CREDIT CARD FRAUDULENT TRANSACTIONS DETECTION

Project presentation of AI for Cybersecurity

University of Pisa - Master Degree in Cybersecurity 22/23

DATASET KNOWLEDGE

Credit card transactions made by european cardholders in 2 days of September 2013. It contains only numeric input variables which are the result of the PCA transformation, so dimensionality reduction already applied to database.

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V23	V24	V25	V26	V27	V28	Amount	Class
0	0.0	-1.359807	-0.072781	2.536347	1.378155	-0.338321	0.462388	0.239599	0.098698	0.363787	...	-0.018307	0.277838	-0.110474	0.066928	0.128539	-0.189115	0.133558	-0.021053	149.62	0
1	0.0	1.191857	0.266151	0.166480	0.448154	0.060018	-0.082361	-0.078803	0.085102	-0.255425	...	-0.225775	-0.638672	0.101288	-0.339846	0.167170	0.125895	-0.008983	0.014724	2.69	0

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...
count	284807.000000	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	...
mean	94813.859575	1.168375e-15	3.416908e-16	-1.379537e-15	2.074095e-15	9.604066e-16	1.487313e-15	-5.556467e-16	1.213481e-16	-2.406331e-15	...
std	47488.145955	1.958696e+00	1.651309e+00	1.516255e+00	1.415869e+00	1.380247e+00	1.332271e+00	1.237094e+00	1.194353e+00	1.098632e+00	...
min	0.000000	-5.640751e+01	-7.271573e+01	-4.832559e+01	-5.683171e+00	-1.137433e+02	-2.616051e+01	-4.355724e+01	-7.321672e+01	-1.343407e+01	...
25%	54201.500000	-9.203734e-01	-5.985499e-01	-8.903648e-01	-8.486401e-01	-6.915971e-01	-7.682956e-01	-5.540759e-01	-2.086297e-01	-6.430976e-01	...
50%	84692.000000	1.810880e-02	6.548556e-02	1.798463e-01	-1.984653e-02	-5.433583e-02	-2.741871e-01	4.010308e-02	2.235804e-02	-5.142873e-02	...
75%	139320.500000	1.315642e+00	8.037239e-01	1.027196e+00	7.433413e-01	6.119264e-01	3.985649e-01	5.704361e-01	3.273459e-01	5.971390e-01	...
max	172792.000000	2.454930e+00	2.205773e+01	9.382558e+00	1.687534e+01	3.480167e+01	7.330163e+01	1.205895e+02	2.000721e+01	1.559499e+01	...

```
RangeIndex: 284807 entries, 0 to 284806
Data columns (total 31 columns):
```

DATASET KNOWLEDGE

31 features and most of these has confidentiality protection for security reasons (V1,V2, ….., V28). No possibility to have other informations, instead in 3 features:

- **Time**: Interval quantity attribute starts from 0 (first transaction) and corresponds to the second between the actual and the first transactions.

- **Amount**: Ratio quantity attribute represents the amount of euros in each transaction

- **Class**: Binary attribute for labelling data objects, 1 for “fraudulent” and 0 for “legal”

WORK STEPS

Classification & Performance Evaluation

3 different learners to create models:

1) LOGISTIC REGRESSION
2) NEURAL NETWORK
3) RANDOM FOREST

Dataset Rebalanced

Sampling process due to the imbalance dataset.

