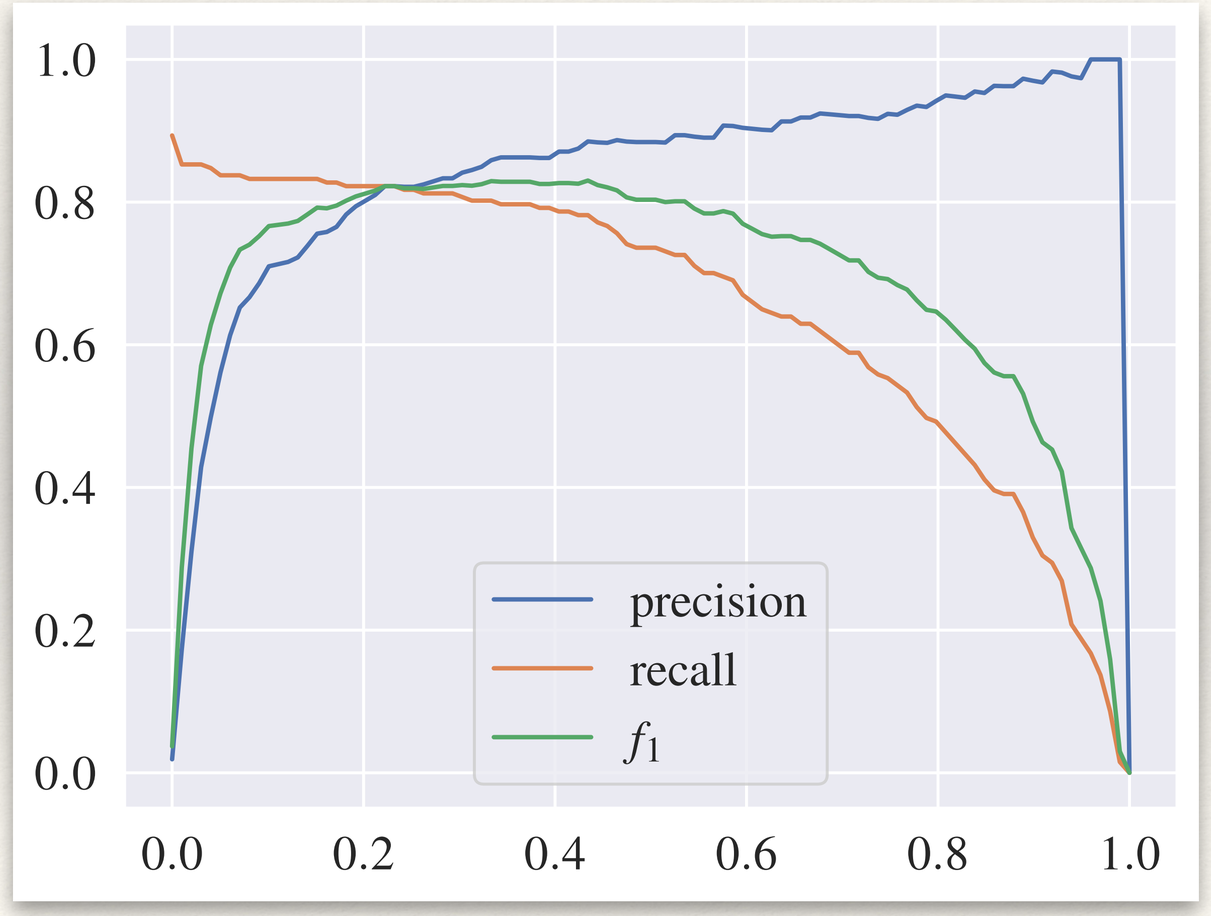
Thr = 0.5	Legitime	Fraud
Predicted legitime	113692	34
Predicted Fraud	68	129

Thr = 0.16	Legitime	Fraud
Predicted legitime	113682	44
Predicted Fraud	51	146

Take away messages logistic regression

- By adjusting the threshold we are able to improve significantly the recall and f1 score at the expanse of some precision (2%).
- We are getting more true frauds with this approach although the number of false positives raised by 10.
- We have a good AUC: 0.96 (for both, it is threshold independent).
- We keep the predictions with a threshold of 0.15 for the logistic regression.

