Comparing ML Techniques for Authenticating Banknotes

Ajinkya Fotedar

Overview

- Paper money remains one of the main options for exchanging products and services.
- Protection against forgery is an important concern of the money economy.
- State-of-the-art devices used for counterfeit detection tasks are often expensive.
- Classical computer vision techniques suffer from low generalization of new examples and low accuracy rates.

Machine Learning

- An agent has access to information in the form of data or patterns. The goal is to learn and understand that data to perform a future task.
- Supervised Learning: given a data set of input-output pairs, agent learns a function to map inputs to outputs.
- Classification:
    - The task of figuring out a function mapping an input point to a discrete category
    - Entails human involvement during the data set creation process
    - Data points are labelled (class) before the data set can serve as input to a model
- Deep Learning:
    - Subset of ML
    - Artificial Neural Network used as the primary unit for statistical modeling
    - Inspired by modeling agent learning off of human learning (neurons)

Data Set

- The paper’s [1] authentication approach makes four passes per banknote - image digitization, preprocessing, feature extraction, and classification.
- Image preprocessed using a wavelet transform tool for extracting spectral components (edges and textures).
- Features are then extracted using the wavelet coefficient.
    - Variance
    - Skewness
    - Curtosis
    - Entropy
- The data set [2], taken from the UC Irvine Machine Learning Repository, has features from 1372 banknotes.

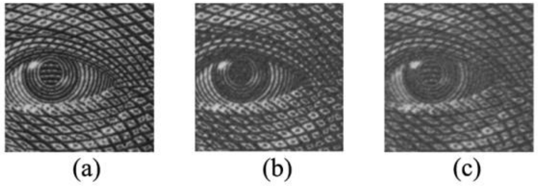


Fig. 1. Different printing techniques for banknote reproduction: (a) Genuine, (b) High-Quality Forgery and (c) Low-Quality Forgery.

[https://www.researchgate.net/publication/266673146_Banknote_Authentication]

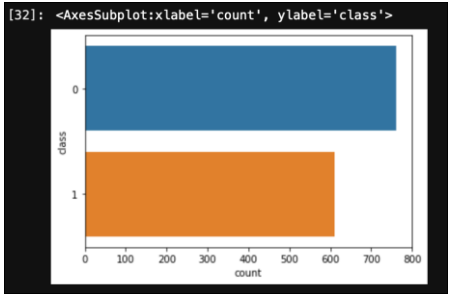
Exploratory Analysis

- It's a balanced data set.
- There are no missing values.
- An even distribution of authentic and counterfeit banknotes.

```
[25]: data.describe()
```

[25]:

	variance	skewness	curtosis	entropy	class
count	1372.000000	1372.000000	1372.000000	1372.000000	1372.000000
mean	0.433735	1.922353	1.397627	-1.191657	0.444606
std	2.842763	5.869047	4.310030	2.101013	0.497103
min	-7.042100	-13.773100	-5.286100	-8.548200	0.000000
25%	-1.773000	-1.708200	-1.574975	-2.413450	0.000000
50%	0.496180	2.319650	0.616630	-0.586650	0.000000
75%	2.821475	6.814625	3.179250	0.394810	1.000000
max	6.824800	12.951600	17.927400	2.449500	1.000000

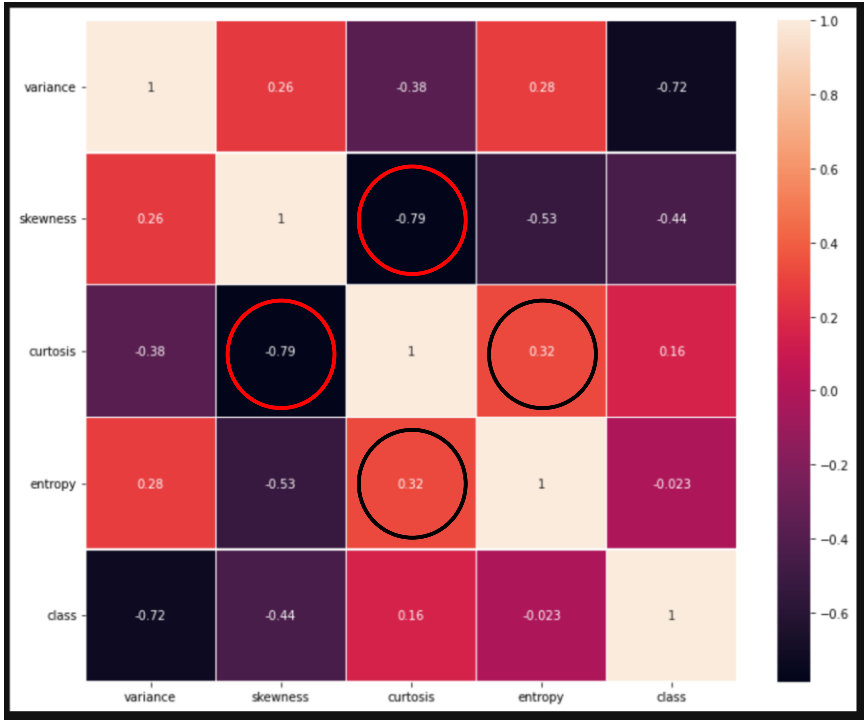



```
[28]: data.isna().sum()

[28]: variance    0
      skewness    0
      curtosis    0
      entropy     0
      class       0
      dtype: int64
```