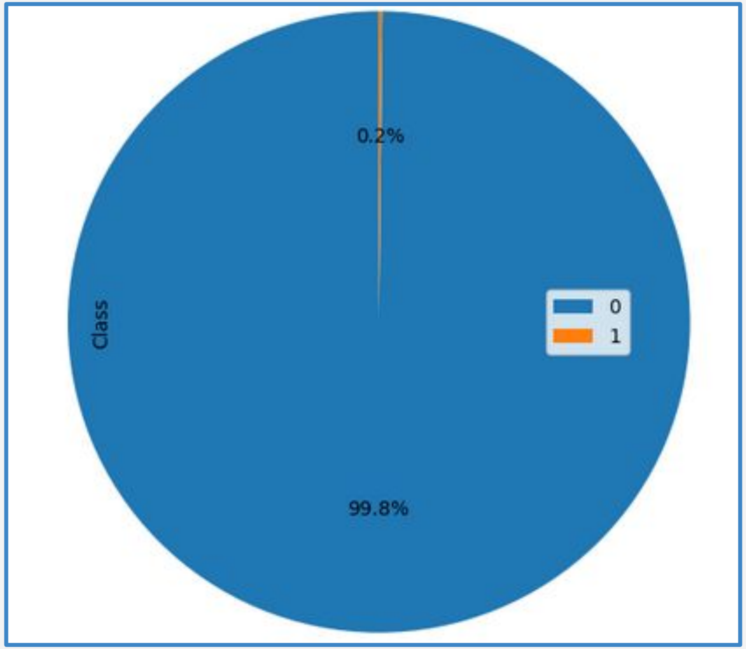
Preprocessing & Data Cleaning

Remove incomplete data and duplicated data objects, normalize features.

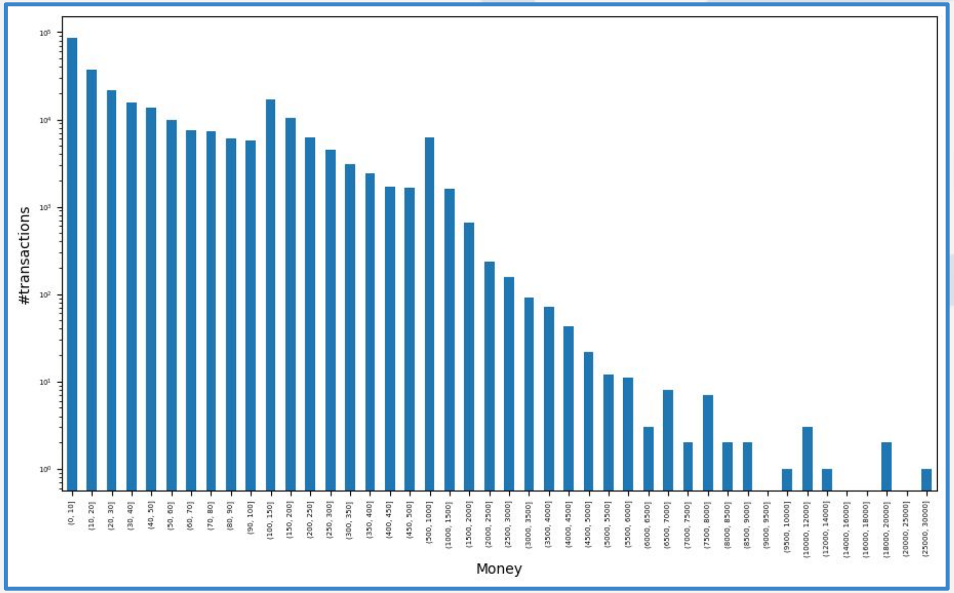
Data Visualization & Data Analysis

Visualize data distribution, plot data to acquire informations.

