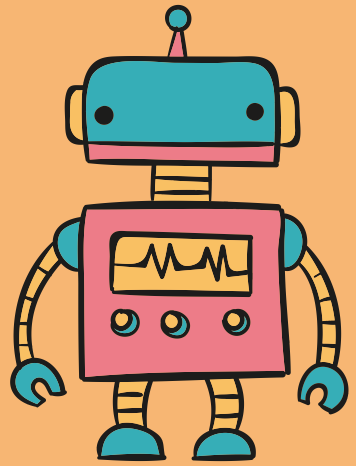
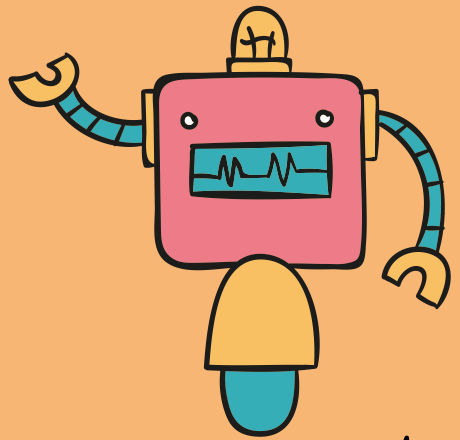
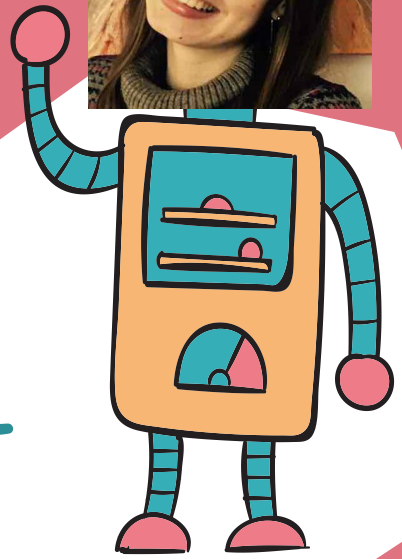
Solving Malware Classification Task using Python





Student: Yana Cherepinina
Matriculation number: 28345



My interests:

data analysis and visualization; machine learning; cybersecurity-related data analytics

Topic is important because:

application of machine learning techniques for malware detection allows to keep pace with malware evolution and combat security threats more effectively compared to other methods.

Terms

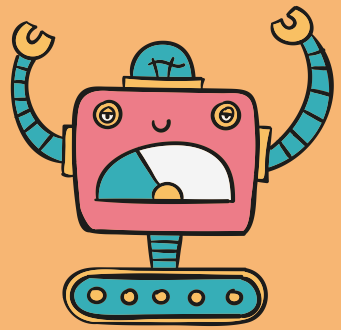
Malware

software that is specifically designed to disrupt, damage, or gain unauthorized access to a computer system

Benign Ware

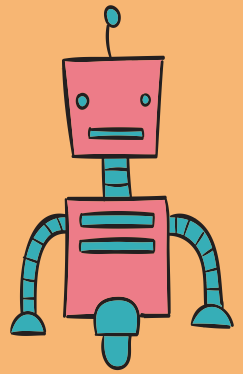
ordinary software without any malicious activity

Main Steps



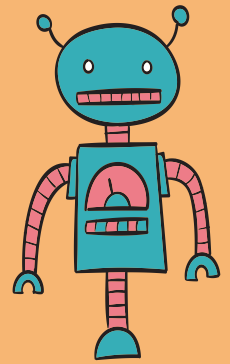
01

Dataset collection



02

Data reduction



03

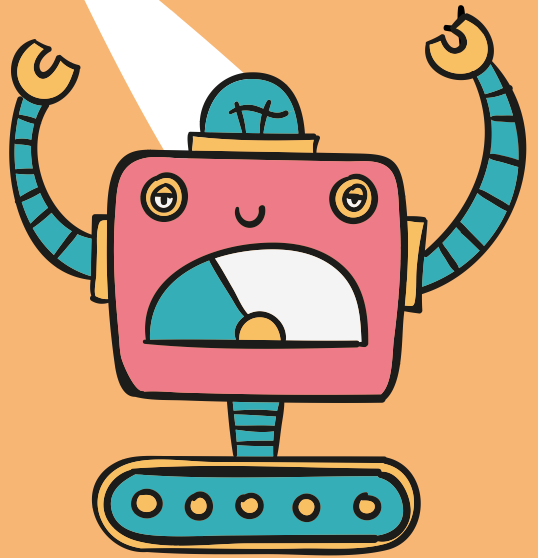
Building a machine learning model

01.

