# Credit Card Fraud Detection

## Content

In [50]:
```
pip install jupyterthemes
```

```
Collecting jupyterthemes
  Downloading jupyterthemes-0.20.0-py2.py3-none-any.whl (7.0 MB)
     |████████████████████████████████████████| 7.0 MB 4.2 MB/s eta 0:00:01
Requirement already satisfied: ipython>=5.4.1 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pa
ckages (from jupyterthemes) (7.8.0)
Collecting lesscpy>=0.11.2
  Downloading lesscpy-0.14.0-py2.py3-none-any.whl (46 kB)
     |████████████████████████████████████████| 46 kB 6.0 MB/s eta 0:00:011
Requirement already satisfied: notebook>=5.6.0 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-p
ackages (from jupyterthemes) (6.0.1)
Requirement already satisfied: matplotlib>=1.4.3 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site
-packages (from jupyterthemes) (3.2.1)
Requirement already satisfied: jupyter-core in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pack
ages (from jupyterthemes) (4.5.0)
Requirement already satisfied: backcall in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages
(from ipython>=5.4.1->jupyterthemes) (0.1.0)
Requirement already satisfied: pexpect; sys_platform != "win32" in /Library/Frameworks/Python.framework/Versions/3.7/li
b/python3.7/site-packages (from ipython>=5.4.1->jupyterthemes) (4.7.0)
Requirement already satisfied: traitlets>=4.2 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pa
ckages (from ipython>=5.4.1->jupyterthemes) (4.3.3)
Requirement already satisfied: setuptools>=18.5 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-
packages (from ipython>=5.4.1->jupyterthemes) (40.8.0)
Requirement already satisfied: pickleshare in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packa
ges (from ipython>=5.4.1->jupyterthemes) (0.7.5)
Requirement already satisfied: pygments in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages
(from ipython>=5.4.1->jupyterthemes) (2.4.2)
Requirement already satisfied: jedi>=0.10 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packag
es (from ipython>=5.4.1->jupyterthemes) (0.15.1)
Requirement already satisfied: decorator in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-package
s (from ipython>=5.4.1->jupyterthemes) (4.4.0)
Requirement already satisfied: prompt-toolkit<2.1.0,>=2.0.0 in /Library/Frameworks/Python.framework/Versions/3.7/lib/pyt
hon3.7/site-packages (from ipython>=5.4.1->jupyterthemes) (2.0.10)
Requirement already satisfied: appnope; sys_platform == "darwin" in /Library/Frameworks/Python.framework/Versions/3.7/li
b/python3.7/site-packages (from ipython>=5.4.1->jupyterthemes) (0.1.0)
Requirement already satisfied: six in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages (fro
m lesscpy>=0.11.2->jupyterthemes) (1.12.0)
Collecting ply
  Downloading ply-3.11-py2.py3-none-any.whl (49 kB)
     |████████████████████████████████████████| 49 kB 2.2 MB/s eta 0:00:01
Requirement already satisfied: ipykernel in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-package
s (from notebook>=5.6.0->jupyterthemes) (5.1.2)
Requirement already satisfied: nbformat in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages
(from notebook>=5.6.0->jupyterthemes) (4.4.0)
Requirement already satisfied: jupyter-client>=5.3.1 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/
site-packages (from notebook>=5.6.0->jupyterthemes) (5.3.3)
Requirement already satisfied: prometheus-client in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site
-packages (from notebook>=5.6.0->jupyterthemes) (0.7.1)
Requirement already satisfied: tornado>=5.0 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pack
ages (from notebook>=5.6.0->jupyterthemes) (6.0.3)
Requirement already satisfied: jinja2 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages
(from notebook>=5.6.0->jupyterthemes) (2.10.3)
Requirement already satisfied: nbconvert in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-package
s (from notebook>=5.6.0->jupyterthemes) (5.6.0)
Requirement already satisfied: ipython-genutils in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-
packages (from notebook>=5.6.0->jupyterthemes) (0.2.0)
Requirement already satisfied: pyzmq>=17 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-package
s (from notebook>=5.6.0->jupyterthemes) (18.1.0)
Requirement already satisfied: Send2Trash in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packag
es (from notebook>=5.6.0->jupyterthemes) (1.5.0)
Requirement already satisfied: terminado>=0.8.1 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-
packages (from notebook>=5.6.0->jupyterthemes) (0.8.2)
Requirement already satisfied: numpy>=1.11 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packa
ges (from matplotlib>=1.4.3->jupyterthemes) (1.18.2)
Requirement already satisfied: python-dateutil>=2.1 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/s
ite-packages (from matplotlib>=1.4.3->jupyterthemes) (2.8.0)
Requirement already satisfied: kiwisolver>=1.0.1 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site
-packages (from matplotlib>=1.4.3->jupyterthemes) (1.2.0)
Requirement already satisfied: cycler>=0.10 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pack
ages (from matplotlib>=1.4.3->jupyterthemes) (0.10.0)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /Library/Frameworks/Python.framework/Version
s/3.7/lib/python3.7/site-packages (from matplotlib>=1.4.3->jupyterthemes) (2.4.7)
```

```
Requirement already satisfied: ptyprocess>=0.5 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-p
ackages (from pexpect; sys_platform != "win32"->ipython>=5.4.1->jupyterthemes) (0.6.0)
Requirement already satisfied: parso>=0.5.0 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pack
ages (from jedi>=0.10->ipython>=5.4.1->jupyterthemes) (0.5.1)
Requirement already satisfied: wcwidth in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages
(from prompt-toolkit<2.1.0,>=2.0.0->ipython>=5.4.1->jupyterthemes) (0.1.7)
Requirement already satisfied: jsonschema!=2.5.0,>=2.4 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.
7/site-packages (from nbformat->notebook>=5.6.0->jupyterthemes) (3.0.2)
Requirement already satisfied: MarkupSafe>=0.23 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-
packages (from jinja2->notebook>=5.6.0->jupyterthemes) (1.1.1)
Requirement already satisfied: testpath in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages
(from nbconvert->notebook>=5.6.0->jupyterthemes) (0.4.2)
Requirement already satisfied: defusedxml in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packag
es (from nbconvert->notebook>=5.6.0->jupyterthemes) (0.6.0)
Requirement already satisfied: bleach in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages
(from nbconvert->notebook>=5.6.0->jupyterthemes) (3.1.0)
Requirement already satisfied: entrypoints>=0.2.2 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/sit
e-packages (from nbconvert->notebook>=5.6.0->jupyterthemes) (0.3)
Requirement already satisfied: mistune<2,>=0.8.1 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site
-packages (from nbconvert->notebook>=5.6.0->jupyterthemes) (0.8.4)
Requirement already satisfied: pandocfilters>=1.4.1 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/s
ite-packages (from nbconvert->notebook>=5.6.0->jupyterthemes) (1.4.2)
Requirement already satisfied: attrs>=17.4.0 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pac
kages (from jsonschema!=2.5.0,>=2.4->nbformat->notebook>=5.6.0->jupyterthemes) (19.2.0)
Requirement already satisfied: pyrsistent>=0.14.0 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/sit
e-packages (from jsonschema!=2.5.0,>=2.4->nbformat->notebook>=5.6.0->jupyterthemes) (0.15.4)
Requirement already satisfied: webencodings in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pack
ages (from bleach->nbconvert->notebook>=5.6.0->jupyterthemes) (0.5.1)
Could not build wheels for ipython, since package 'wheel' is not installed.
Could not build wheels for notebook, since package 'wheel' is not installed.
Could not build wheels for matplotlib, since package 'wheel' is not installed.
Could not build wheels for jupyter-core, since package 'wheel' is not installed.
Could not build wheels for backcall, since package 'wheel' is not installed.
Could not build wheels for pexpect, since package 'wheel' is not installed.
Could not build wheels for traitlets, since package 'wheel' is not installed.
Could not build wheels for setuptools, since package 'wheel' is not installed.
Could not build wheels for pickleshare, since package 'wheel' is not installed.
Could not build wheels for pygments, since package 'wheel' is not installed.
Could not build wheels for jedi, since package 'wheel' is not installed.
Could not build wheels for decorator, since package 'wheel' is not installed.
Could not build wheels for prompt-toolkit, since package 'wheel' is not installed.
Could not build wheels for appnope, since package 'wheel' is not installed.
Could not build wheels for six, since package 'wheel' is not installed.
Could not build wheels for ipykernel, since package 'wheel' is not installed.
Could not build wheels for nbformat, since package 'wheel' is not installed.
Could not build wheels for jupyter-client, since package 'wheel' is not installed.
Could not build wheels for prometheus-client, since package 'wheel' is not installed.
Could not build wheels for tornado, since package 'wheel' is not installed.
Could not build wheels for jinja2, since package 'wheel' is not installed.
Could not build wheels for nbconvert, since package 'wheel' is not installed.
Could not build wheels for ipython-genutils, since package 'wheel' is not installed.
Could not build wheels for pyzmq, since package 'wheel' is not installed.
Could not build wheels for Send2Trash, since package 'wheel' is not installed.
Could not build wheels for terminado, since package 'wheel' is not installed.
Could not build wheels for numpy, since package 'wheel' is not installed.
Could not build wheels for python-dateutil, since package 'wheel' is not installed.
Could not build wheels for kiwisolver, since package 'wheel' is not installed.
Could not build wheels for cycler, since package 'wheel' is not installed.
Could not build wheels for pyparsing, since package 'wheel' is not installed.
Could not build wheels for ptyprocess, since package 'wheel' is not installed.
Could not build wheels for parso, since package 'wheel' is not installed.
Could not build wheels for wcwidth, since package 'wheel' is not installed.
Could not build wheels for jsonschema, since package 'wheel' is not installed.
Could not build wheels for MarkupSafe, since package 'wheel' is not installed.
Could not build wheels for testpath, since package 'wheel' is not installed.
Could not build wheels for defusedxml, since package 'wheel' is not installed.
Could not build wheels for bleach, since package 'wheel' is not installed.
Could not build wheels for entrypoints, since package 'wheel' is not installed.
Could not build wheels for mistune, since package 'wheel' is not installed.
Could not build wheels for pandocfilters, since package 'wheel' is not installed.
Could not build wheels for attrs, since package 'wheel' is not installed.
Could not build wheels for pyrsistent, since package 'wheel' is not installed.
```

```
Could not build wheels for webencodings, since package 'wheel' is not installed.
Installing collected packages: ply, lesscpy, jupyterthemes
Successfully installed jupyterthemes-0.20.0 lesscpy-0.14.0 ply-3.11
Note: you may need to restart the kernel to use updated packages.
```

In [2]:
```
pip install wheel
```

```
Collecting wheel
  Downloading wheel-0.34.2-py2.py3-none-any.whl (26 kB)
Installing collected packages: wheel
Successfully installed wheel-0.34.2
Note: you may need to restart the kernel to use updated packages.
```

In [5]:
```
!jt -t oceans16
```

# Introduction

The datasets contains transactions made by credit cards in **September 2013** by european cardholders. This dataset presents transactions that occurred in two days, where we have **492 frauds** out of **284,807 transactions**. **The dataset is highly imbalanced**, the **positive class (frauds)** account for **0.172%** of all transactions.

It is important that credit card companies are able to recognize fraudulent credit card transactions so that customers are not charged for items that they did not purchase. The goal for this analysis is to predict credit card fraud in the transactional data. As far as I am concerned, it will have great value in preventing financial crimes.

**Goal:** Determine the Classifiers we are going to use and decide which one has a higher accuracy.

**PCA:** It contains only numerical input variables which are the result of a **PCA transformation**.

Due to confidentiality issues, there are not provided the original features and more background information about the data.

- Features **V1**, **V2**, ... **V28** are the **principal components** obtained with **PCA**;
- The only features which have not been transformed with PCA are **Time** and **Amount**. Feature **Time** contains the seconds elapsed between each transaction and the first transaction in the dataset. The feature **Amount** is the transaction Amount, this feature can be used for example-dependant cost-senstive learning.
- Feature **Class** is the response variable and it takes value **1** in case of fraud and **0** otherwise.

Boosting is based on weak learners (high bias, low variance)/ Random Forest (low bias, high variance).

GBMs are more sensitive to overfitting if the data is noisy. Training generally takes longer because of the fact that trees are built sequentially. GBMs are harder to tune than RF. There are typically three parameters: number of trees, depth of trees and learning rate, and each tree built is generally shallow

The most prominent application of random forest is multi-class object detection in large-scale real-world computer vision problems. RF methods can handle a large amount of training data efficiently and are inherently suited for multi-class problems.