DATA VISUALIZATION & DATA ANALYSIS

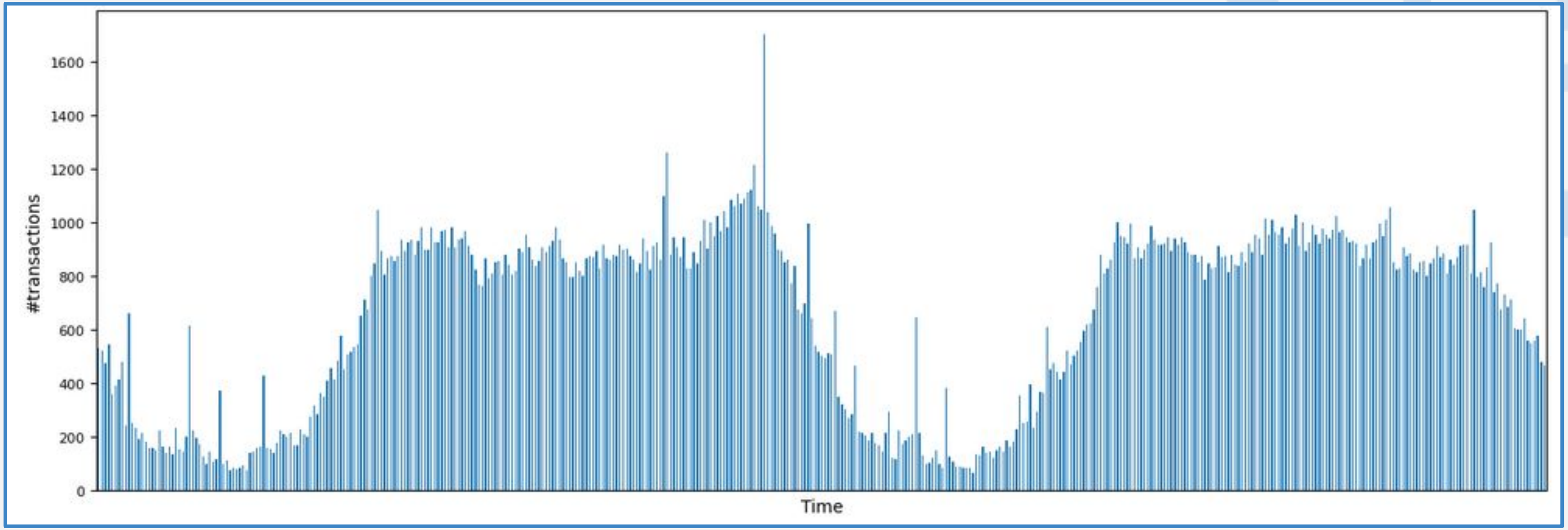


Pie Chart - Imbalance transactions dataset: Label 0 legal, Label 1 fraudulent



Bar Chart - Number of transactions depending on money ranges in a log scale

DATA VISUALIZATION & DATA ANALYSIS



Bar Chart - Number of transactions depending on time ranges (steps of 400 sec)

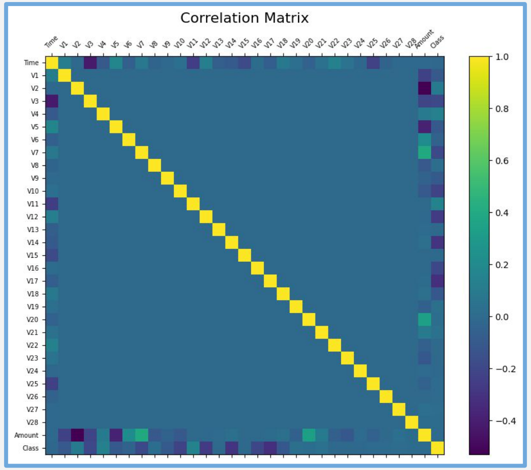
PREPROCESSING

NULL VALUES: no NULL values in the dataset

DUPLICATES: discard 1081 duplicate transactions

NORMALIZATION: normalize all features with Min - Max Normalization to obtain more efficient learning phase above all with Neural Network.

```
 0   Time    284807 non-null  float64
 1   V1      284807 non-null  float64
 2   V2      284807 non-null  float64
 3   V3      284807 non-null  float64
 4   V4      284807 non-null  float64
 5   V5      284807 non-null  float64
 6   V6      284807 non-null  float64
 7   V7      284807 non-null  float64
 8   V8      284807 non-null  float64
 9   V9      284807 non-null  float64
 10  V10     284807 non-null  float64
 11  V11     284807 non-null  float64
 12  V12     284807 non-null  float64
 13  V13     284807 non-null  float64
 14  V14     284807 non-null  float64
 15  V15     284807 non-null  float64
 16  V16     284807 non-null  float64
 17  V17     284807 non-null  float64
 18  V18     284807 non-null  float64
 19  V19     284807 non-null  float64
 20  V20     284807 non-null  float64
 21  V21     284807 non-null  float64
 22  V22     284807 non-null  float64
 23  V23     284807 non-null  float64
 24  V24     284807 non-null  float64
 25  V25     284807 non-null  float64
 26  V26     284807 non-null  float64
 27  V27     284807 non-null  float64
 28  V28     284807 non-null  float64
 29  Amount  284807 non-null  float64
 30  Class   284807 non-null  int64
```




```
0    284315
1       492
Name: Class, dtype: int64
```