Random Forest threshold verification



- #trees=500, max depth=20 for all RF model training presented from now on.
- This is applied to the model trained in the dev set.
- We observe that for the brute data without balancing the best threshold is about **~0.4**
- This choice provides a good precision and a good f1

Random forest: thr = 0.5 vs thr=0.45

Model	Precision	Recall	F1 score	ROC-AUC
0.5	0.93	0.78	0.85	0.96
0.45	0.92	0.8	0.85	0.96

Thr = 0.5	Legitime	Fraud
Predicted legitime	113714	12
Predicted Fraud	43	154

Thr = 0.45	Legitime	Fraud
Predicted legitime	113712	14
Predicted Fraud	40	157

Take away messages random forest

- By adjusting the threshold we are able to improve slightly the recall expanse of some precision (1%).
- We are getting more true frauds with the adjusted threshold, but in general the metrics do not change much, and the model performs well in classifying the frauds.
- We have a good AUC: 0.96 (for both, it is threshold independent).
- Since we had less false positives with 0.5 we keep this threshold for models comparison.

Random forest performs better.

Model	Precision	Recall	F1 score	ROC-AUC
Logistic regression (thr = 0.15)	0.77	0.74	0.75	0.96
Random forest (Thr = 0.5)	0.93	0.78	0.85	0.96

LR: Thr = 0.15	Legitime	Fraud
Predicted legitime	113682	44
Predicted Fraud	51	146

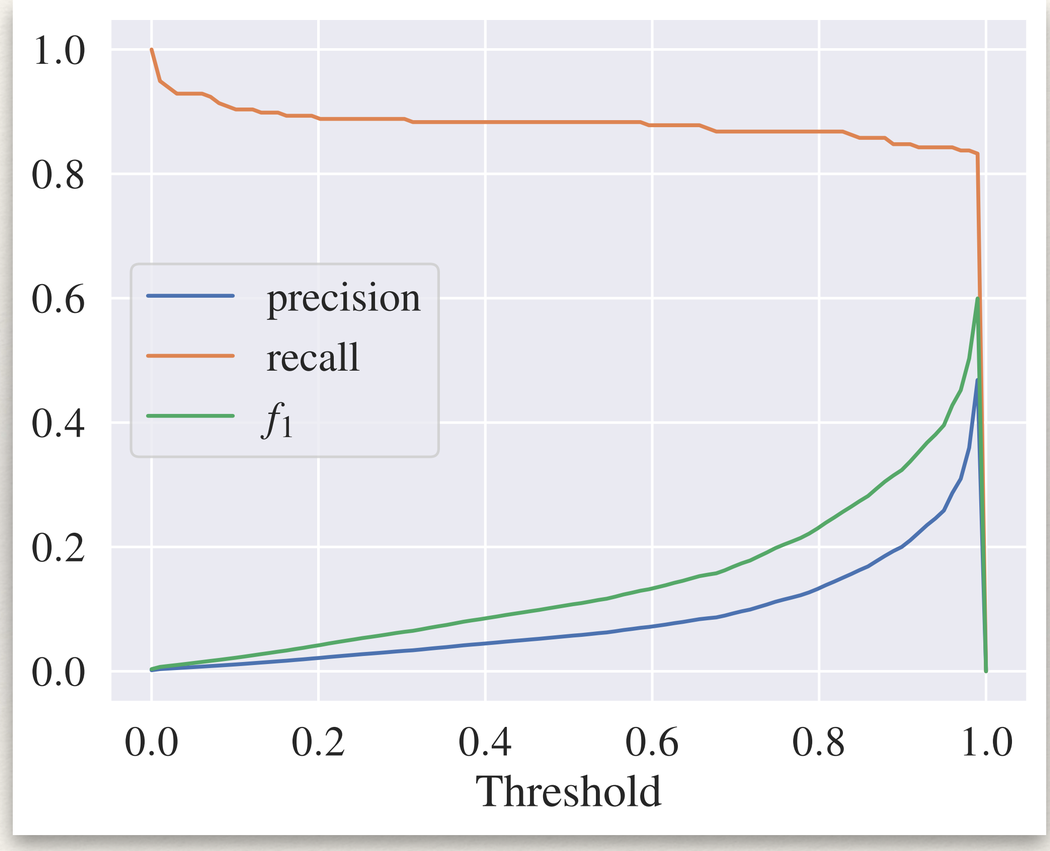
RF: Thr = 0.5	Legitime	Fraud
Predicted legitime	113714	12
Predicted Fraud	43	154