In [48]:
```
pip install --upgrade pip
```

```
Requirement already up-to-date: pip in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages (20.1)
Note: you may need to restart the kernel to use updated packages.
```

In [2]:
```
pip install seaborn
```

```
Requirement already satisfied: seaborn in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages
(0.10.1)
Requirement already satisfied: scipy>=1.0.1 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pack
ages (from seaborn) (1.4.1)
Requirement already satisfied: pandas>=0.22.0 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pa
ckages (from seaborn) (1.0.3)
Requirement already satisfied: numpy>=1.13.3 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pac
kages (from seaborn) (1.18.2)
Requirement already satisfied: matplotlib>=2.1.2 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site
-packages (from seaborn) (3.2.1)
Requirement already satisfied: python-dateutil>=2.6.1 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.
7/site-packages (from pandas>=0.22.0->seaborn) (2.8.0)
Requirement already satisfied: pytz>=2017.2 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pack
ages (from pandas>=0.22.0->seaborn) (2019.3)
Requirement already satisfied: cycler>=0.10 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pack
ages (from matplotlib>=2.1.2->seaborn) (0.10.0)
Requirement already satisfied: kiwisolver>=1.0.1 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site
-packages (from matplotlib>=2.1.2->seaborn) (1.2.0)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /Library/Frameworks/Python.framework/Version
s/3.7/lib/python3.7/site-packages (from matplotlib>=2.1.2->seaborn) (2.4.7)
Requirement already satisfied: six>=1.5 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages
(from python-dateutil>=2.6.1->pandas>=0.22.0->seaborn) (1.12.0)
Could not build wheels for seaborn, since package 'wheel' is not installed.
Could not build wheels for scipy, since package 'wheel' is not installed.
Could not build wheels for pandas, since package 'wheel' is not installed.
Could not build wheels for numpy, since package 'wheel' is not installed.
Could not build wheels for matplotlib, since package 'wheel' is not installed.
Could not build wheels for python-dateutil, since package 'wheel' is not installed.
Could not build wheels for pytz, since package 'wheel' is not installed.
Could not build wheels for cycler, since package 'wheel' is not installed.
Could not build wheels for kiwisolver, since package 'wheel' is not installed.
Could not build wheels for pyparsing, since package 'wheel' is not installed.
Could not build wheels for six, since package 'wheel' is not installed.
Note: you may need to restart the kernel to use updated packages.
```

In [3]:
```
pip install plotly
```

```
Requirement already satisfied: plotly in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages
(4.6.0)
Requirement already satisfied: six in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages (fro
m plotly) (1.12.0)
Requirement already satisfied: retrying>=1.3.3 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-p
ackages (from plotly) (1.3.3)
Could not build wheels for plotly, since package 'wheel' is not installed.
Could not build wheels for six, since package 'wheel' is not installed.
Could not build wheels for retrying, since package 'wheel' is not installed.
Note: you may need to restart the kernel to use updated packages.
```

# Load packages

In [4]:
```
pip install sklearn
```

```
Requirement already satisfied: sklearn in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages
(0.0)
Requirement already satisfied: scikit-learn in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pack
ages (from sklearn) (0.22.2.post1)
Requirement already satisfied: scipy>=0.17.0 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pac
kages (from scikit-learn->sklearn) (1.4.1)
Requirement already satisfied: joblib>=0.11 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pack
ages (from scikit-learn->sklearn) (0.14.1)
Requirement already satisfied: numpy>=1.11.0 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pac
kages (from scikit-learn->sklearn) (1.18.2)
Could not build wheels for sklearn, since package 'wheel' is not installed.
Could not build wheels for scikit-learn, since package 'wheel' is not installed.
Could not build wheels for scipy, since package 'wheel' is not installed.
Could not build wheels for joblib, since package 'wheel' is not installed.
Could not build wheels for numpy, since package 'wheel' is not installed.
Note: you may need to restart the kernel to use updated packages.
```

In [5]:
```
pip install catboost
```

```
Requirement already satisfied: catboost in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages
(0.23)
Requirement already satisfied: plotly in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages
(from catboost) (4.6.0)
Requirement already satisfied: numpy>=1.16.0 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pac
kages (from catboost) (1.18.2)
Requirement already satisfied: scipy in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages (f
rom catboost) (1.4.1)
Requirement already satisfied: six in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages (fro
m catboost) (1.12.0)
Requirement already satisfied: pandas>=0.24.0 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pa
ckages (from catboost) (1.0.3)
Requirement already satisfied: matplotlib in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packag
es (from catboost) (3.2.1)
Requirement already satisfied: graphviz in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages
(from catboost) (0.14)
Requirement already satisfied: retrying>=1.3.3 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-p
ackages (from plotly->catboost) (1.3.3)
Requirement already satisfied: python-dateutil>=2.6.1 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.
7/site-packages (from pandas>=0.24.0->catboost) (2.8.0)
Requirement already satisfied: pytz>=2017.2 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pack
ages (from pandas>=0.24.0->catboost) (2019.3)
Requirement already satisfied: kiwisolver>=1.0.1 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site
-packages (from matplotlib->catboost) (1.2.0)
Requirement already satisfied: cycler>=0.10 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pack
ages (from matplotlib->catboost) (0.10.0)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /Library/Frameworks/Python.framework/Version
s/3.7/lib/python3.7/site-packages (from matplotlib->catboost) (2.4.7)
Could not build wheels for catboost, since package 'wheel' is not installed.
Could not build wheels for plotly, since package 'wheel' is not installed.
Could not build wheels for numpy, since package 'wheel' is not installed.
Could not build wheels for scipy, since package 'wheel' is not installed.
Could not build wheels for six, since package 'wheel' is not installed.
Could not build wheels for pandas, since package 'wheel' is not installed.
Could not build wheels for matplotlib, since package 'wheel' is not installed.
Could not build wheels for graphviz, since package 'wheel' is not installed.
Could not build wheels for retrying, since package 'wheel' is not installed.
Could not build wheels for python-dateutil, since package 'wheel' is not installed.
Could not build wheels for pytz, since package 'wheel' is not installed.
Could not build wheels for kiwisolver, since package 'wheel' is not installed.
Could not build wheels for cycler, since package 'wheel' is not installed.
Could not build wheels for pyparsing, since package 'wheel' is not installed.
Note: you may need to restart the kernel to use updated packages.
```

In [6]:
```
pip install lightgbm
```

```
Requirement already satisfied: lightgbm in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages
(2.3.1)
Requirement already satisfied: scipy in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages (f
rom lightgbm) (1.4.1)
Requirement already satisfied: scikit-learn in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pack
ages (from lightgbm) (0.22.2.post1)
Requirement already satisfied: numpy in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages (f
rom lightgbm) (1.18.2)
Requirement already satisfied: joblib>=0.11 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pack
ages (from scikit-learn->lightgbm) (0.14.1)
Could not build wheels for lightgbm, since package 'wheel' is not installed.
Could not build wheels for scipy, since package 'wheel' is not installed.
Could not build wheels for scikit-learn, since package 'wheel' is not installed.
Could not build wheels for numpy, since package 'wheel' is not installed.
Could not build wheels for joblib, since package 'wheel' is not installed.
Note: you may need to restart the kernel to use updated packages.
```

In [50]:
```
pip install -U setuptools
```

```
Collecting setuptools
  Downloading setuptools-46.1.3-py3-none-any.whl (582 kB)
     |████████████████████████████████| 582 kB 1.7 MB/s eta 0:00:01
Installing collected packages: setuptools
  Attempting uninstall: setuptools
    Found existing installation: setuptools 40.8.0
    Uninstalling setuptools-40.8.0:
      Successfully uninstalled setuptools-40.8.0
Successfully installed setuptools-46.1.3
Note: you may need to restart the kernel to use updated packages.
```

In [51]:
```
pip install xgboost
```

```
Collecting xgboost
  Using cached xgboost-1.0.2.tar.gz (821 kB)
    ERROR: Command errored out with exit status 1:
     command: /Library/Frameworks/Python.framework/Versions/3.7/bin/python3 -c 'import sys, setuptools, tokenize; sys.ar
gv[0] = '"'"'/private/var/folders/ps/mfrj3db90v3g79pw29r2pncw0000gn/T/pip-install-gz077kfk/xgboost/setup.py'"'"'; __file
__='"'"'/private/var/folders/ps/mfrj3db90v3g79pw29r2pncw0000gn/T/pip-install-gz077kfk/xgboost/setup.py'"'"';f=getattr(to
kenize, '"'"'open'"'"', open)(__file__);code=f.read().replace('"'"'\r\n'"'"', '"'"'\n'"'"');f.close();exec(compile(code,
__file__, '"'"'exec'"'"'))' egg_info --egg-base /private/var/folders/ps/mfrj3db90v3g79pw29r2pncw0000gn/T/pip-pip-egg-inf
o-vghta4cq
         cwd: /private/var/folders/ps/mfrj3db90v3g79pw29r2pncw0000gn/T/pip-install-gz077kfk/xgboost/
    Complete output (27 lines):
    ++ pwd
    + oldpath=/private/var/folders/ps/mfrj3db90v3g79pw29r2pncw0000gn/T/pip-install-gz077kfk/xgboost
    + cd ./xgboost/
    + mkdir -p build
    + cd build
    + cmake ..
    ./xgboost/build-python.sh: line 21: cmake: command not found
    + echo ----------------------------
    ----------------------------
    + echo 'Building multi-thread xgboost failed'
    Building multi-thread xgboost failed
    + echo 'Start to build single-thread xgboost'
    Start to build single-thread xgboost
    + cmake .. -DUSE_OPENMP=0
    ./xgboost/build-python.sh: line 27: cmake: command not found
    Traceback (most recent call last):
      File "<string>", line 1, in <module>
      File "/private/var/folders/ps/mfrj3db90v3g79pw29r2pncw0000gn/T/pip-install-gz077kfk/xgboost/setup.py", line 42, in
<module>
        LIB_PATH = libpath['find_lib_path']()
      File "/private/var/folders/ps/mfrj3db90v3g79pw29r2pncw0000gn/T/pip-install-gz077kfk/xgboost/xgboost/libpath.py", l
ine 50, in find_lib_path
        'List of candidates:\n' + ('\n'.join(dll_path)))
    XGBoostLibraryNotFound: Cannot find XGBoost Library in the candidate path, did you install compilers and run build.s
h in root path?
    List of candidates:
    /private/var/folders/ps/mfrj3db90v3g79pw29r2pncw0000gn/T/pip-install-gz077kfk/xgboost/xgboost/libxgboost.dylib
    /private/var/folders/ps/mfrj3db90v3g79pw29r2pncw0000gn/T/pip-install-gz077kfk/xgboost/xgboost/../../lib/libxgboost.d
ylib
    /private/var/folders/ps/mfrj3db90v3g79pw29r2pncw0000gn/T/pip-install-gz077kfk/xgboost/xgboost/./lib/libxgboost.dylib
    /Library/Frameworks/Python.framework/Versions/3.7/xgboost/libxgboost.dylib
    ----------------------------------------
ERROR: Command errored out with exit status 1: python setup.py egg_info Check the logs for full command output.
Note: you may need to restart the kernel to use updated packages.
```

In [8]:
```
pip install --no-binary :all: lightgbm
```

```
Requirement already satisfied: lightgbm in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages
(2.3.1)
Requirement already satisfied: scikit-learn in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pack
ages (from lightgbm) (0.22.2.post1)
Requirement already satisfied: numpy in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages (f
rom lightgbm) (1.18.2)
Requirement already satisfied: scipy in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages (f
rom lightgbm) (1.4.1)
Requirement already satisfied: joblib>=0.11 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pack
ages (from scikit-learn->lightgbm) (0.14.1)
Could not build wheels for lightgbm, since package 'wheel' is not installed.
Could not build wheels for scikit-learn, since package 'wheel' is not installed.
Could not build wheels for numpy, since package 'wheel' is not installed.
Could not build wheels for scipy, since package 'wheel' is not installed.
Could not build wheels for joblib, since package 'wheel' is not installed.
Note: you may need to restart the kernel to use updated packages.
```