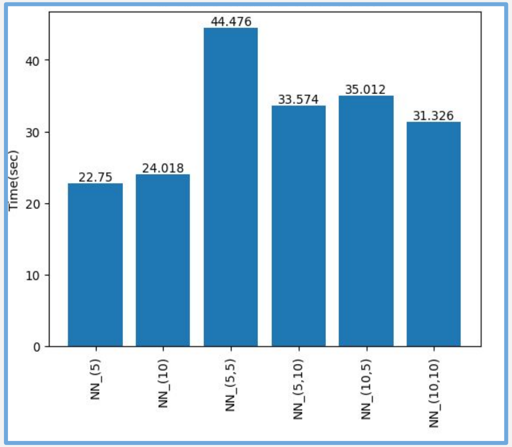
```
0    283253
1       473
Name: Class, dtype: int64
```

CLASSIFICATION CHOICES

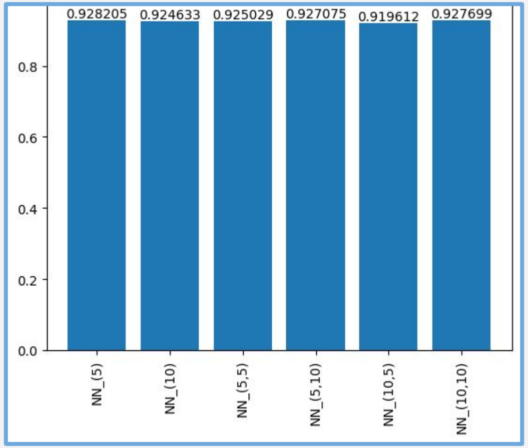
- Exploit all features to train classifiers: Time, Amount, V1, V2, ..., V28
- Test all classifiers with the imbalanced dataset and the rebalanced dataset:
  - under_sampler = RandomUnderSampler(sampling_strategy = 'majority')
  - over_sampler = RandomOverSampler(sampling_strategy = 'minority')
- 5-folds cross validation in all cases:
  - skf = StratifiedKFold(n_splits = 5, shuffle = True, random_state = 123)
- Main evaluation metrics: AUC, Recall and Precision

CLASSIFICATION - IMBALANCED DS
Neural Networks cross validation

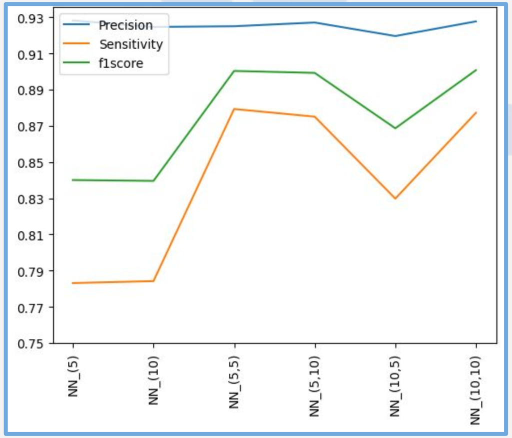
neural_net1 = MLPClassifier(solver = 'adam', max_iter = 200, hidden_layer_sizes = (5), random_state = 123)



Fit time of all NNs



AUC evaluation of NNs



Other metrics for all NNs

AUC values are very similar in all configurations, as the precision. Evaluating Fit Time, Sensitivity and the F1-score I chose the last network, NN(10,10) for next tests.