Dataset collection

With data collection, “the sooner the better”, is always the best answer.

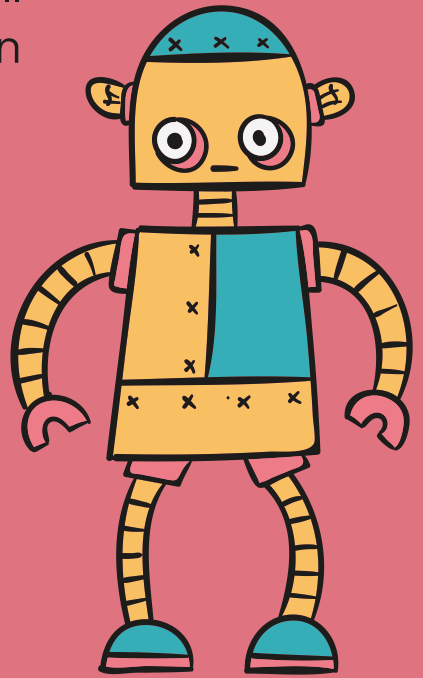
—Marissa Mayer



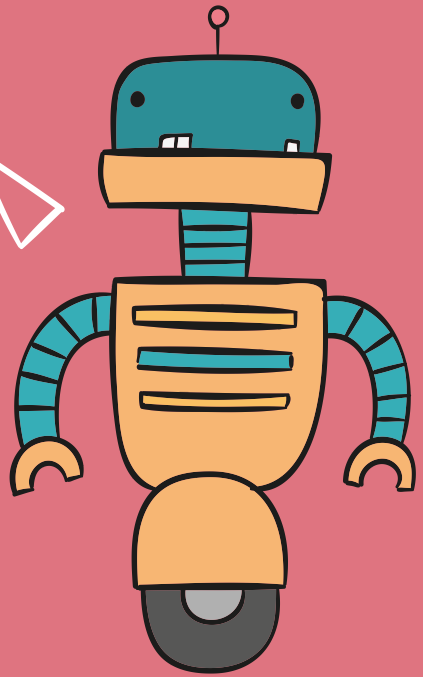
Problem

Create a dataset with features that will help the system distinguish between good and bad files:

- find files representing malicious and benign activity
- extract features from these files and tabulate them



Solution



Found:

3077 binary malicious files

collected from “VX Heavens Virus Collection”

1952 binary benign files

collected on local PC

Solution

.exe .dll .sys

pefile

Extracted:

100 features from binary portable executable files (.exe, .dll, .sys, etc.) using “pefile” python module

.csv

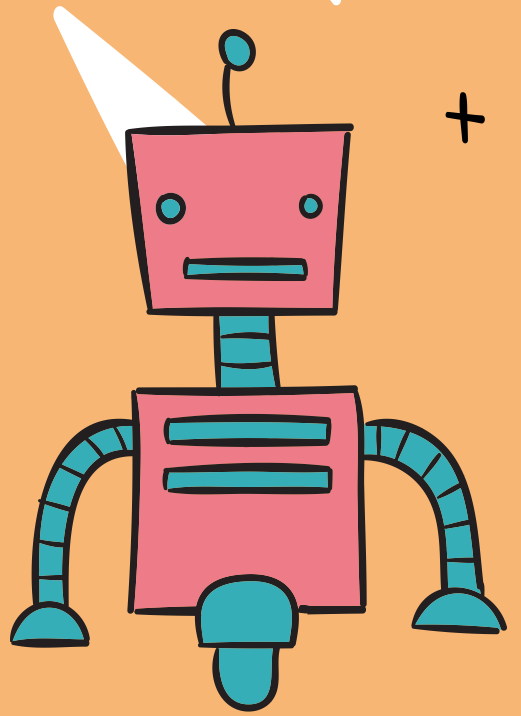
IMAGE_DIRECTORY_ENTRY_DEBUG:Size	IMAGE_DIRECTORY_ENTRY_LOAD_CONFIG:Size	IMAGE_DIRECTORY_ENTRY_IAT:Size	…	Target
0	0	880	…	Malware
28	148	1168	…	Benign
0	0	348	…	Malware
0	0	2392	…	Benign
0	0	1632	…	Benign
28	64	600	…	Malware
56	64	852	…	Malware
112	64	528	…	Benign
56	64	468	…	Benign