In [9]:
```
brew install cmake
brew install gcc --without-multilib
git clone --recursive https://github.com/Microsoft/LightGBM ; cd LightGBM
mkdir build ; cd build
cmake ..
make -j
```

```
  File "<ipython-input-9-4070e04e51b5>", line 1
    brew install cmake
               ^
SyntaxError: invalid syntax
```

In [7]:
```
import pandas as pd
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
import plotly.graph_objs as go
import plotly.figure_factory as ff
from plotly import tools
from plotly.offline import download_plotlyjs, init_notebook_mode, plot, iplot
init_notebook_mode(connected=True)


import gc
from datetime import datetime
from sklearn.model_selection import train_test_split
from sklearn.model_selection import KFold
from sklearn.metrics import roc_auc_score
from sklearn.ensemble import RandomForestClassifier
from sklearn.ensemble import AdaBoostClassifier
from catboost import CatBoostClassifier
from sklearn import svm


import lightgbm as lgb
from lightgbm import LGBMClassifier


import xgboost as xgb

pd.set_option('display.max_columns', 100)


RFC_METRIC = 'gini'  #metric used for RandomForrestClassifier
NUM_ESTIMATORS = 100 #number of estimators used for RandomForrestClassifier
NO_JOBS = 4 #number of parallel jobs used for RandomForrestClassifier


#TRAIN/VALIDATION/TEST SPLIT
#VALIDATION
VALID_SIZE = 0.20 # simple validation using train_test_split
TEST_SIZE = 0.20 # test size using_train_test_split

#CROSS-VALIDATION
NUMBER_KFOLDS = 5 #number of KFolds for cross-validation
```

```
RANDOM_STATE = 2020

MAX_ROUNDS = 1000 #lgb iterations
EARLY_STOP = 50 #lgb early stop
OPT_ROUNDS = 1000  #To be adjusted based on best validation rounds
VERBOSE_EVAL = 50 #Print out metric result

IS_LOCAL = False

import os

if(IS_LOCAL):
    PATH="../input/credit-card-fraud-detection"
else:
    PATH="../input"
print(os.listdir(PATH))
```

```
---------------------------------------------------------------------------
OSError                                   Traceback (most recent call last)
<ipython-input-7-28d3e95be5e6> in <module>
     21 from catboost import CatBoostClassifier
     22 from sklearn import svm
---> 23 import lightgbm as lgb
     24 from lightgbm import LGBMClassifier
     25 import xgboost as xgb

/Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages/lightgbm/__init__.py in <module>
      6 from __future__ import absolute_import
      7
----> 8 from .basic import Booster, Dataset
      9 from .callback import (early_stopping, print_evaluation, record_evaluation,
     10                        reset_parameter)

/Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages/lightgbm/basic.py in <module>
     31
     32
---> 33 _LIB = _load_lib()
     34
     35

/Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages/lightgbm/basic.py in _load_lib()
     26     if len(lib_path) == 0:
     27         return None
---> 28     lib = ctypes.cdll.LoadLibrary(lib_path[0])
     29     lib.LGBM_GetLastError.restype = ctypes.c_char_p
     30     return lib

/Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/ctypes/__init__.py in LoadLibrary(self, name)
    440
    441     def LoadLibrary(self, name):
--> 442         return self._dlltype(name)
    443
    444 cdll = LibraryLoader(CDLL)

/Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/ctypes/__init__.py in __init__(self, name, mode, handle,
use_errno, use_last_error)
    362
    363         if handle is None:
--> 364             self._handle = _dlopen(self._name, mode)
    365         else:
    366             self._handle = handle

OSError: dlopen(/Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages/lightgbm/lib_lightgbm.so,
 6): Library not loaded: /usr/local/opt/libomp/lib/libomp.dylib
  Referenced from: /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages/lightgbm/lib_lightgbm.s
o
  Reason: image not found
```

# Read the data

In [8]:
```
data_df = pd.read_csv('creditcard.csv')
print(data_df)
```

```
             Time         V1         V2        V3        V4        V5  \
0             0.0  -1.359807  -0.072781  2.536347  1.378155 -0.338321
1             0.0   1.191857   0.266151  0.166480  0.448154  0.060018
2             1.0  -1.358354  -1.340163  1.773209  0.379780 -0.503198
3             1.0  -0.966272  -0.185226  1.792993 -0.863291 -0.010309
4             2.0  -1.158233   0.877737  1.548718  0.403034 -0.407193
...           ...        ...        ...       ...       ...       ...
284802  172786.0 -11.881118  10.071785 -9.834783 -2.066656 -5.364473
284803  172787.0  -0.732789  -0.055080  2.035030 -0.738589  0.868229
284804  172788.0   1.919565  -0.301254 -3.249640 -0.557828  2.630515
284805  172788.0  -0.240440   0.530483  0.702510  0.689799 -0.377961
284806  172792.0  -0.533413  -0.189733  0.703337 -0.506271 -0.012546

                V6        V7        V8        V9  ...       V21       V22  \
0         0.462388  0.239599  0.098698  0.363787  ... -0.018307  0.277838
1        -0.082361 -0.078803  0.085102 -0.255425  ... -0.225775 -0.638672
2         1.800499  0.791461  0.247676 -1.514654  ...  0.247998  0.771679
3         1.247203  0.237609  0.377436 -1.387024  ... -0.108300  0.005274
4         0.095921  0.592941 -0.270533  0.817739  ... -0.009431  0.798278
...            ...       ...       ...       ...  ...       ...       ...
284802 -2.606837 -4.918215  7.305334  1.914428  ...  0.213454  0.111864
284803  1.058415  0.024330  0.294869  0.584800  ...  0.214205  0.924384
284804  3.031260 -0.296827  0.708417  0.432454  ...  0.232045  0.578229
284805  0.623708 -0.686180  0.679145  0.392087  ...  0.265245  0.800049
284806 -0.649617  1.577006 -0.414650  0.486180  ...  0.261057  0.643078

               V23       V24       V25       V26       V27       V28  Amount  \
0        -0.110474  0.066928  0.128539 -0.189115  0.133558 -0.021053  149.62
1         0.101288 -0.339846  0.167170  0.125895 -0.008983  0.014724    2.69
2         0.909412 -0.689281 -0.327642 -0.139097 -0.055353 -0.059752  378.66
3        -0.190321 -1.175575  0.647376 -0.221929  0.062723  0.061458  123.50
4        -0.137458  0.141267 -0.206010  0.502292  0.219422  0.215153   69.99
...            ...       ...       ...       ...       ...       ...     ...
284802   1.014480 -0.509348  1.436807  0.250034  0.943651  0.823731    0.77
284803   0.012463 -1.016226 -0.606624 -0.395255  0.068472 -0.053527   24.79
284804  -0.037501  0.640134  0.265745 -0.087371  0.004455 -0.026561   67.88
284805  -0.163298  0.123205 -0.569159  0.546668  0.108821  0.104533   10.00
284806   0.376777  0.008797 -0.473649 -0.818267 -0.002415  0.013649  217.00

        Class
0           0
1           0
2           0
3           0
4           0
...       ...
284802      0
284803      0
284804      0
284805      0
284806      0

[284807 rows x 31 columns]
```

# Check the data

In [9]:
```
print("Credit Card Fraud Detection data -  rows:",data_df.shape[0]," columns:", data_df.shape[1])
```

```
Credit Card Fraud Detection data -  rows: 284807  columns: 31
```

# Glimpse the data

We start by looking to the data features (first 5 rows).

In [10]:
```
data_df.head()
```

| | Time | V1 | V2 | V3 | V4 | V5 | V6 | V7 | V8 | V9 | ... | V21 | V2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| 1 | 0.0 | 1.191857 | 0.266151 | 0.166480 | 0.448154 | 0.060018 | −0.082361 | −0.078803 | 0.085102 | −0.255425 | ... | −0.225775 | −0.63867 |
| 2 | 1.0 | −1.358354 | −1.340163 | 1.773209 | 0.379780 | −0.503198 | 1.800499 | 0.791461 | 0.247676 | −1.514654 | ... | 0.247998 | 0.771679 |
| 3 | 1.0 | −0.966272 | −0.185226 | 1.792993 | −0.863291 | −0.010309 | 1.247203 | 0.237609 | 0.377436 | −1.387024 | ... | −0.108300 | 0.005274 |
| 4 | 2.0 | −1.158233 | 0.877737 | 1.548718 | 0.403034 | −0.407193 | 0.095921 | 0.592941 | −0.270533 | 0.817739 | ... | −0.009431 | 0.798278 |

5 rows × 31 columns

Let's look into more details to the data.

In [11]:
```
data_df.describe()
```

| | Time | V1 | V2 | V3 | V4 | V5 | V6 | V7 | |
|---|---|---|---|---|---|---|---|---|---|
| count | 284807.000000 | 2.848070e+05 | 2.848070e+05 | 2.848070e+05 | 2.848070e+05 | 2.848070e+05 | 2.848070e+05 | 2.848070e+05 | 2.848 |
| mean | 94813.859575 | 1.165980e−15 | 3.416908e−16 | −1.373150e−15 | 2.086869e−15 | 9.604066e−16 | 1.490107e−15 | −5.556467e−16 | 1.1775 |
| std | 47488.145955 | 1.958696e+00 | 1.651309e+00 | 1.516255e+00 | 1.415869e+00 | 1.380247e+00 | 1.332271e+00 | 1.237094e+00 | 1.1943 |
| min | 0.000000 | −5.640751e+01 | −7.271573e+01 | −4.832559e+01 | −5.683171e+00 | −1.137433e+02 | −2.616051e+01 | −4.355724e+01 | −7.32 |
| 25% | 54201.500000 | −9.203734e−01 | −5.985499e−01 | −8.903648e−01 | −8.486401e−01 | −6.915971e−01 | −7.682956e−01 | −5.540759e−01 | −2.08 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 | 2.235 |
| 75% | 139320.500000 | 1.315642e+00 | 8.037239e−01 | 1.027196e+00 | 7.433413e−01 | 6.119264e−01 | 3.985649e−01 | 5.704361e−01 | 3.273 |
| max | 172792.000000 | 2.454930e+00 | 2.205773e+01 | 9.382558e+00 | 1.687534e+01 | 3.480167e+01 | 7.330163e+01 | 1.205895e+02 | 2.000 |

8 rows × 31 columns

Looking to the **Time** feature, we can confirm that the data contains **284,807** transactions, during 2 consecutive days (or **172792** seconds).

# Check missing data

Let's check if there is any missing data.

In [12]:
```
total = data_df.isnull().sum().sort_values(ascending = False)
percent = (data_df.isnull().sum()/data_df.isnull().count()*100).sort_values(ascending = Fa
lse)
pd.concat([total, percent], axis=1, keys=['Total', 'Percent']).transpose()
```

| | Class | V14 | V1 | V2 | V3 | V4 | V5 | V6 | V7 | V8 | ... | V20 | V21 | V22 | V23 | V24 | V25 | V26 | V27 | V28 | Time |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Total | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| Percent | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |

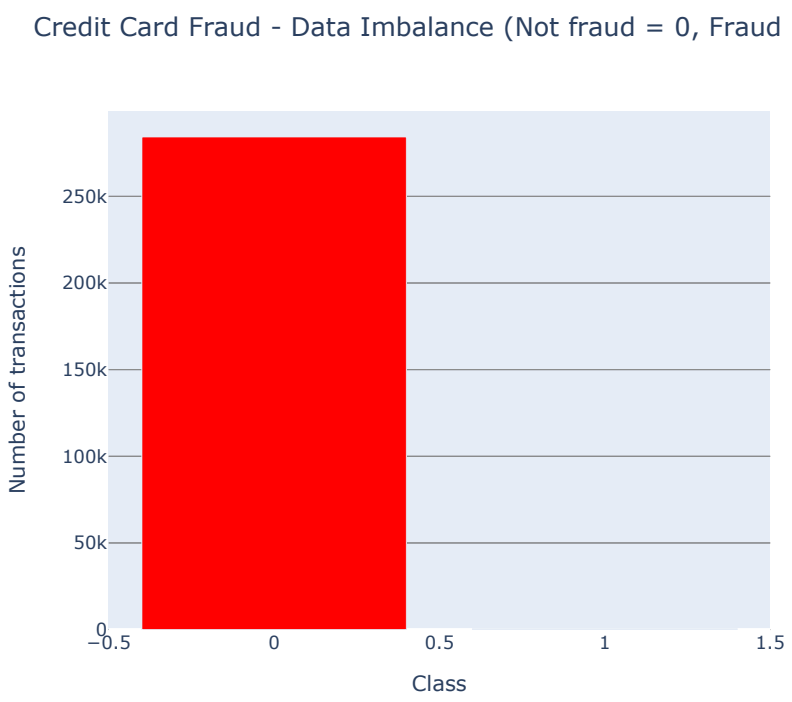
2 rows × 31 columns

There is no missing data in the entire dataset.

