In [1]:
```python
import numpy as np
import pandas as pd
from keras.models import Sequential
from keras.layers import Dense, Dropout, SReLU
from keras.callbacks import BaseLogger, ModelCheckpoint,
EarlyStopping, ReduceLROnPlateau, TensorBoard
from sklearn.model_selection import StratifiedKFold
from sklearn.metrics import classification_report
from sklearn.preprocessing import StandardScaler
from sklearn.utils import compute_class_weight
```

Using TensorFlow backend.

In [2]:
```python
# Reproducible random seed
seed = 3
np.random.seed(seed)
```

In [3]:
```python
# Import and normalize the data
input_data = pd.read_csv('creditcard.csv')
input_data.iloc[:, 1:29] = StandardScaler().fit_transform(input_data.iloc[:, 1:29])
data_matrix = input_data.as_matrix()
X = data_matrix[:, 1:29]
Y = data_matrix[:, 30]
class_weights = dict(zip([0, 1], compute_class_weight('balanced', [0, 1], Y)))
```

In [4]:
```python
# Create model with k-fold cross-validation
kfold = StratifiedKFold(n_splits=3, shuffle=True, random_state=seed)
cvscores = []
predictions = np.empty(len(Y))
predictions[:] = np.NAN
proba = np.empty([len(Y), kfold.n_splits])
proba[:] = np.NAN
k = 0
```

In [5]:
```python
for train, test in kfold.split(X, Y):
    # Define model
    model = Sequential()
    model.add(Dense(28, input_dim=28))
    model.add(SReLU())
    model.add(Dropout(0.2))
    model.add(Dense(22))
    model.add(SReLU())
    model.add(Dropout(0.2))
    model.add(Dense(1, activation='sigmoid'))
    # Defining callbacks
    baselogger = BaseLogger()
    checkpointer = ModelCheckpoint(filepath="weights.hdf5", verbose=1,
 save_best_only=True)
    earlystop = EarlyStopping(monitor='val_loss', min_delta=1e-4, pati
ence=8, verbose=0, mode='auto')
    reduce_lr = ReduceLROnPlateau(monitor='val_loss', factor=0.2, pati
ence=8, min_lr=0.001)
    tensor_board = TensorBoard(log_dir='./logs', histogram_freq=0, wri
te_graph=True, write_images=False)
    # Compiling model
    metrics = ['binary_accuracy', 'fmeasure', 'precision', 'recall']
    model.compile(loss='binary_crossentropy', optimizer='adam', metric
s=metrics)
    # Fitting the model on training data
    history = model.fit(X[train], Y[train],
                        batch_size=1000,
                        nb_epoch=100,
                        verbose=0,
                        shuffle=True,
                        validation_data=(X[test], Y[test]),
                        class_weight=class_weights,
                        callbacks=[baselogger, checkpointer,
earlystop, reduce_lr, tensor_board])
    # Evaluating the model at each iteration
    scores = model.evaluate(X[test], Y[test], verbose=1)
    print("%s: %.2f%%" % (model.metrics_names[1], scores[1]*100))
    cvscores.append(scores[1] * 100)
    # Storing the predicted probabilities and iterating k
    proba[train, k] = model.predict_proba(X[train]).flatten()
    k += 1
```

```
Epoch 00000: val_loss improved from inf to 0.21429, saving model to we
ights.hdf5
Epoch 00001: val_loss improved from 0.21429 to 0.14726, saving model t
o weights.hdf5
Epoch 00002: val_loss improved from 0.14726 to 0.10732, saving model t
o weights.hdf5
Epoch 00003: val_loss did not improve
Epoch 00004: val_loss improved from 0.10732 to 0.09527, saving model t
o weights.hdf5
Epoch 00005: val_loss did not improve
Epoch 00006: val_loss improved from 0.09527 to 0.09200, saving model t
o weights.hdf5
Epoch 00007: val_loss did not improve
Epoch 00008: val_loss improved from 0.09200 to 0.08473, saving model t
o weights.hdf5
Epoch 00009: val_loss did not improve
Epoch 00010: val_loss did not improve
Epoch 00011: val_loss improved from 0.08473 to 0.06970, saving model t