Random forest performs better.

- By looking at the confusion matrix and assuming the transactions are about 1000€
  - The logistic regression makes us loose 44000€ by legitimating frauds
  - Random forest the misclassification is only of 12000€, ~3.5 less than LG

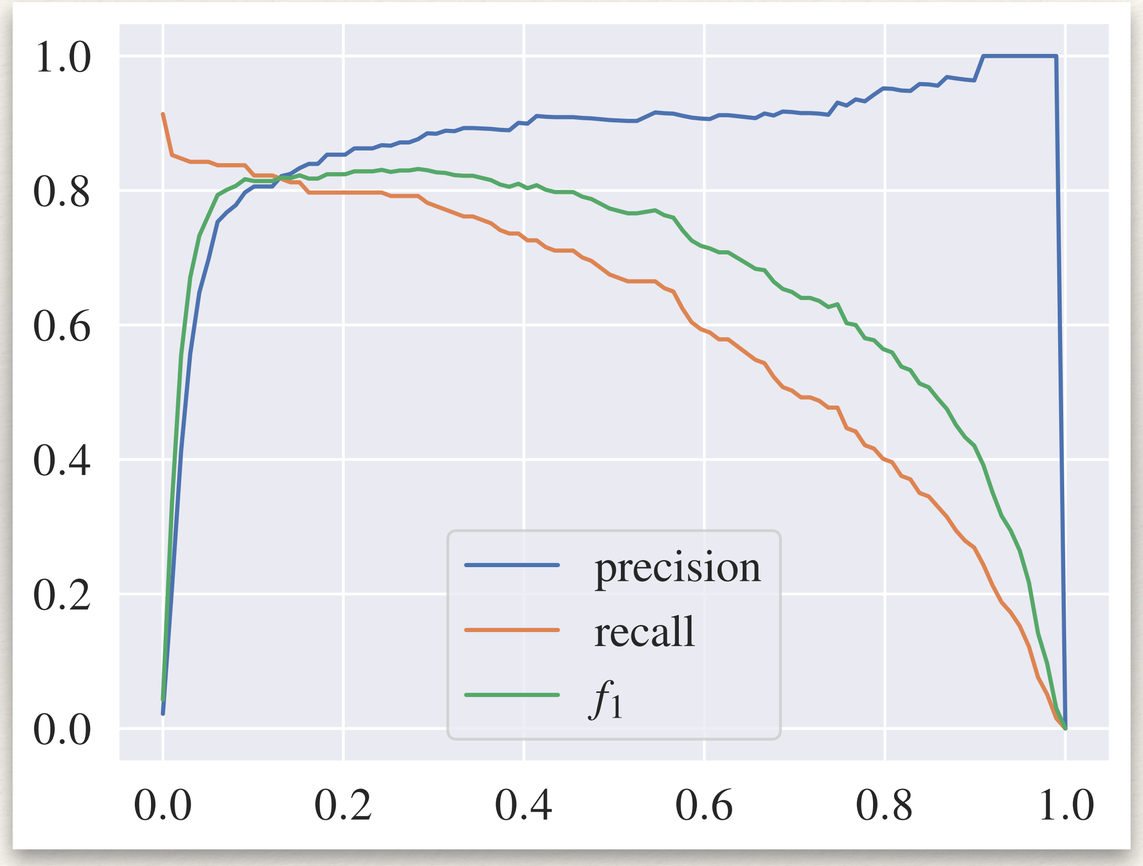
Brute data with balancing

Logistic regression threshold verification



- This is applied to the model trained in the dev set with class balance activated.
- I'll drop the Logistic regression as it is very imprecise and do not provide a good F1 score.