# Imbalance Dataset

Let's check data imbalance with respect with *target* value, i.e. **Class**.

In [13]:
```
temp = data_df["Class"].value_counts()
df = pd.DataFrame({'Class': temp.index,'values': temp.values})

trace = go.Bar(
    x = df['Class'],y = df['values'],
    name="Credit Card Fraud - Data Imbalance(Not fraud = 0, Fraud = 1)",
    marker=dict(color="Red"),
    text=df['values']
)
data = [trace]
layout = dict(title = 'Credit Card Fraud - Data Imbalance (Not fraud = 0, Fraud = 1)',
          xaxis = dict(title = 'Class', showticklabels=True),
          yaxis = dict(title = 'Number of transactions'),
          hovermode = 'closest',width=600
         )
fig = dict(data=data, layout=layout)
iplot(fig, filename='class')
```

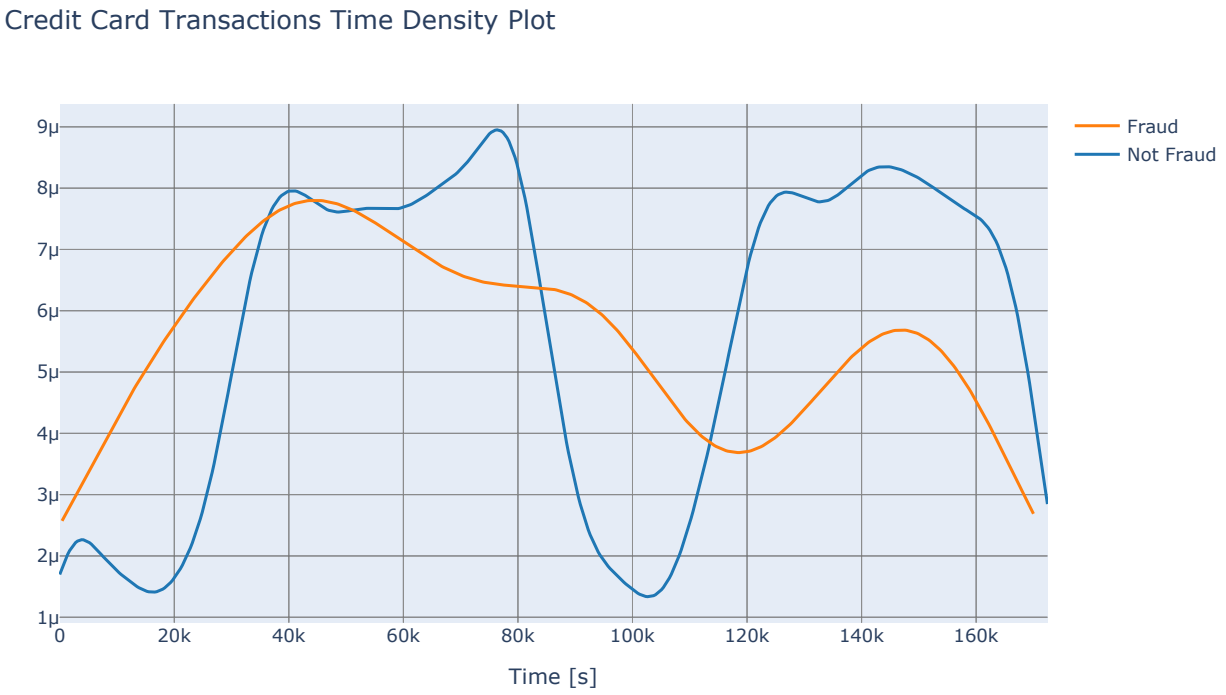



Only **492** (or **0.172%**) of transaction are fraudulent. That means the data is highly imbalanced with respect with target variable **Class**.

# Data Exploration

# Transactions in time

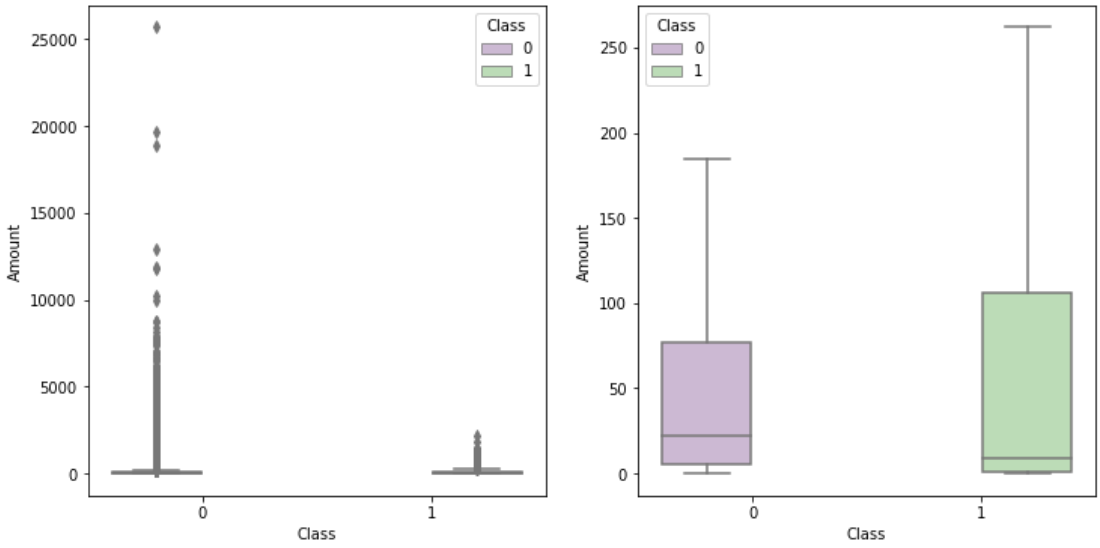
In [14]:
```
class_0 = data_df.loc[data_df['Class'] == 0]["Time"]
class_1 = data_df.loc[data_df['Class'] == 1]["Time"]
#plt.figure(figsize = (14,4))
#plt.title('Credit Card Transactions Time Density Plot')
#sns.set_color_codes("pastel")
#sns.distplot(class_0,kde=True,bins=480)
#sns.distplot(class_1,kde=True,bins=480)
#plt.show()
hist_data = [class_0, class_1]
group_labels = ['Not Fraud', 'Fraud']

fig = ff.create_distplot(hist_data, group_labels, show_hist=False, show_rug=False)
fig['layout'].update(title='Credit Card Transactions Time Density Plot', xaxis=dict(title=
'Time [s]'))
iplot(fig, filename='dist_only')
```




Fraudulent transactions have a distribution more even than valid transactions – are equaly distributed in time, including the low real transaction times, during night in Europe timezone.

# Transactions amount

In [15]:
```
fig, (ax1, ax2) = plt.subplots(ncols=2, figsize=(12,6))
s = sns.boxplot(ax = ax1, x="Class", y="Amount", hue="Class",data=data_df, palette="PRGn",
showfliers=True)
s = sns.boxplot(ax = ax2, x="Class", y="Amount", hue="Class",data=data_df, palette="PRGn",
showfliers=False)
plt.show();
```




In [16]:
```
tmp = data_df[['Amount','Class']].copy()
class_0 = tmp.loc[tmp['Class'] == 0]['Amount']
class_1 = tmp.loc[tmp['Class'] == 1]['Amount']
class_0.describe()
```

```
count    284315.000000
mean         88.291022
std         250.105092
min           0.000000
25%           5.650000
50%          22.000000
75%          77.050000
max       25691.160000
Name: Amount, dtype: float64
```

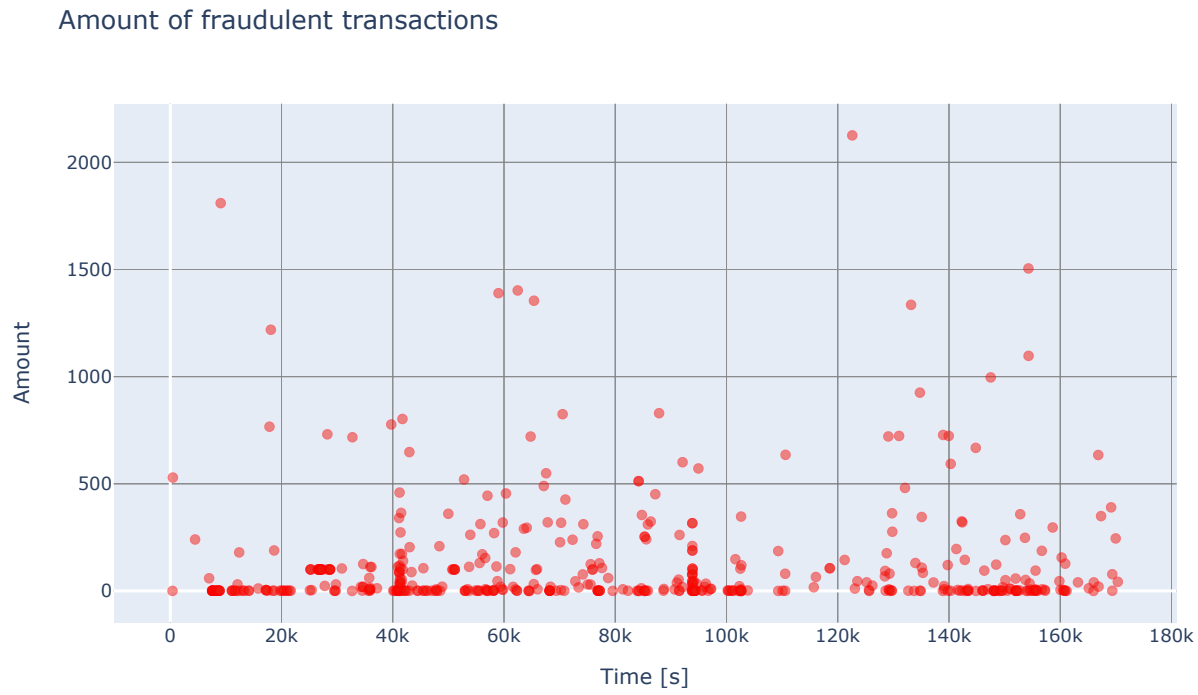
In [17]:
```
class_1.describe()
```

```
count     492.000000
mean      122.211321
std       256.683288
min         0.000000
25%         1.000000
50%         9.250000
75%       105.890000
max      2125.870000
Name: Amount, dtype: float64
```

The real transaction have a larger mean value, larger Q1, smaller Q3 and Q4 and larger outliers; fraudulent transactions have a smaller Q1 and mean, larger Q4 and smaller outliers.