o weights.hdf5
Epoch 00012: val_loss did not improve
Epoch 00013: val_loss improved from 0.06970 to 0.06660, saving model t
o weights.hdf5
Epoch 00014: val_loss improved from 0.06660 to 0.05912, saving model t
o weights.hdf5
Epoch 00015: val_loss did not improve
Epoch 00016: val_loss did not improve
Epoch 00017: val_loss improved from 0.05912 to 0.05583, saving model t
o weights.hdf5
Epoch 00018: val_loss did not improve
Epoch 00019: val_loss did not improve
Epoch 00020: val_loss did not improve
Epoch 00021: val_loss did not improve
Epoch 00022: val_loss did not improve
Epoch 00023: val_loss improved from 0.05583 to 0.05310, saving model t
o weights.hdf5
Epoch 00024: val_loss did not improve
Epoch 00025: val_loss did not improve
Epoch 00026: val_loss did not improve
Epoch 00027: val_loss did not improve
Epoch 00028: val_loss improved from 0.05310 to 0.04731, saving model t
o weights.hdf5
Epoch 00029: val_loss did not improve
Epoch 00030: val_loss improved from 0.04731 to 0.04433, saving model t
o weights.hdf5
Epoch 00031: val_loss improved from 0.04433 to 0.03697, saving model t
o weights.hdf5
Epoch 00032: val_loss did not improve
Epoch 00033: val_loss improved from 0.03697 to 0.03628, saving model t
o weights.hdf5
Epoch 00034: val_loss did not improve
Epoch 00035: val_loss did not improve
Epoch 00036: val_loss did not improve
Epoch 00037: val_loss did not improve
Epoch 00038: val_loss did not improve
Epoch 00039: val_loss improved from 0.03628 to 0.03267, saving model t
o weights.hdf5
Epoch 00040: val_loss did not improve
```

```
Epoch 00041: val_loss did not improve
Epoch 00042: val_loss improved from 0.03267 to 0.02929, saving model t
o weights.hdf5
Epoch 00043: val_loss improved from 0.02929 to 0.02892, saving model t
o weights.hdf5
Epoch 00044: val_loss did not improve
Epoch 00045: val_loss did not improve
Epoch 00046: val_loss did not improve
Epoch 00047: val_loss did not improve
Epoch 00048: val_loss improved from 0.02892 to 0.02877, saving model t
o weights.hdf5
Epoch 00049: val_loss did not improve
Epoch 00050: val_loss improved from 0.02877 to 0.02382, saving model t
o weights.hdf5
Epoch 00051: val_loss did not improve
Epoch 00052: val_loss did not improve
Epoch 00053: val_loss did not improve
Epoch 00054: val_loss improved from 0.02382 to 0.02335, saving model t
o weights.hdf5
Epoch 00055: val_loss did not improve
Epoch 00056: val_loss did not improve
Epoch 00057: val_loss did not improve
Epoch 00058: val_loss did not improve
Epoch 00059: val_loss did not improve
Epoch 00060: val_loss did not improve
Epoch 00061: val_loss did not improve
Epoch 00062: val_loss did not improve
Epoch 00063: val_loss did not improve
94880/94936 [============================>.] - ETA: 0s


□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
inary_accuracy: 99.06%
188704/189871 [============================>.] - ETA: 0s


□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
poch 00000: val_loss improved from inf to 0.37665, saving model to wei
ghts.hdf5
Epoch 00001: val_loss improved from 0.37665 to 0.21310, saving model t
o weights.hdf5
Epoch 00002: val_loss improved from 0.21310 to 0.16941, saving model t
o weights.hdf5
Epoch 00003: val_loss improved from 0.16941 to 0.14620, saving model t
o weights.hdf5
Epoch 00004: val_loss improved from 0.14620 to 0.10171, saving model t
o weights.hdf5
Epoch 00005: val_loss did not improve
Epoch 00006: val_loss improved from 0.10171 to 0.08838, saving model t
o weights.hdf5
Epoch 00007: val_loss did not improve
Epoch 00008: val_loss did not improve
Epoch 00009: val_loss did not improve
Epoch 00010: val_loss did not improve