Random Forest threshold verification



- #trees=500, max depth=20
- This is applied to the model trained in the dev set balanced.
- We observe that for the brute data without balancing the best threshold is about ~**0.5**
- This choice provides a good precision and a good f1

Random forests: balanced vs. not balanced

Thr = 0.5	Precision	Recall	F1 score	ROC-AUC
Not balanced	0.93	0.78	0.85	0.96
Balanced	0.94	0.77	0.85	0.97

Not balanced	Legitime	Fraud
Predicted legitime	113714	12
Predicted Fraud	43	154

Balanced	Legitime	Fraud
Predicted legitime	113716	10
Predicted Fraud	45	152

Take away messages random forest

- By balancing the classes we obtain a better precision and AUC, while the recall and F1 remain similar
- We are getting less true frauds with the balanced data set, but we are also predicting less frauds as legitimate.
- We have a good AUC: 0.97>0.96 for the balanced data.
- At this point the best model and strategy we have is the Random forest with 500 forests 20 as the max depth and balancing the classes.

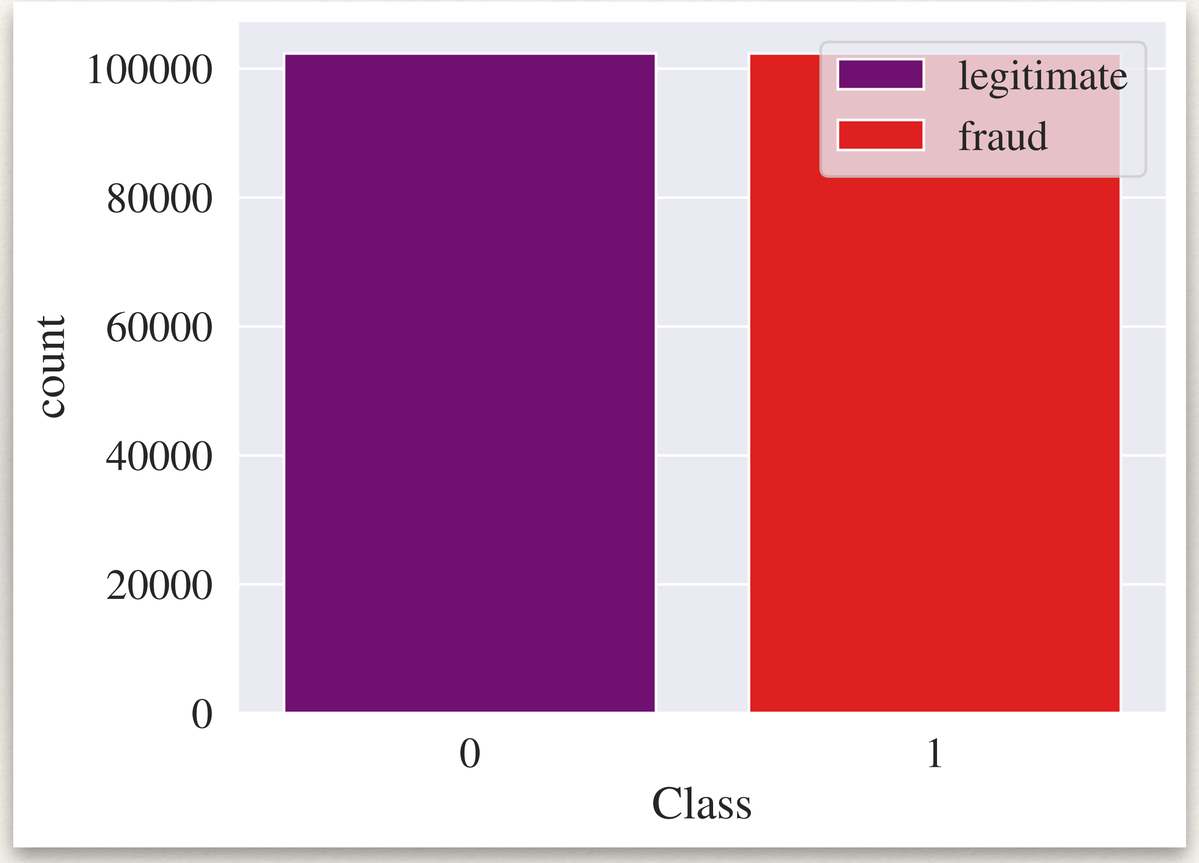
Random forest performs better.