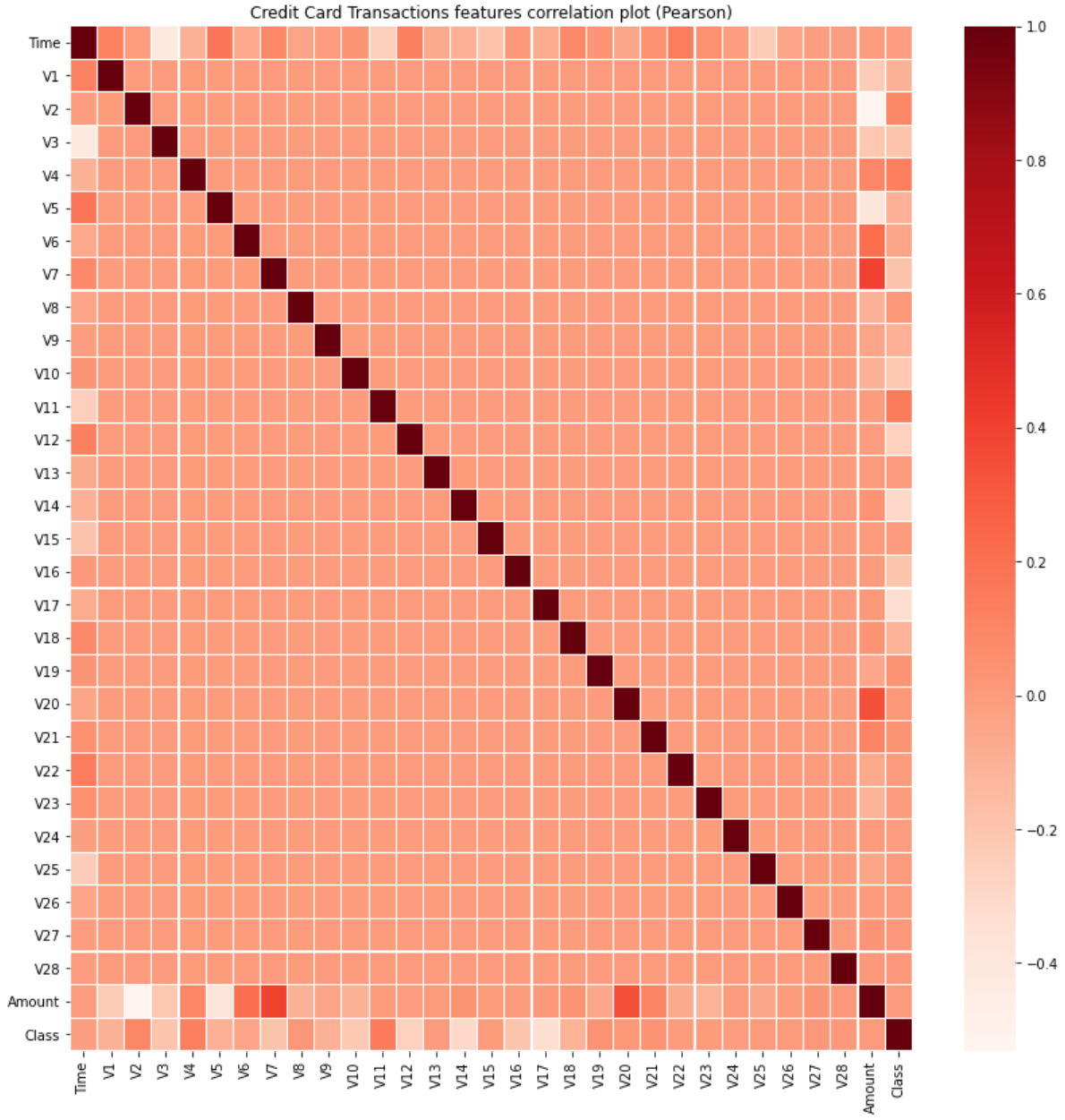
Let's plot the fraudulent transactions (amount) against time. The time is shown is seconds from the start of the time period (totaly 48h, over 2 days).

In [18]:
```
fraud = data_df.loc[data_df['Class'] == 1]

trace = go.Scatter(
    x = fraud['Time'],y = fraud['Amount'],
    name="Amount",
     marker=dict(
                color='rgb(238,23,11)',
                line=dict(
                    color='red',
                    width=1),
                opacity=0.5,
            ),
    text= fraud['Amount'],
    mode = "markers"
)
data = [trace]
layout = dict(title = 'Amount of fraudulent transactions',
          xaxis = dict(title = 'Time [s]', showticklabels=True),
          yaxis = dict(title = 'Amount'),
          hovermode='closest'
         )
fig = dict(data=data, layout=layout)
iplot(fig, filename='fraud-amount')
```




# Features Correlation

In [19]:
```
plt.figure(figsize = (14,14))
plt.title('Credit Card Transactions features correlation plot (Pearson)')
corr = data_df.corr()
sns.heatmap(corr,xticklabels=corr.columns,yticklabels=corr.columns,linewidths=.1,cmap="Reds")
plt.show()
```

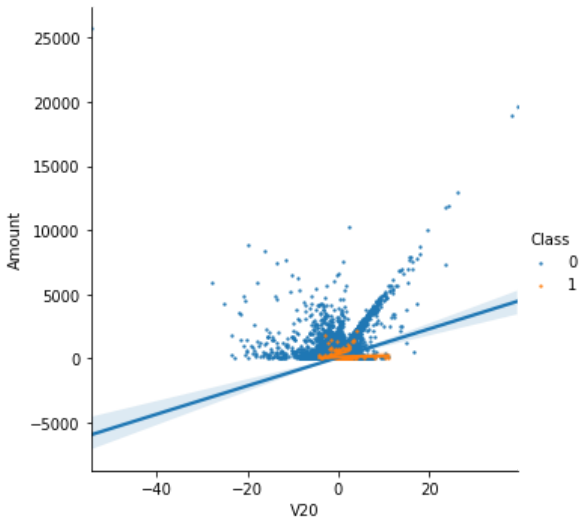


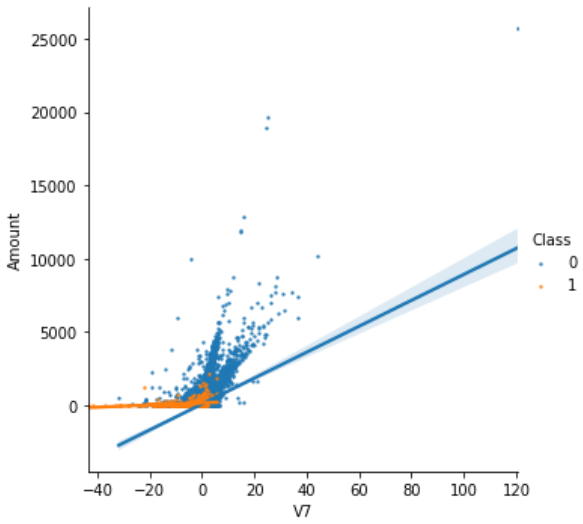

As expected, there is no notable correlation between features **V1–V28**. There are certain correlations between some of these features and **Time** (inverse correlation with **V3**) and **Amount** (direct correlation with **V7** and **V20**, inverse correlation with **V1** and **V5**).

Let's plot the correlated and inverse correlated values on the same graph.

Let's start with the direct correlated values: {V20;Amount} and {V7;Amount}.

In [20]:
```
s = sns.lmplot(x='V20', y='Amount',data=data_df, hue='Class', fit_reg=True,scatter_kws={
's':2})
s = sns.lmplot(x='V7', y='Amount',data=data_df, hue='Class', fit_reg=True,scatter_kws={'s'
:2})
plt.show()
```

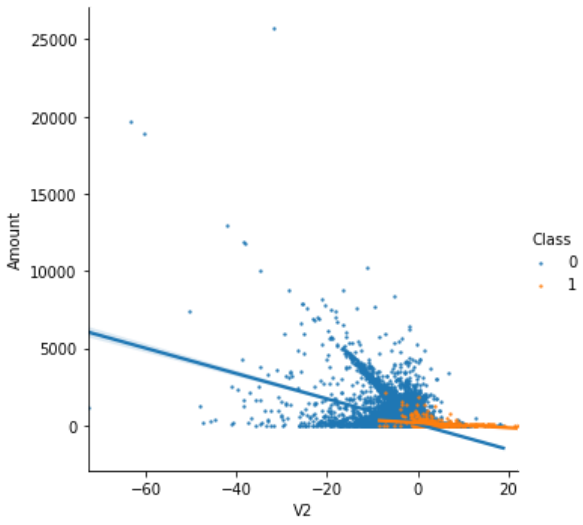


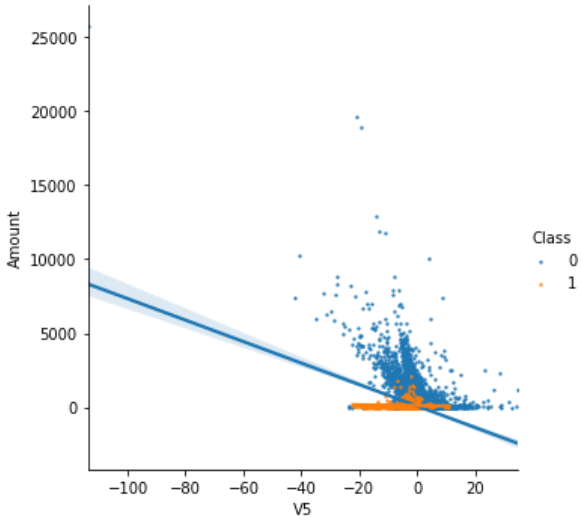




We can confirm that the two couples of features are correlated (the regression lines for **Class = 0** have a positive slope, whilst the regression line for **Class = 1** have a smaller positive slope).

Let's plot now the inverse correlated values.

In [21]:
```
s = sns.lmplot(x='V2', y='Amount',data=data_df, hue='Class', fit_reg=True,scatter_kws={'s'
:2})
s = sns.lmplot(x='V5', y='Amount',data=data_df, hue='Class', fit_reg=True,scatter_kws={'s'
:2})
plt.show()
```



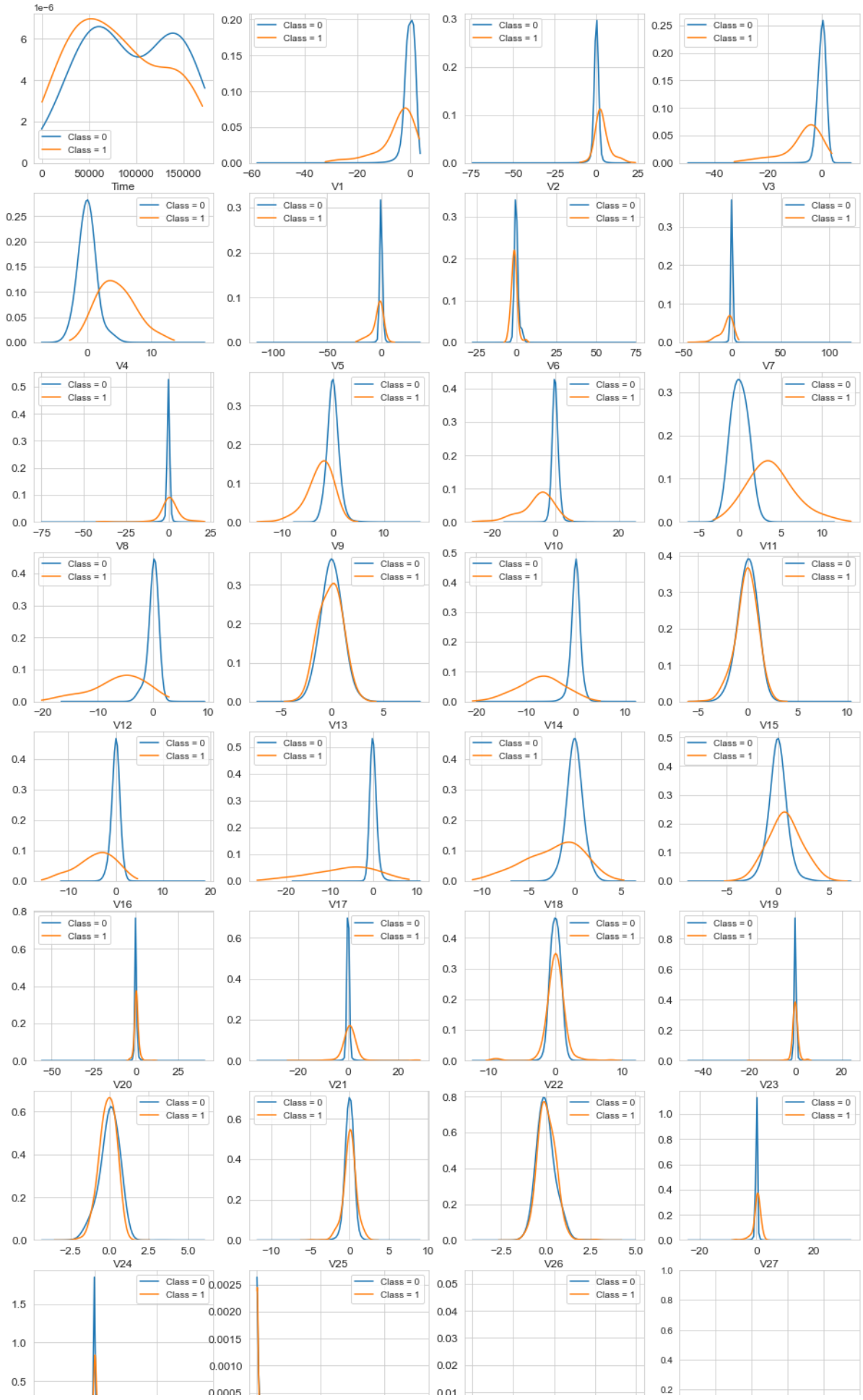

We can confirm that the two couples of features are inverse correlated (the regression lines for **Class = 0** have a negative slope while the regression lines for **Class = 1** have a very small negative slope).