Epoch 00011: val_loss improved from 0.08838 to 0.08335, saving model t
o weights.hdf5
Epoch 00012: val_loss improved from 0.08335 to 0.08120, saving model t
o weights.hdf5
Epoch 00013: val_loss improved from 0.08120 to 0.08083, saving model t
```

```
o weights.hdf5
Epoch 00014: val_loss improved from 0.08083 to 0.07533, saving model t
o weights.hdf5
Epoch 00015: val_loss improved from 0.07533 to 0.07325, saving model t
o weights.hdf5
Epoch 00016: val_loss improved from 0.07325 to 0.06982, saving model t
o weights.hdf5
Epoch 00017: val_loss improved from 0.06982 to 0.06306, saving model t
o weights.hdf5
Epoch 00018: val_loss did not improve
Epoch 00019: val_loss did not improve
Epoch 00020: val_loss did not improve
Epoch 00021: val_loss improved from 0.06306 to 0.05905, saving model t
o weights.hdf5
Epoch 00022: val_loss did not improve
Epoch 00023: val_loss improved from 0.05905 to 0.04736, saving model t
o weights.hdf5
Epoch 00024: val_loss did not improve
Epoch 00025: val_loss did not improve
Epoch 00026: val_loss did not improve
Epoch 00027: val_loss did not improve
Epoch 00028: val_loss did not improve
Epoch 00029: val_loss did not improve
Epoch 00030: val_loss improved from 0.04736 to 0.04331, saving model t
o weights.hdf5
Epoch 00031: val_loss did not improve
Epoch 00032: val_loss improved from 0.04331 to 0.03606, saving model t
o weights.hdf5
Epoch 00033: val_loss did not improve
Epoch 00034: val_loss did not improve
Epoch 00035: val_loss did not improve
Epoch 00036: val_loss did not improve
Epoch 00037: val_loss did not improve
Epoch 00038: val_loss did not improve
Epoch 00039: val_loss did not improve
Epoch 00040: val_loss improved from 0.03606 to 0.03059, saving model t
o weights.hdf5
Epoch 00041: val_loss did not improve
Epoch 00042: val_loss did not improve
Epoch 00043: val_loss did not improve
Epoch 00044: val_loss did not improve
Epoch 00045: val_loss did not improve
Epoch 00046: val_loss did not improve
Epoch 00047: val_loss did not improve
Epoch 00048: val_loss did not improve
Epoch 00049: val_loss did not improve
94112/94936 [============================>.] - ETA: 0s

□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
inary_accuracy: 98.82%
189696/189871 [============================>.] - ETA: 0s

□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
poch 00000: val_loss improved from inf to 0.30850, saving model to wei
ghts.hdf5
Epoch 00001: val_loss improved from 0.30850 to 0.19852, saving model t
o weights.hdf5
```

```
Epoch 00002: val_loss improved from 0.19852 to 0.18065, saving model t
o weights.hdf5
Epoch 00003: val_loss improved from 0.18065 to 0.13213, saving model t
o weights.hdf5
Epoch 00004: val_loss improved from 0.13213 to 0.12922, saving model t
o weights.hdf5
Epoch 00005: val_loss improved from 0.12922 to 0.12271, saving model t
o weights.hdf5
Epoch 00006: val_loss did not improve
Epoch 00007: val_loss improved from 0.12271 to 0.09395, saving model t
o weights.hdf5
Epoch 00008: val_loss improved from 0.09395 to 0.08226, saving model t
o weights.hdf5
Epoch 00009: val_loss did not improve
Epoch 00010: val_loss improved from 0.08226 to 0.08185, saving model t
o weights.hdf5
Epoch 00011: val_loss did not improve
Epoch 00012: val_loss did not improve
Epoch 00013: val_loss improved from 0.08185 to 0.06879, saving model t
o weights.hdf5
Epoch 00014: val_loss did not improve
Epoch 00015: val_loss did not improve
Epoch 00016: val_loss did not improve
Epoch 00017: val_loss improved from 0.06879 to 0.05830, saving model t
o weights.hdf5
Epoch 00018: val_loss did not improve
Epoch 00019: val_loss did not improve
Epoch 00020: val_loss did not improve