CLASSIFICATION - IMBALANCED DS

MODEL	AUC	RECALL	PRECISION	F1-SCORE	ACCURACY
LR	0.927	0.759	0.927	0.823	0.999
NN(10,10)	0.928	0.877	0.928	0.901	0.999
RF	0.969	0.885	0.969	0.922	0.999

Initially i trained classifiers without any sampling technique as starting point to observe changes in performance metrics with future resample train sets.

CLASSIFICATION - REBALANCED TS

	MODEL	AUC	RECALL	PRECISION	F1-SCORE	ACCURACY
T	LR	0.533	0.943	0.533	0.556	0.978
	NN(10,10)	0.514	0.937	0.52	0.529	0.941
T	RF	0.964	0.888	0.964	0.923	0.999

Random Oversampling of Train-Set

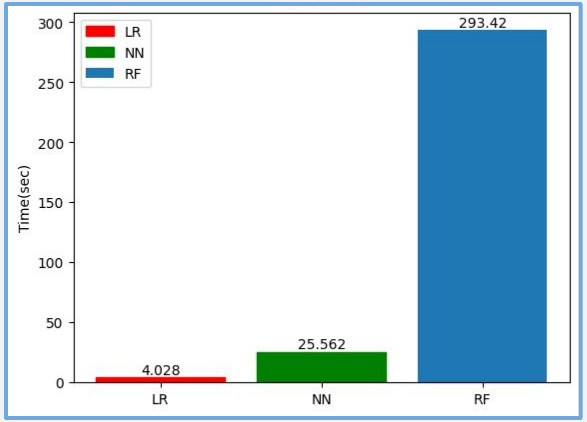
	MODEL	AUC	RECALL	PRECISION	F1-SCORE	ACCURACY
T	LR	0.662	0.920	0.662	0.733	0.996
T	NN(10,10)	0.68	0.917	0.68	0.751	0.997
	RF	0.528	0.937	0.527	0.545	0.973

Random Undersampling of Train-Set

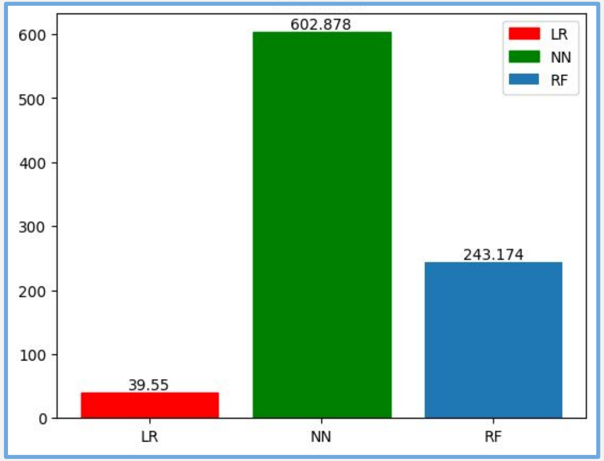
T-TEST (AUC)

MODELS	P-VALUE	RESULT (conf = 0.05)
NEURAL NETWORK - RANDOM FOREST *[IMBALANCED DS]*	0.0015	H_0 rejected **Random Forest Wins**
LOGISTIC REGR. - RANDOM FOREST *[OVERSAMPLED TRAIN SET]*	6.042 e-15	H_0 rejected **Random Forest Wins**
LOGISTIC REGR. - NEURAL NETWORK *[UNDERSAMPLED TRAIN SET]*	0.277	H_0 confirmed