# Features density plot

In [22]:
```
var = data_df.columns.values

i = 0
t0 = data_df.loc[data_df['Class'] == 0]
t1 = data_df.loc[data_df['Class'] == 1]

sns.set_style('whitegrid')
plt.figure()
fig, ax = plt.subplots(8,4,figsize=(16,28))

for feature in var:
    i += 1
    plt.subplot(8,4,i)
    sns.kdeplot(t0[feature], bw=0.5,label="Class = 0")
    sns.kdeplot(t1[feature], bw=0.5,label="Class = 1")
    plt.xlabel(feature, fontsize=12)
    locs, labels = plt.xticks()
    plt.tick_params(axis='both', which='major', labelsize=12)
plt.show();
```

```
/Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages/seaborn/distributions.py:283: UserWarning:
Data must have variance to compute a kernel density estimate.
/Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages/seaborn/distributions.py:283: UserWarning:
Data must have variance to compute a kernel density estimate.
```

```
<Figure size 432x288 with 0 Axes>
```


Time
V1
V2
V3
V4
V5
V6
V7
V8
V9
V10
V11
V12
V13
V14
V15
V16
V17
V18
V19
V20
V21
V22
V23
V24
V25
V26
V27
Class = 0
Class = 1


For some of the features we can observe a good selectivity in terms of distribution for the two values of **Class**: **V4**, **V11** have clearly separated distributions for **Class** values 0 and 1, **V12**, **V14**, **V18** are partially separated, **V1**, **V2**, **V3**, **V10** have a quite distinct profile, whilst **V25**, **V26**, **V28** have similar profiles for the two values of **Class**.

In general, with just few exceptions (**Time** and **Amount**), the features distribution for legitimate transactions (values of **Class = 0**) is centered around 0, sometime with a long queue at one of the extremities. In the same time, the fraudulent transactions (values of **Class = 1**) have a skewed (asymmetric) distribution.

# Predictive models

## Define predictors and target values

Let's define the predictor features and the target features. Categorical features, if any, are also defined. In our case, there are no categorical feature.

In [23]:
```
target = 'Class'
predictors = ['Time', 'V1', 'V2', 'V3', 'V4', 'V5', 'V6', 'V7', 'V8', 'V9', 'V10',\
       'V11', 'V12', 'V13', 'V14', 'V15', 'V16', 'V17', 'V18', 'V19',\
       'V20', 'V21', 'V22', 'V23', 'V24', 'V25', 'V26', 'V27', 'V28',\
       'Amount']
```

## Split data in train, test and validation set

Let's define train, validation and test sets.

In [24]:
```
#VALIDATION
VALID_SIZE = 0.20 # simple validation using train_test_split
TEST_SIZE = 0.20 # test size using_train_test_split

RANDOM_STATE = 2020

train_df, test_df = train_test_split(data_df, test_size=TEST_SIZE, random_state=RANDOM_STA
TE, shuffle=True )
train_df, valid_df = train_test_split(train_df, test_size=VALID_SIZE, random_state=RANDOM_
STATE, shuffle=True )
```

Let's start with a RandomForrestClassifier [3] model.

# RandomForestClassifier

## Define model parameters

Let's set the parameters for the model.

Let's run a model using the training set for training. Then, we will use the validation set for validation.

We will use as validation criterion **GINI**, which formula is **GINI = 2 * (AUC) – 1**, where **AUC** is the **Receiver Operating Characteristic – Area Under Curve (ROC–AUC)** [4]. Number of estimators is set to **100** and number of parallel jobs is set to **4**.

We start by initializing the Random Forest Classifier.

In [25]:
```
RFC_METRIC = 'gini'  #metric used for RandomForrestClassifier
NUM_ESTIMATORS = 100 #number of estimators used for RandomForrestClassifier
NO_JOBS = 4 #number of parallel jobs used for RandomForrestClassifier

clf = RandomForestClassifier(n_jobs=NO_JOBS,
                             random_state=RANDOM_STATE,
                             criterion=RFC_METRIC,
                             n_estimators=NUM_ESTIMATORS,
                             verbose=False)
```

Let's train the **Randon Forest Classifier** using the **train_df** data and **fit** function.

In [26]:
```
clf.fit(train_df[predictors], train_df[target].values)
```

```
RandomForestClassifier(bootstrap=True, ccp_alpha=0.0, class_weight=None,
                       criterion='gini', max_depth=None, max_features='auto',
                       max_leaf_nodes=None, max_samples=None,
                       min_impurity_decrease=0.0, min_impurity_split=None,
                       min_samples_leaf=1, min_samples_split=2,
                       min_weight_fraction_leaf=0.0, n_estimators=100, n_jobs=4,
                       oob_score=False, random_state=2020, verbose=False,
                       warm_start=False)
```

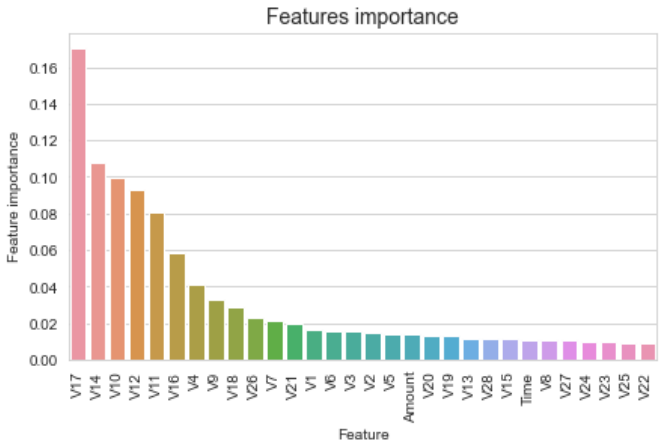
Let's now predict the **target** values for the **valid_df** data, using **predict** function.

In [27]:
```
preds = clf.predict(valid_df[predictors])
```

Let's also visualize the features importance.

## Features importance

In [28]:
```
tmp = pd.DataFrame({'Feature': predictors, 'Feature importance': clf.feature_importances_
})
tmp = tmp.sort_values(by='Feature importance',ascending=False)
plt.figure(figsize = (7,4))
plt.title('Features importance',fontsize=14)
s = sns.barplot(x='Feature',y='Feature importance',data=tmp)
s.set_xticklabels(s.get_xticklabels(),rotation=90)
plt.show()
```

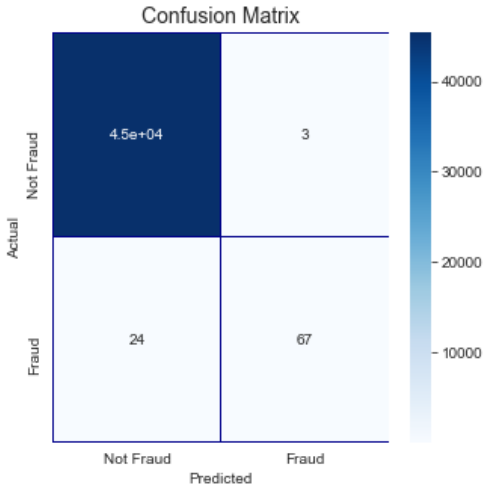


The most important features are V17, V12, V14, V10, V11, V16.

## Confusion matrix

Let's show a confusion matrix for the results we obtained.

In [29]:
```
cm = pd.crosstab(valid_df[target].values, preds, rownames=['Actual'], colnames=['Predicted'])
fig, (ax1) = plt.subplots(ncols=1, figsize=(5,5))
sns.heatmap(cm,
            xticklabels=['Not Fraud', 'Fraud'],
            yticklabels=['Not Fraud', 'Fraud'],
            annot=True,ax=ax1,
            linewidths=.2,linecolor="Darkblue", cmap="Blues")
plt.title('Confusion Matrix', fontsize=14)
plt.show()
```



## Type I error and Type II error

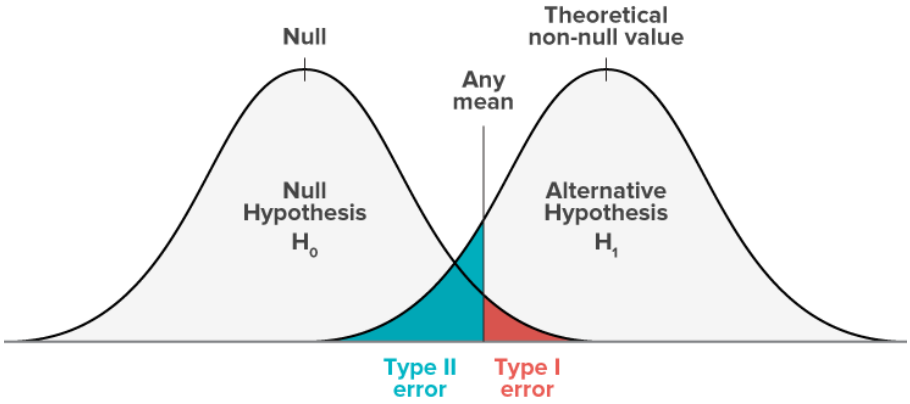
We need to clarify that confussion matrix are not a very good tool to represent the results in the case of largely unbalanced data, because we will actually need a different metrics that accounts in the same time for the **selectivity** and **specificity** of the method we are using, so that we minimize in the same time both **Type I errors** and **Type II errors**.

**Null Hypothesis** (**H0**) – The transaction is not a fraud.
**Alternative Hypothesis** (**H1**) – The transaction is a fraud.

**Type I error** – You reject the null hypothesis when the null hypothesis is actually true.
**Type II error** – You fail to reject the null hypothesis when the the alternative hypothesis is true.

**Cost of Type I error** – You erroneously presume that the the transaction is a fraud, and a true transaction is rejected.
**Cost of Type II error** – You erroneously presume that the transaction is not a fraud and a ffraudulent transaction is accepted.

The following image explains what **Type I error** and **Type II error** are:




Before re sampling lets have look at the different accuracy matrices

Accuracy = TP+TN/Total

Precison = TP/(TP+FP)

Recall = TP/(TP+FN)

TP = True possitive means no of possitve cases which are predicted possitive

TN = True negative means no of negative cases which are predicted negative

FP = False possitve means no of negative cases which are predicted possitive

FN= False Negative means no of possitive cases which are predicted negative

Now for our case recall will be a better option because in these case no of normal transacations will be very high than the no of fraud cases and sometime a fraud case will be predicted as normal. So, recall will give us a sense of only fraud cases

Resampling

in this we will resample our data with different size

then we will try to use this resampled data to train our model

then we will use this model to predict for our original data

"Let's calculate the ROC-AUC score" [4].

## Area under curve

In [30]:
```
roc_auc_score(valid_df[target].values, preds)
```

```
0.8680988851510862
```

The **ROC-AUC** score obtained with **Random Forest Classifier** is **0.868**.

# AdaBoostClassifier

AdaBoostClassifier stands for Adaptive Boosting Classifier [5].

## Prepare the model

Let's set the parameters for the model and initialize the model.

In [31]:
```
clf = AdaBoostClassifier(random_state=RANDOM_STATE,
                         algorithm='SAMME.R',
                         learning_rate=0.8,
                             n_estimators=NUM_ESTIMATORS)
```

## Fit the model

Let's fit the model.

In [32]:
```
clf.fit(train_df[predictors], train_df[target].values)
```

```
AdaBoostClassifier(algorithm='SAMME.R', base_estimator=None, learning_rate=0.8,
                   n_estimators=100, random_state=2020)
```

## Predict the target values

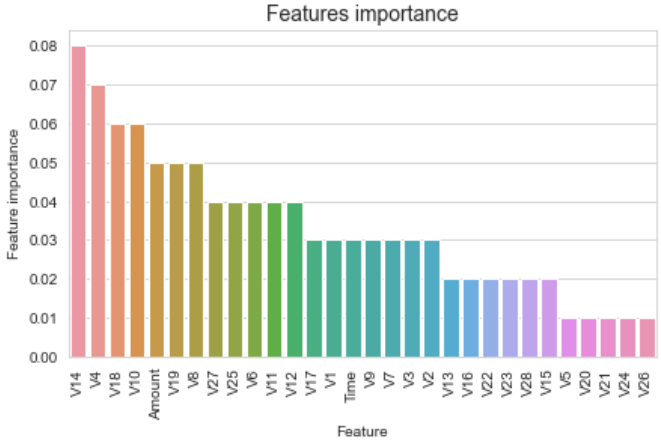
Let's now predict the **target** values for the **valid_df** data, using predict function.