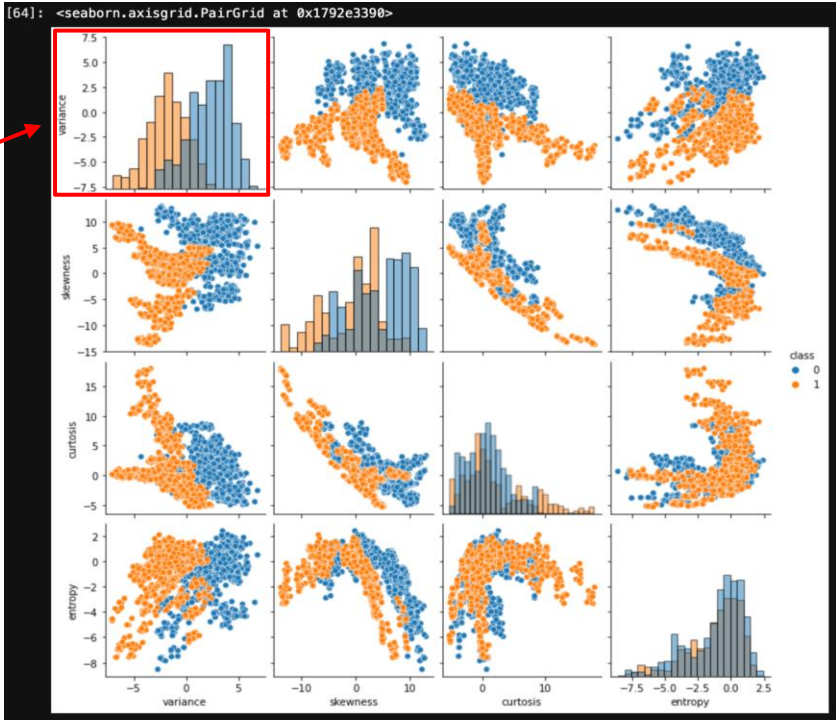
Correlation Heatmap

- Kurtosis and skewness are negatively correlated.
- High correlation between kurtosis and entropy.



Scatter Plots

- The classes are distinct and separate.
- Variance distribution is very discriminative compared to other features indicating that it might the most influential variable.



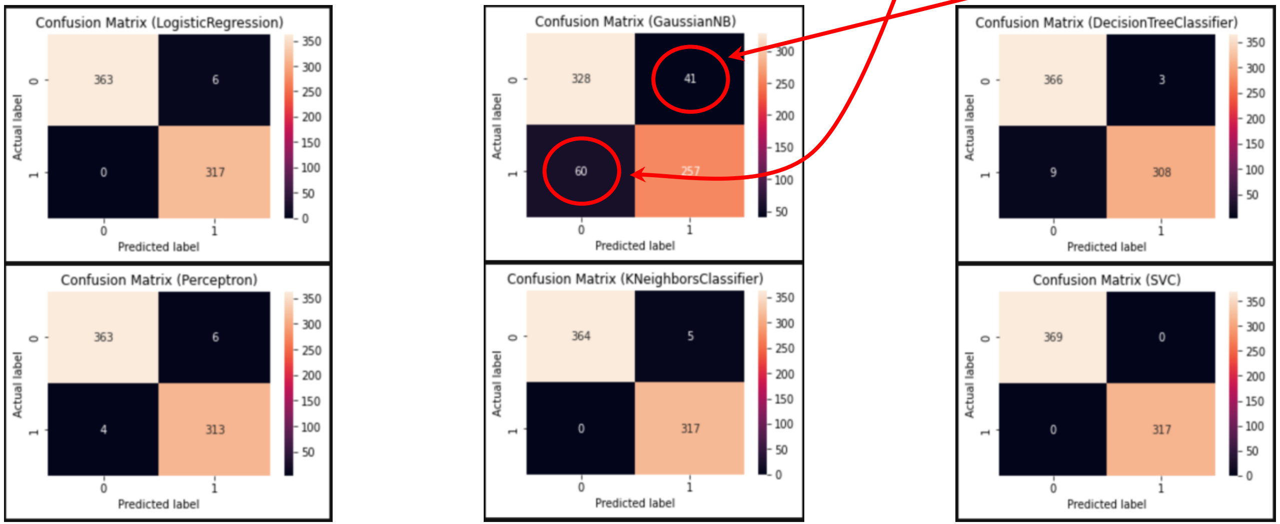
Results

- 50/50 split between training and testing.
- Support Vector Machine performs the best.
- But with the nearest neighbors parameter of KNN set to 3 (default = 1), the accuracy rate is consistently 100%.
- Naive Bayes classifier consistently had the lowest accuracy.