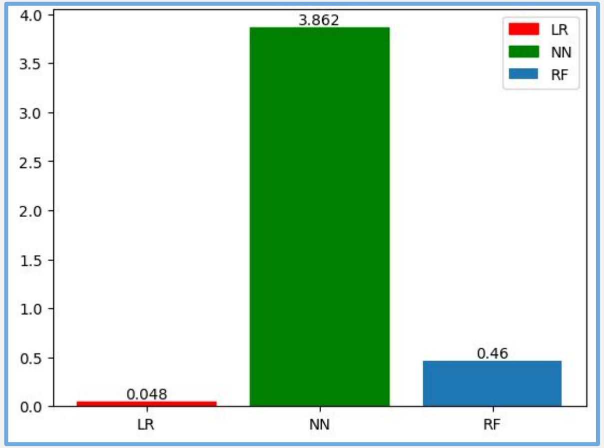
FIT TIME EVALUATION



Fit time without sampling train set



Fit time with oversampled train set



Fit time with undersampled train set

CONCLUSIONS

Evaluating AUC, Recall and Precision i can assume the **Random Forest** as better classifier for fraudulent detection transactions than Logistic Regression and Neural Network with this dataset.

IMPROVEMENTS

- Increase the number of transactions, above all fraudulent transactions, decreasing the imbalance ratio.

PYTHON PACKAGES

- **PANDAS** -> database handling
- **NUMPY** -> array handling
- **TIME** -> fit time
- **MATPLOTLIB** -> plots
- **IMBLEARN** -> sampling
- **SKLEARN** -> cross validation, models, performance metrics
- **STATISTICS** -> statistical measures
- **SCIPY** -> t-test

REFERENCES

- *Andrea Dal Pozzolo, Olivier Caelen, Reid A. Johnson and Gianluca Bontempi. Calibrating Probability with Undersampling for Unbalanced Classification. In Symposium on Computational Intelligence and Data Mining (CIDM), IEEE, 2015*
- *Dal Pozzolo, Andrea; Caelen, Olivier; Le Borgne, Yann-Ael; Waterschoot, Serge; Bontempi, Gianluca. Learned lessons in credit card fraud detection from a practitioner perspective, Expert systems with applications,41,10,4915-4928,2014, Pergamon*
- *Dal Pozzolo, Andrea; Boracchi, Giacomo; Caelen, Olivier; Alippi, Cesare; Bontempi, Gianluca. Credit card fraud detection: a realistic modeling and a novel learning strategy, IEEE transactions on neural networks and learning systems,29,8,3784-3797,2018,IEEE*
- *Dal Pozzolo, Andrea Adaptive Machine learning for credit card fraud detection ULB MLG PhD thesis (supervised by G. Bontempi)*
- *Carcillo, Fabrizio; Dal Pozzolo, Andrea; Le Borgne, Yann-Aël; Caelen, Olivier; Mazzer, Yannis; Bontempi, Gianluca. Scarff: a scalable framework for streaming credit card fraud detection with Spark, Information fusion,41, 182-194,2018,Elsevier*
- *Carcillo, Fabrizio; Le Borgne, Yann-Aël; Caelen, Olivier; Bontempi, Gianluca. Streaming active learning strategies for real-life credit card fraud detection: assessment and visualization, International Journal of Data Science and Analytics, 5,4,285-300,2018,Springer International Publishing*
- *Bertrand Lebichot, Yann-Aël Le Borgne, Liyun He, Frederic Oblé, Gianluca Bontempi Deep-Learning Domain Adaptation Techniques for Credit Cards Fraud Detection, INNSBDDL 2019: Recent Advances in Big Data and Deep Learning, pp 78-88, 2019*
- *Fabrizio Carcillo, Yann-Aël Le Borgne, Olivier Caelen, Frederic Oblé, Gianluca Bontempi Combining Unsupervised and Supervised Learning in Credit Card Fraud Detection Information Sciences, 2019*
- *Yann-Aël Le Borgne, Gianluca Bontempi Reproducible machine Learning for Credit Card Fraud Detection - Practical Handbook*
- *Bertrand Lebichot, Gianmarco Paldino, Wissam Siblini, Liyun He, Frederic Oblé, Gianluca Bontempi Incremental learning strategies for credit cards fraud detection, IInternational Journal of Data Science and Analytics*