In [34]:
```
preds = clf.predict(valid_df[predictors])
```

## Features importance

Let's see also the features importance.

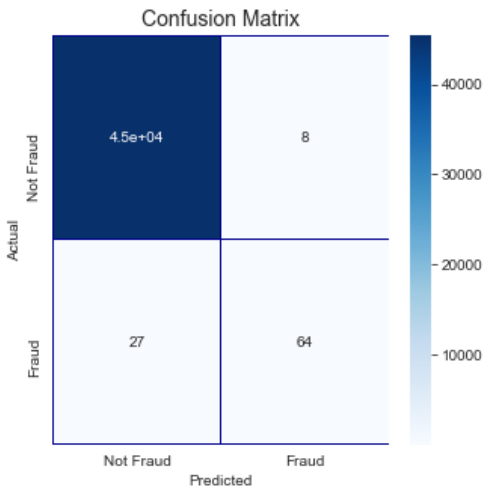
In [35]:
```
tmp = pd.DataFrame({'Feature': predictors, 'Feature importance': clf.feature_importances_
})
tmp = tmp.sort_values(by='Feature importance',ascending=False)
plt.figure(figsize = (7,4))
plt.title('Features importance',fontsize=14)
s = sns.barplot(x='Feature',y='Feature importance',data=tmp)
s.set_xticklabels(s.get_xticklabels(),rotation=90)
plt.show()
```



## Confusion matrix

Let's visualize the confusion matrix.

In [36]:
```
cm = pd.crosstab(valid_df[target].values, preds, rownames=['Actual'], colnames=['Predicte
d'])
fig, (ax1) = plt.subplots(ncols=1, figsize=(5,5))
sns.heatmap(cm,
            xticklabels=['Not Fraud', 'Fraud'],
            yticklabels=['Not Fraud', 'Fraud'],
            annot=True,ax=ax1,
            linewidths=.2,linecolor="Darkblue", cmap="Blues")
plt.title('Confusion Matrix', fontsize=14)
plt.show()
```




Let's calculate also the ROC–AUC.

## Area under curve

In [37]:
```
roc_auc_score(valid_df[target].values, preds)
```

```
0.8515603970329333
```

The ROC–AUC score obtained with AdaBoost Classifier is **0.852**.

# CatBoost Classifier

CatBoost Classifier is a gradient boosting for decision trees algorithm with support for handling categorical data [6].

## Prepare the model

Let's set the parameters for the model and initialize the model.

In [39]:
```
VERBOSE_EVAL = 50 #Print out metric result

clf = CatBoostClassifier(iterations=500,
                                learning_rate=0.02,
                                depth=12,
                                eval_metric='AUC',
                                random_seed = RANDOM_STATE,
                                bagging_temperature = 0.2,
                                od_type='Iter',
                                metric_period = VERBOSE_EVAL,
                                od_wait=100)
```

In [40]:
```
clf.fit(train_df[predictors], train_df[target].values,verbose=True)
```

```
0:      total: 962ms    remaining: 8m
50:     total: 32.6s    remaining: 4m 46s
100:    total: 1m 2s    remaining: 4m 8s
150:    total: 1m 32s   remaining: 3m 34s
200:    total: 2m 3s    remaining: 3m 3s
250:    total: 2m 33s   remaining: 2m 32s
300:    total: 3m 4s    remaining: 2m 2s
350:    total: 3m 38s   remaining: 1m 32s
400:    total: 4m 46s   remaining: 1m 10s
450:    total: 5m 48s   remaining: 37.9s
499:    total: 6m 52s   remaining: 0us


<catboost.core.CatBoostClassifier at 0x12b127050>
```

## Predict the target values

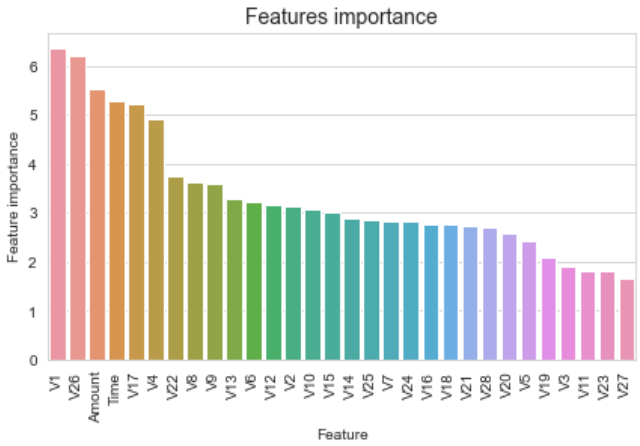
Let's now predict the **target** values for the **val_df** data, using predict function.

In [41]:
```
preds = clf.predict(valid_df[predictors])
```

## Features importance

Let's see also the features importance.

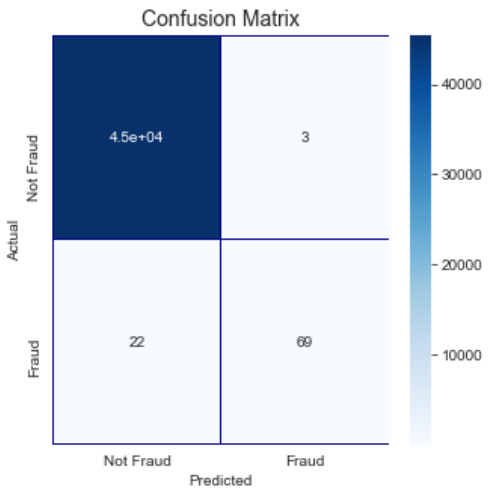
In [42]:
```
tmp = pd.DataFrame({'Feature': predictors, 'Feature importance': clf.feature_importances_
})
tmp = tmp.sort_values(by='Feature importance',ascending=False)
plt.figure(figsize = (7,4))
plt.title('Features importance',fontsize=14)
s = sns.barplot(x='Feature',y='Feature importance',data=tmp)
s.set_xticklabels(s.get_xticklabels(),rotation=90)
plt.show()
```




## Confusion matrix

Let's visualize the confusion matrix.

In [43]:
```
cm = pd.crosstab(valid_df[target].values, preds, rownames=['Actual'], colnames=['Predicte
d'])
fig, (ax1) = plt.subplots(ncols=1, figsize=(5,5))
sns.heatmap(cm,
            xticklabels=['Not Fraud', 'Fraud'],
            yticklabels=['Not Fraud', 'Fraud'],
            annot=True,ax=ax1,
            linewidths=.2,linecolor="Darkblue", cmap="Blues")
plt.title('Confusion Matrix', fontsize=14)
plt.show()
```



Let's calculate also the ROC-AUC.

## Area under curve

In [44]:
```
roc_auc_score(valid_df[target].values, preds)
```

```
0.8790878961400972
```

The ROC-AUC score obtained with CatBoostClassifier is **0.879**.

# XGBoost

XGBoost is a gradient boosting algorithm [7].

Let's prepare the model.

## Prepare the model

We initialize the DMatrix objects for training and validation, starting from the datasets. We also set some of the parameters used for the model tuning.

In [64]:
```
pip install cmake
```

```
Collecting cmake
  Downloading cmake-3.17.2-py3-none-macosx_10_6_x86_64.whl (41.8 MB)
     |████████████████████████████████████████| 41.8 MB 16.6 MB/s eta 0:00:01    |
| 1.2 MB 2.1 MB/s eta 0:00:20     |                                | 8.9 MB 4.5 MB/s eta 0:00:08     |
                | 20.1 MB 16.1 MB/s eta 0:00:02     |                                | 24.9 MB
10.1 MB/s eta 0:00:02
Installing collected packages: cmake
Successfully installed cmake-3.17.2
Note: you may need to restart the kernel to use updated packages.
```

In [65]:
```
pip install xgboost
```

```
Collecting xgboost
  Using cached xgboost-1.0.2.tar.gz (821 kB)
Requirement already satisfied: numpy in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages (f
rom xgboost) (1.18.2)
Requirement already satisfied: scipy in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages (f
rom xgboost) (1.4.1)
Building wheels for collected packages: xgboost
  Building wheel for xgboost (setup.py) ... done
  Created wheel for xgboost: filename=xgboost-1.0.2-cp37-cp37m-macosx_10_15_intel.whl size=3413992 sha256=fc9cb930366e92
5f76fae47bab5e4c06520b9eae772eb7e243d77c4bc2dabb05
  Stored in directory: /Users/zhengkunye/Library/Caches/pip/wheels/c9/1c/c9/db0188e82127a4f3c66d031147312a03e8a0676d907e
cd7fbb
Successfully built xgboost
Installing collected packages: xgboost
Successfully installed xgboost-1.0.2
Note: you may need to restart the kernel to use updated packages.
```

In [66]:
```
import xgboost as xgb

# Prepare the train and valid datasets
dtrain = xgb.DMatrix(train_df[predictors], train_df[target].values)
dvalid = xgb.DMatrix(valid_df[predictors], valid_df[target].values)
dtest = xgb.DMatrix(test_df[predictors], test_df[target].values)

#What to monitor (in this case, **train** and **valid**)
watchlist = [(dtrain, 'train'), (dvalid, 'valid')]

# Set xgboost parameters
params = {}
params['objective'] = 'binary:logistic'
params['eta'] = 0.039
params['silent'] = True
params['max_depth'] = 2
params['subsample'] = 0.8
params['colsample_bytree'] = 0.9
params['eval_metric'] = 'auc'
params['random_state'] = RANDOM_STATE
```

## Train the model

Let's train the model.

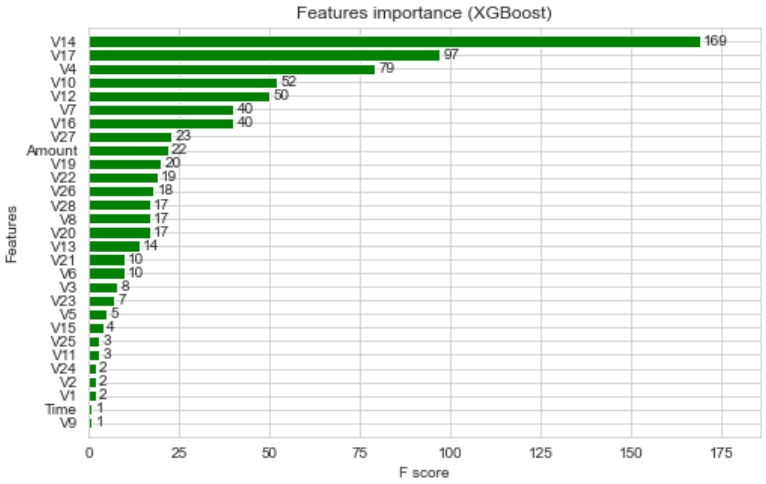
In [68]:
```
MAX_ROUNDS = 1000 #iterations
EARLY_STOP = 50 #early stop

model = xgb.train(params,
                  dtrain,
                  MAX_ROUNDS,
                  watchlist,
                  early_stopping_rounds=EARLY_STOP,
                  maximize=True,
                  verbose_eval=VERBOSE_EVAL)
```

```
[0]     train-auc:0.86186       valid-auc:0.83501
Multiple eval metrics have been passed: 'valid-auc' will be used for early stopping.

Will train until valid-auc hasn't improved in 50 rounds.
[50]    train-auc:0.91859       valid-auc:0.92285
[100]   train-auc:0.95066       valid-auc:0.94726
[150]   train-auc:0.97502       valid-auc:0.97256
[200]   train-auc:0.98911       valid-auc:0.98063
[250]   train-auc:0.99247       valid-auc:0.97975
Stopping. Best iteration:
[200]   train-auc:0.98911       valid-auc:0.98063
```

The best validation score (ROC-AUC) was **0.981**, for round **200**.

## Plot variable importance

In [70]:
```
fig, (ax) = plt.subplots(ncols=1, figsize=(8,5))
xgb.plot_importance(model, height=0.8, title="Features importance (XGBoost)", ax=ax, color
="green")
plt.show()
```




## Predict test set

We used the train and validation sets for training and validation. We will use the trained model now to predict the target value for the test set.

In [71]:
```
preds = model.predict(dtest)
```

## Area under curve

Let's calculate ROC-AUC.

In [72]:
```
roc_auc_score(test_df[target].values, preds)
```

```
0.9874261106028059
```

The AUC score for the prediction of fresh data (test set) is **0.987**.