02.

Dataset reduction

Redundancy is expensive but indispensable.

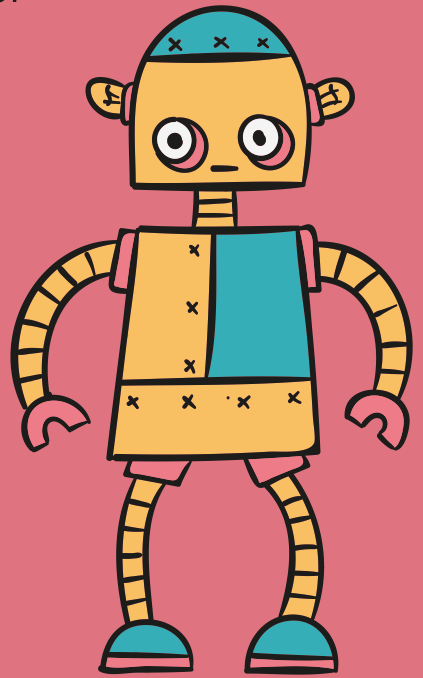
—Jane Jacobs



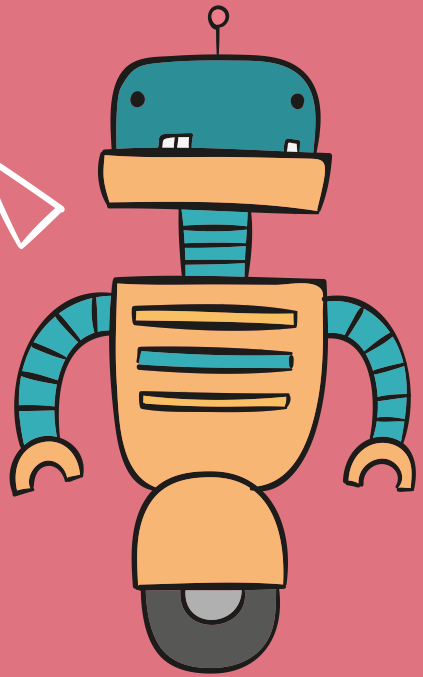
Problem

Select features that yield the most accurate results:

- apply data reduction algorithms
- obtain dataset with reduced dimensionality



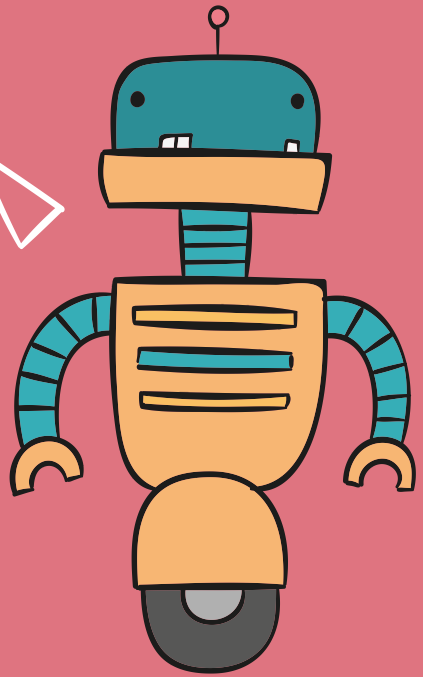
Solution



Applied:

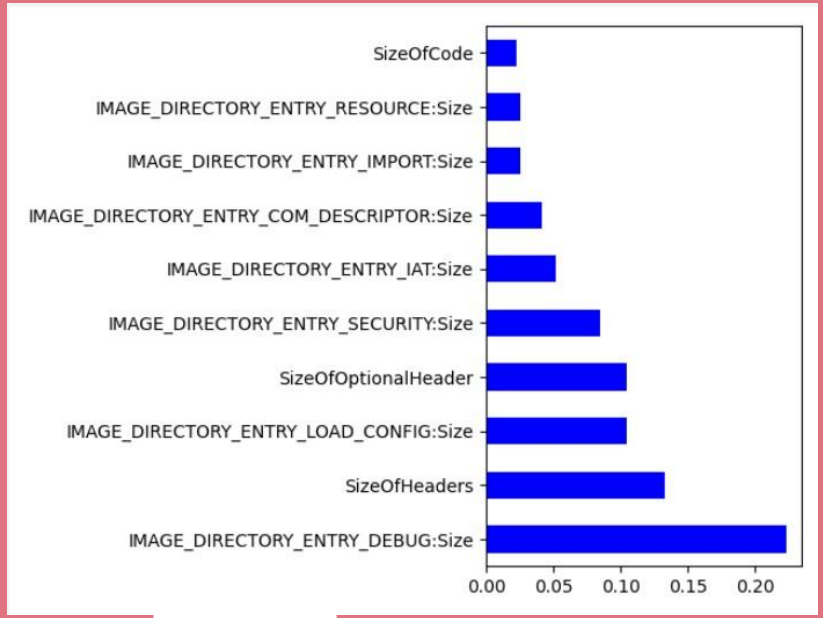
- Feature importance technique based on Gini importance metric
  - for input features with low correlation
- Principal component analysis (PCA)
  - for input features with high correlation

Solution



Obtained:

10 features with the highest scores; the higher, the more important the feature



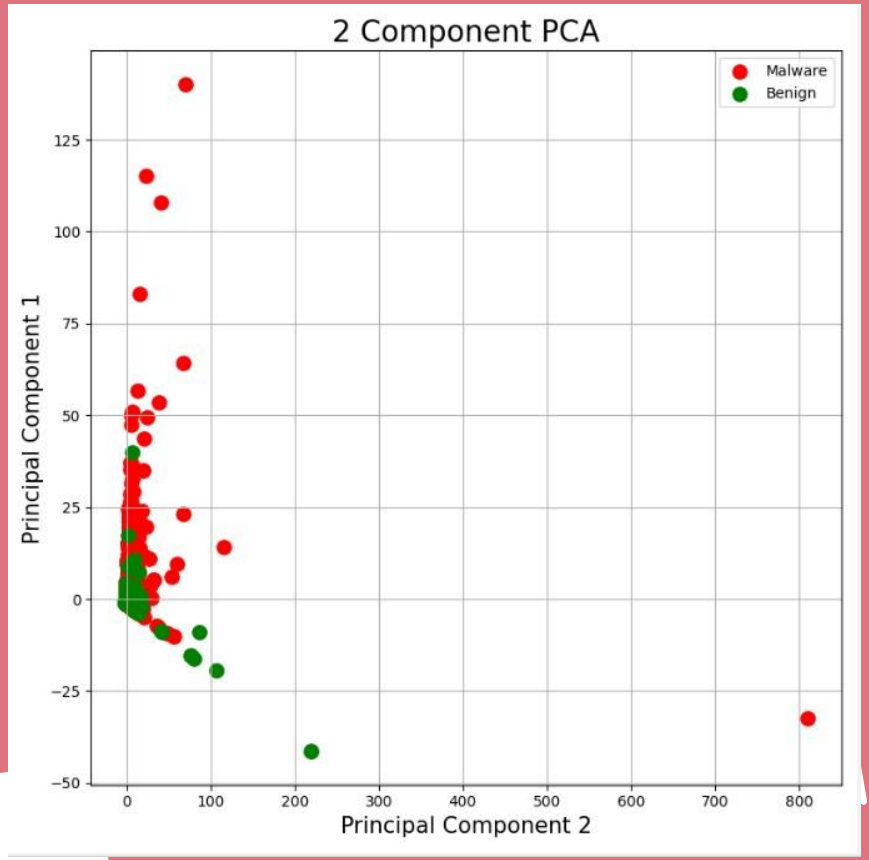
Solution

Obtained:

reduced the dimensionality of the data from **8** to **2**

Principal component 1 - **78.77%** of the variance

Principal component 2 - **13.03%** of the variance

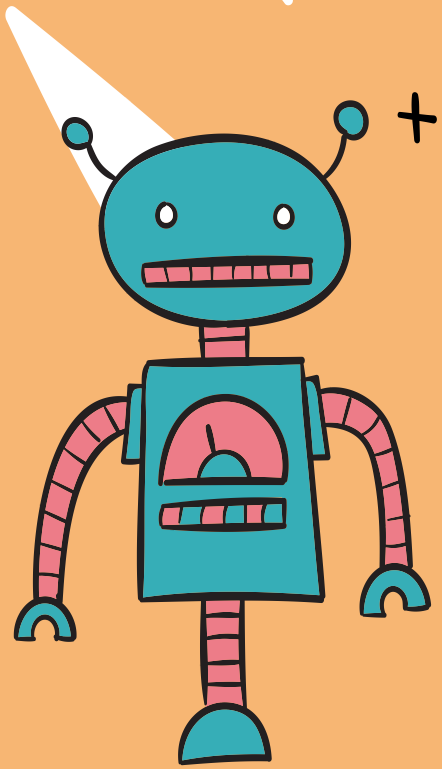


03.

Building a machine learning model

What we want is a machine that can learn from experience.

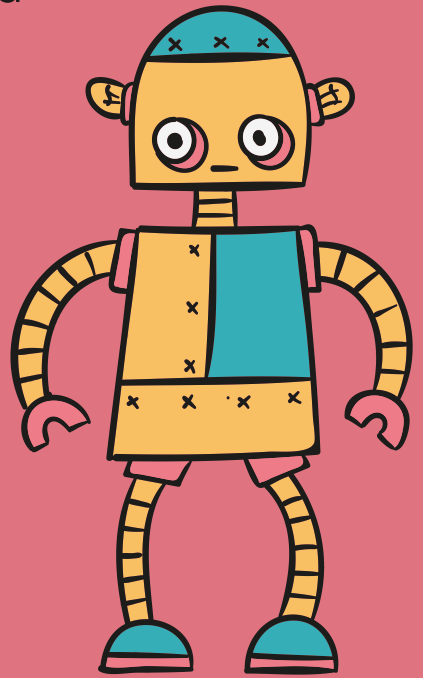
—Alan Turing



Problem

Determine which file is malicious and which is benign:

- split the data into training and validation sets
- apply a machine learning algorithm



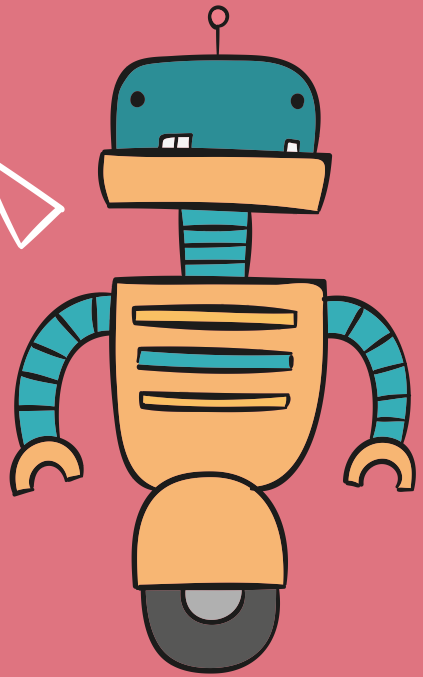
Solution

The data was split into:

5 equal folds

Each fold was used for both training and validation.