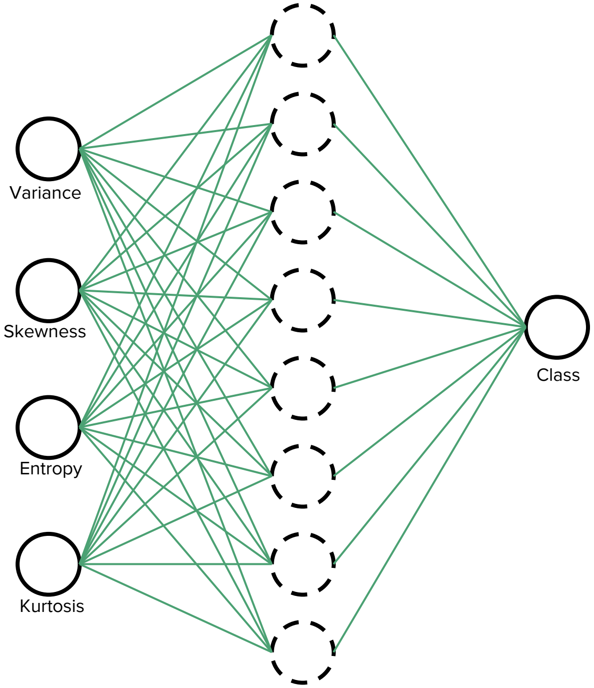
	Model	Correct	Incorrect	Accuracy
0	LogisticRegression	680	6	99.13%
1	Perceptron	676	10	98.54%
2	GaussianNB	585	101	85.28%
3	KNeighborsClassifier	681	5	99.27%
4	DecisionTreeClassifier	674	12	98.25%
5	SVC	686	0	100.00%

Confusion Matrix

- A closer look at the misclassifications a.k.a false positives and false negatives.



ANN Results




```
Epoch 1/20
22/22 [==============================] - 0s 781us/step - loss: 0.5670 - accuracy: 0.7055
Epoch 2/20
22/22 [==============================] - 0s 634us/step - loss: 0.4811 - accuracy: 0.7493
Epoch 3/20
22/22 [==============================] - 0s 664us/step - loss: 0.4164 - accuracy: 0.8076
Epoch 4/20
22/22 [==============================] - 0s 663us/step - loss: 0.3633 - accuracy: 0.8411
Epoch 5/20
22/22 [==============================] - 0s 666us/step - loss: 0.3255 - accuracy: 0.8834
Epoch 6/20
22/22 [==============================] - 0s 682us/step - loss: 0.2938 - accuracy: 0.9169
Epoch 7/20
22/22 [==============================] - 0s 665us/step - loss: 0.2686 - accuracy: 0.9257
Epoch 8/20
22/22 [==============================] - 0s 642us/step - loss: 0.2482 - accuracy: 0.9329
Epoch 9/20
22/22 [==============================] - 0s 689us/step - loss: 0.2300 - accuracy: 0.9417
Epoch 10/20
22/22 [==============================] - 0s 659us/step - loss: 0.2145 - accuracy: 0.9475
Epoch 11/20
22/22 [==============================] - 0s 663us/step - loss: 0.1996 - accuracy: 0.9548
Epoch 12/20
22/22 [==============================] - 0s 642us/step - loss: 0.1858 - accuracy: 0.9592
Epoch 13/20
22/22 [==============================] - 0s 673us/step - loss: 0.1732 - accuracy: 0.9636
Epoch 14/20
22/22 [==============================] - 0s 664us/step - loss: 0.1612 - accuracy: 0.9694
Epoch 15/20
22/22 [==============================] - 0s 664us/step - loss: 0.1495 - accuracy: 0.9694
Epoch 16/20
22/22 [==============================] - 0s 638us/step - loss: 0.1390 - accuracy: 0.9738
Epoch 17/20
22/22 [==============================] - 0s 655us/step - loss: 0.1293 - accuracy: 0.9752
Epoch 18/20
22/22 [==============================] - 0s 658us/step - loss: 0.1206 - accuracy: 0.9796
Epoch 19/20
22/22 [==============================] - 0s 659us/step - loss: 0.1128 - accuracy: 0.9825
Epoch 20/20
22/22 [==============================] - 0s 670us/step - loss: 0.1059 - accuracy: 0.9854
22/22 - 0s - loss: 0.0953 - accuracy: 0.9840
[178]: [0.09525473415851593, 0.9839650392532349]
```

Testing

Training

Summary

- Future Work:
  - Retrieving additional features from the spectral components of the print surface for a potentially better data set.
  - Extending the wavelet transform method to adapt automatically to unknown denominations.
- Libraries:
  - Scikit-learn (ML models)
  - TensorFlow (Neural Networks)
  - Seaborn (Visualization)
  - Pandas (Data preprocessing)
- References:

  - [1] https://www.researchgate.net/publication/266673146_Banknote_Authentication
  - [2] https://archive.ics.uci.edu/ml/datasets/banknote+authentication

Thank you!