- By looking at the confusion matrix and assuming the transactions are about 1000€
  - Before we were predicting 12000€ as legitimate transactions
  - Now we predict 10000€ as legitimate, a considerable improvement of 2k€ just by balancing the classes!

Resampling

Oversample minority class

- This first strategy: add more copies of the frauds.
- Proportion between the classes 1-to-1 proportion.
- I drop the logistic regression for the same reason of the former case.



Random forests: balanced vs. upsampled

	Precision	Recall	F1 score	ROC-AUC
Balanced	0.94	0.77	0.85	0.97
Upsample, thr=0.5	0.93	0.75	0.83	0.97
Up, thr = 0.55	0.94	0.73	0.82	0.97

Balanced	Legitime	Fraud
Predicted legitime	113716	10
Predicted Fraud	45	152

Up, 0.5	Legitime	Fraud
Predicted legitime	113715	11
Predicted Fraud	50	147

Up, 0.55	Legitime	Fraud
Predicted legitime	113714	9
Predicted Fraud	53	144

Similar performances. Class weights still a better option.

- The Random forest trained with the upsampled or the brute data set with class weight balancing performs similarly.
- There is no clear advantage in using the upsample in this case: we stick with the class weighting strategy.
- I also tried the downsampling, but the performance is not worth discussing. To see these results I provide the GitHub repository of this project in the end.

SMOTE: Synthetic Minority Over-sampling Technique

- Creates synthetic samples for minority class.
- Generates new instances **along line segments** between existing instances (problem: not convex set).
- **Benefits of SMOTE**
  - Balances class distribution, reducing bias.
  - Enhances model's ability to learn from minority class.
  - Reduces overfitting and improves generalization.
- **Considerations**
  - Works well with structured data.
  - Can be combined with other techniques.
  - Choose appropriate k value for nearest neighbors.

Random forests: balanced vs. not balanced

	Precision	Recall	F1 score	ROC-AUC
Balanced	0.94	0.77	0.85	0.97
SMOTE, thr=0.5	0.81	0.8	0.8	0.98
SMOTE, thr = 0.85	0.96	0.68	0.79	0.98

Balanced	Legitime	Fraud
Predicted legitime	113716	10
Predicted Fraud	45	152

SM, 0.5	Legitime	Fraud
Predicted legitime	113688	38
Predicted Fraud	39	158

SM, 0.85	Legitime	Fraud
Predicted legitime	113721	5
Predicted Fraud	64	133

Discussion

- The best strategy, within the studied models, is to train a random forest with the data generated synthetically with the SMOTE algorithm with a threshold of 0.85.
- It misclassifies only 5 frauds as a legitimate transactions: 5k€
  - Much less the 12k€ for the class weighted instance
  - The drawback is that we are using synthetic data, and the recall
- Up and down sampling proved to perform poorly for this data set.
- Logistic regression was dropped due to its imprecision.

Thanks for the attention!