Epoch 00021: val_loss did not improve
Epoch 00022: val_loss did not improve
Epoch 00023: val_loss did not improve
Epoch 00024: val_loss did not improve
Epoch 00025: val_loss improved from 0.05830 to 0.04726, saving model t
o weights.hdf5
Epoch 00026: val_loss did not improve
Epoch 00027: val_loss did not improve
Epoch 00028: val_loss did not improve
Epoch 00029: val_loss did not improve
Epoch 00030: val_loss did not improve
Epoch 00031: val_loss did not improve
Epoch 00032: val_loss improved from 0.04726 to 0.04283, saving model t
o weights.hdf5
Epoch 00033: val_loss improved from 0.04283 to 0.04147, saving model t
o weights.hdf5
Epoch 00034: val_loss did not improve
Epoch 00035: val_loss did not improve
Epoch 00036: val_loss improved from 0.04147 to 0.04026, saving model t
o weights.hdf5
Epoch 00037: val_loss improved from 0.04026 to 0.03568, saving model t
o weights.hdf5
Epoch 00038: val_loss did not improve
Epoch 00039: val_loss did not improve
Epoch 00040: val_loss did not improve
Epoch 00041: val_loss did not improve
Epoch 00042: val_loss did not improve
Epoch 00043: val_loss did not improve
Epoch 00044: val_loss improved from 0.03568 to 0.03518, saving model t
```

```
o weights.hdf5
Epoch 00045: val_loss did not improve
Epoch 00046: val_loss improved from 0.03518 to 0.03150, saving model t
o weights.hdf5
Epoch 00047: val_loss did not improve
Epoch 00048: val_loss improved from 0.03150 to 0.02786, saving model t
o weights.hdf5
Epoch 00049: val_loss did not improve
Epoch 00050: val_loss did not improve
Epoch 00051: val_loss did not improve
Epoch 00052: val_loss did not improve
Epoch 00053: val_loss improved from 0.02786 to 0.02723, saving model t
o weights.hdf5
Epoch 00054: val_loss did not improve
Epoch 00055: val_loss did not improve
Epoch 00056: val_loss did not improve
Epoch 00057: val_loss did not improve
Epoch 00058: val_loss did not improve
Epoch 00059: val_loss did not improve
Epoch 00060: val_loss improved from 0.02723 to 0.02260, saving model t
o weights.hdf5
Epoch 00061: val_loss did not improve
Epoch 00062: val_loss improved from 0.02260 to 0.02193, saving model t
o weights.hdf5
Epoch 00063: val_loss did not improve
Epoch 00064: val_loss did not improve
Epoch 00065: val_loss did not improve
Epoch 00066: val_loss did not improve
Epoch 00067: val_loss did not improve
Epoch 00068: val_loss did not improve
Epoch 00069: val_loss improved from 0.02193 to 0.02009, saving model t
o weights.hdf5
Epoch 00070: val_loss did not improve
Epoch 00071: val_loss did not improve
Epoch 00072: val_loss did not improve
Epoch 00073: val_loss did not improve
Epoch 00074: val_loss did not improve
Epoch 00075: val_loss did not improve
Epoch 00076: val_loss did not improve
Epoch 00077: val_loss did not improve
Epoch 00078: val_loss did not improve
94688/94935 [============================>.] - ETA: 0s


□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
inary_accuracy: 99.34%
188480/189872 [============================>.] - ETA: 0s


□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
```

In [6]:
```python
print("%.2f%% (+/- %.2f%%)" % (np.mean(cvscores), np.std(cvscores)))
pred = np.nanmean(proba, 1) > 0.5
pred = pred.astype(int)
print(classification_report(Y, pred))
print(pd.crosstab(Y, pred, rownames=['Truth'], colnames=
['Predictions']))
```

```
99.07% (+/- 0.21%)
             precision    recall  f1-score   support

        0.0       1.00      0.99      1.00    284315
        1.0       0.24      1.00      0.38       492

avg / total       1.00      0.99      1.00    284807

Predictions       0      1
Truth
0.0          282734   1581
1.0               0    492
```