Solution

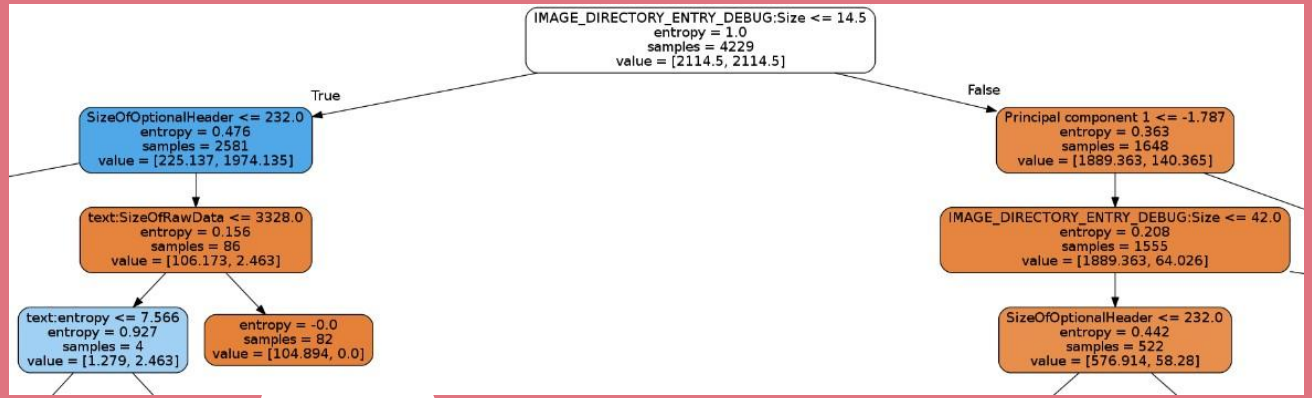


Applied:

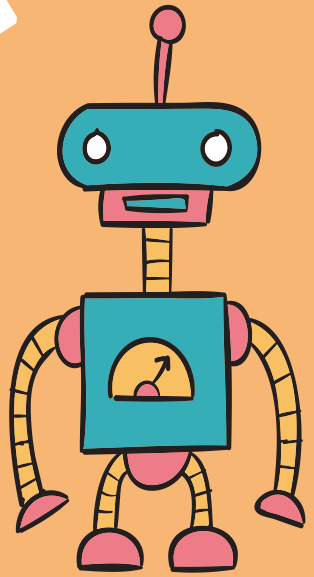
Decision Trees Classifier algorithm.

Built Decision Tree.

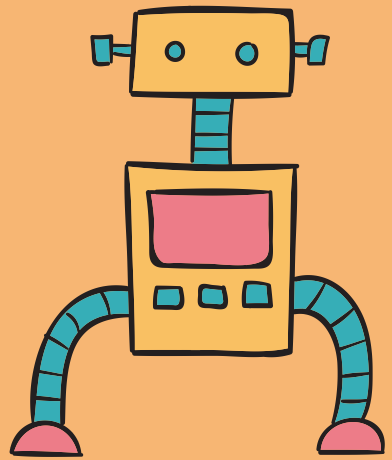
Classification rate (accuracy score): **0.9371**



Libraries & frameworks used



Pandas
Numpy
Pefile
Scikit-learn
Matplotlib
Math



Resources

Presentation template

CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, infographics & images by **Freepik**

M. Zubair Shafiq et al. (2009) PE-Miner: Mining Structural Information to Detect Malicious Executables in Realtime. In: Engin Kirda, Somesh Jha, Davide Balzarotti, eds. Recent Advances in Intrusion Detection, 12th International Symposium, Saint-Malo: Springer, pp. 121-141.

California State University (2021) Malware, Trojan, and Spyware. [online], available from: https://www.csuchico.edu/isec/stories/malware-trojans-spyware.shtml#:~:text=Malware%3A%20Malware%20is%20short%20for,access%20to%20a%20computer%20system.
[accessed 13 June 2021]

Thanks!

Does anyone have any questions?
chereyana3@gmail.com

Source code

https://github.com/YanaCh/MalwareAnalysis