# LightGBM

Let's continue with another gradient boosting algorithm, LightGBM [8] [9].

## Define model parameters

Let's set the parameters for the model.

In [86]:
```
params = {
          'boosting_type': 'gbdt',
          'objective': 'binary',
          'metric':'auc',
          'learning_rate': 0.05,
          'num_leaves': 7,  # we should let it be smaller than 2^(max_depth)
          'max_depth': 4,  # -1 means no limit
          'min_child_samples': 100,  # Minimum number of data need in a child(min_data_in_
leaf)
          'max_bin': 100,  # Number of bucketed bin for feature values
          'subsample': 0.9,  # Subsample ratio of the training instance.
          'subsample_freq': 1,  # frequence of subsample, <=0 means no enable
          'colsample_bytree': 0.7,  # Subsample ratio of columns when constructing each tr
ee.
          'min_child_weight': 0,  # Minimum sum of instance weight(hessian) needed in a ch
ild(leaf)
          'min_split_gain': 0,  # lambda_l1, lambda_l2 and min_gain_to_split to regulariza
tion
          'nthread': 8,
          'verbose': 0,
          'scale_pos_weight':150, # because training data is extremely unbalanced
         }
```

In [95]:
```
!pip install lightgbm
```

```
Requirement already satisfied: lightgbm in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages
(2.3.1)
Requirement already satisfied: numpy in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages (f
rom lightgbm) (1.18.2)
Requirement already satisfied: scikit-learn in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pack
ages (from lightgbm) (0.22.2.post1)
Requirement already satisfied: scipy in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages (f
rom lightgbm) (1.4.1)
Requirement already satisfied: joblib>=0.11 in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-pack
ages (from scikit-learn->lightgbm) (0.14.1)
```

In [100]:
```
import lightgbm as lgb
from lightgbm import LGBMClassifier
```

## Prepare the model

Let's prepare the model, creating the **Dataset**s data structures from the train and validation sets.

In [101]:
```
dtrain = lgb.Dataset(train_df[predictors].values,
                     label=train_df[target].values,
                     feature_name=predictors)

dvalid = lgb.Dataset(valid_df[predictors].values,
                     label=valid_df[target].values,
                     feature_name=predictors)
```

## Run the model

Let's run the model, using the **train** function.

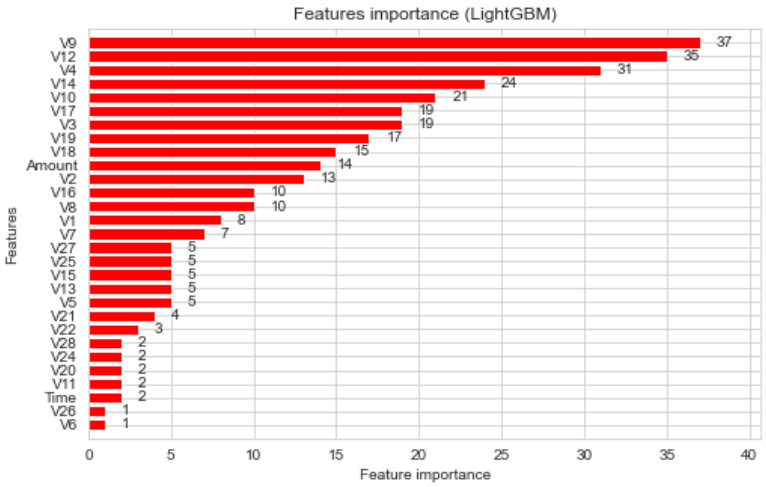
In [102]:
```
evals_results = {}

model = lgb.train(params,
                  dtrain,
                  valid_sets=[dtrain, dvalid],
                  valid_names=['train','valid'],
                  evals_result=evals_results,
                  num_boost_round=MAX_ROUNDS,
                  early_stopping_rounds=2*EARLY_STOP,
                  verbose_eval=VERBOSE_EVAL,
                  feval=None)
```

```
Training until validation scores don't improve for 100 rounds
[50]    train's auc: 0.98424    valid's auc: 0.919256
[100]   train's auc: 0.993413   valid's auc: 0.895608
[150]   train's auc: 0.989089   valid's auc: 0.89223
Early stopping, best iteration is:
[54]    train's auc: 0.985393   valid's auc: 0.92095
```

Best validation score was obtained for round 54, for which AUC ~= 0.985.

Let's plot variable importance.

In [103]:
```
fig, (ax) = plt.subplots(ncols=1, figsize=(8,5))
lgb.plot_importance(model, height=0.8, title="Features importance (LightGBM)", ax=ax,color
="red")
plt.show()
```




Let's predict now the target for the test data.

## Predict test data

In [110]:
```
preds = model.predict(test_df[predictors])
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-110-b9b52bbcc3da> in <module>
----> 1 preds = model.predict(test_df[predictors])

NameError: name 'model' is not defined
```

## Area under curve

Let's calculate the ROC-AUC score for the prediction.

In [105]:
```
roc_auc_score(test_df[target].values, preds)
```

```
0.9478398477391069
```

The ROC-AUC score obtained for the test set is **0.948**.

## Training and validation using cross-validation

Let's use now cross-validation. We will use cross-validation (KFolds) with 5 folds. Data is divided in 5 folds and, by rotation, we are training using 4 folds (n-1) and validate using the 5th (nth) fold.

Test set is calculated as an average of the predictions

In [107]:
```
#CROSS-VALIDATION
NUMBER_KFOLDS = 5 #number of KFolds for cross-validation

kf = KFold(n_splits = NUMBER_KFOLDS, random_state = RANDOM_STATE, shuffle = True)

# Create arrays and dataframes to store results
oof_preds = np.zeros(train_df.shape[0])
test_preds = np.zeros(test_df.shape[0])
feature_importance_df = pd.DataFrame()
n_fold = 0
for train_idx, valid_idx in kf.split(train_df):
    train_x, train_y = train_df[predictors].iloc[train_idx],train_df[target].iloc[train_id
x]
    valid_x, valid_y = train_df[predictors].iloc[valid_idx],train_df[target].iloc[valid_id
x]

    evals_results = {}
    model =  LGBMClassifier(
                  nthread=-1,
                  n_estimators=2000,
                  learning_rate=0.01,
                  num_leaves=80,
                  colsample_bytree=0.98,
                  subsample=0.78,
                  reg_alpha=0.04,
                  reg_lambda=0.073,
                  subsample_for_bin=50,
                  boosting_type='gbdt',
                  is_unbalance=False,
                  min_split_gain=0.025,
                  min_child_weight=40,
                  min_child_samples=510,
                  objective='binary',
                  metric='auc',
                  silent=-1,
                  verbose=-1,
                  feval=None)
    model.fit(train_x, train_y, eval_set=[(train_x, train_y), (valid_x, valid_y)],
                eval_metric= 'auc', verbose= VERBOSE_EVAL, early_stopping_rounds= EARLY_ST
OP)
```

```
    oof_preds[valid_idx] = model.predict_proba(valid_x, num_iteration=model.best_iteration
_)[:, 1]
    test_preds += model.predict_proba(test_df[predictors], num_iteration=model.best_iterat
ion_)[:, 1] / kf.n_splits

    fold_importance_df = pd.DataFrame()
    fold_importance_df["feature"] = predictors
    fold_importance_df["importance"] = clf.feature_importances_
    fold_importance_df["fold"] = n_fold + 1

    feature_importance_df = pd.concat([feature_importance_df, fold_importance_df], axis=0)
    print('Fold %2d AUC : %.6f' % (n_fold + 1, roc_auc_score(valid_y, oof_preds[valid_idx
])))
    del model, train_x, train_y, valid_x, valid_y
    gc.collect()
    n_fold = n_fold + 1
train_auc_score = roc_auc_score(train_df[target], oof_preds)
print('Full AUC score %.6f' % train_auc_score)
```

```
Training until validation scores don't improve for 50 rounds
[50]    training's auc: 0.969446        valid_1's auc: 0.959516
[100]   training's auc: 0.975391        valid_1's auc: 0.962895
Early stopping, best iteration is:
[72]    training's auc: 0.976315        valid_1's auc: 0.967322
Fold  1 AUC : 0.967322
Training until validation scores don't improve for 50 rounds
[50]    training's auc: 0.976681        valid_1's auc: 0.952957
Early stopping, best iteration is:
[49]    training's auc: 0.976834        valid_1's auc: 0.953152
Fold  2 AUC : 0.953152
Training until validation scores don't improve for 50 rounds
[50]    training's auc: 0.974153        valid_1's auc: 0.963634
[100]   training's auc: 0.977403        valid_1's auc: 0.97551
Early stopping, best iteration is:
[90]    training's auc: 0.976759        valid_1's auc: 0.976591
Fold  3 AUC : 0.976591
Training until validation scores don't improve for 50 rounds
[50]    training's auc: 0.970799        valid_1's auc: 0.975831
Early stopping, best iteration is:
[46]    training's auc: 0.97193 valid_1's auc: 0.97691
Fold  4 AUC : 0.976910
Training until validation scores don't improve for 50 rounds
[50]    training's auc: 0.972326        valid_1's auc: 0.987801
[100]   training's auc: 0.969539        valid_1's auc: 0.9881
Early stopping, best iteration is:
[81]    training's auc: 0.970589        valid_1's auc: 0.989584
Fold  5 AUC : 0.989584
Full AUC score 0.969356
```

The AUC score for the prediction from the test data was 0.969.

We prepare the test prediction, from the averaged predictions for test over the 5 folds.

In [109]:
```
pred = test_preds
```

# Conclusions

We investigated the data, checking for data unbalancing, visualizing the features and understanding the relationship between different features.
The data was split in 3 parts, a train set, a validation set and a test set.

For the first three models, we only used the train and test set.

1.We started with **Random Forest Classifier**, for which we obtained an AUC scode of **0.868** when predicting the target for the test set.

2.We followed with an **AdaBoost Classifier** model, with lower AUC score (**0.852**) for prediction of the test set target values.

3.We then followed with an **CatBoost Classifier**, with the AUC score after training 500 iterations **0.879**.

4.We then experimented with a **XGBoost** model. In this case, se used the validation set for validation of the training model. The best validation score obtained was **0.987**. Then we used the model with the best training step, to predict target value from the test data; the AUC score obtained was **0.981**.

5.We then presented the data to a **LightGBM** model. We used both train–validation split and cross–validation to evaluate the model effectiveness to predict 'Class' value, i.e. detecting if a transaction was fraudulent. With the first method we obtained values of AUC for the validation set around **0.985**. For the test set, the score obtained was **0.948**.

With the cross–validation, we obtained an AUC score for the test prediction of **0.969**.</font>

# References

[1] Credit Card Fraud Detection Database, Anonymized credit card transactions labeled as fraudulent or genuine, https://www.kaggle.com/mlg–ulb/creditcardfraud (https://www.kaggle.com/mlg–ulb/creditcardfraud)
[2] Principal Component Analysis, Wikipedia Page, https://en.wikipedia.org/wiki/Principal_component_analysis (https://en.wikipedia.org/wiki/Principal_component_analysis)
[3] RandomForrestClassifier, http://scikit–learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html (http://scikit–learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html)
[4] ROC–AUC characteristic, https://en.wikipedia.org/wiki/Receiver_operating_characteristic#Area_under_the_curve (https://en.wikipedia.org/wiki/Receiver_operating_characteristic#Area_under_the_curve)
[5] AdaBoostClassifier, http://scikit–learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostClassifier.html (http://scikit–learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostClassifier.html)
[6] CatBoostClassifier, https://tech.yandex.com/catboost/doc/dg/concepts/python–reference_catboostclassifier–docpage/ (https://tech.yandex.com/catboost/doc/dg/concepts/python–reference_catboostclassifier–docpage/)
[7] XGBoost Python API Reference, http://xgboost.readthedocs.io/en/latest/python/python_api.html (http://xgboost.readthedocs.io/en/latest/python/python_api.html)
[8] LightGBM Python implementation, https://github.com/Microsoft/LightGBM/tree/master/python–package (https://github.com/Microsoft/LightGBM/tree/master/python–package)
[9] LightGBM algorithm, https://www.microsoft.com/en–us/research/wp–content/uploads/2017/11/lightgbm.pdf (https://www.microsoft.com/en–us/research/wp–content/uploads/2017/11/lightgbm.pdf)