MALWARE ANALYSIS AUTOMATION

Paolo Palumbo
F-Secure Labs
11th March 2015

```
<.text> @00003104  push ebx
sub_4030d0+34
  4030d0 !
  ...... ! ;-----------------------
  ...... ! ;  S U B R O U T I N E
  ...... ! ;-----------------------
  ...... ! sub_4030d0:                        ;xref c407d2b
  ...... !   sub          esp, 174h
  4030d6 !   push         ebx
  4030d7 !   push         ebp
  4030d8 !   push         esi
  4030d9 !   push         edi
  4030da !   mov          ecx, 40h
  4030df !   xor          eax, eax
  4030e1 !   lea          edi, [esp+81h]
  4030e8 !   mov          byte ptr [esp+80h], 0
  4030f0 !   repz stosd
  4030f2 !   stosw
  4030f4 !   stosb
  4030f5 !   lea          eax, [esp+80h]
  4030fc !   push         104h
  403101 !   xor          ebx, ebx
  403103 !   push         eax
  403104 !   push         ebx
  403105 !   call         dword ptr [KERNEL32.dll:GetModuleFileNameA]
  40310b !   mov          esi, [esp+190h]
  403112 !   cmp          byte ptr [esi], 0
  403115 !   jnz          loc_4031c8
```

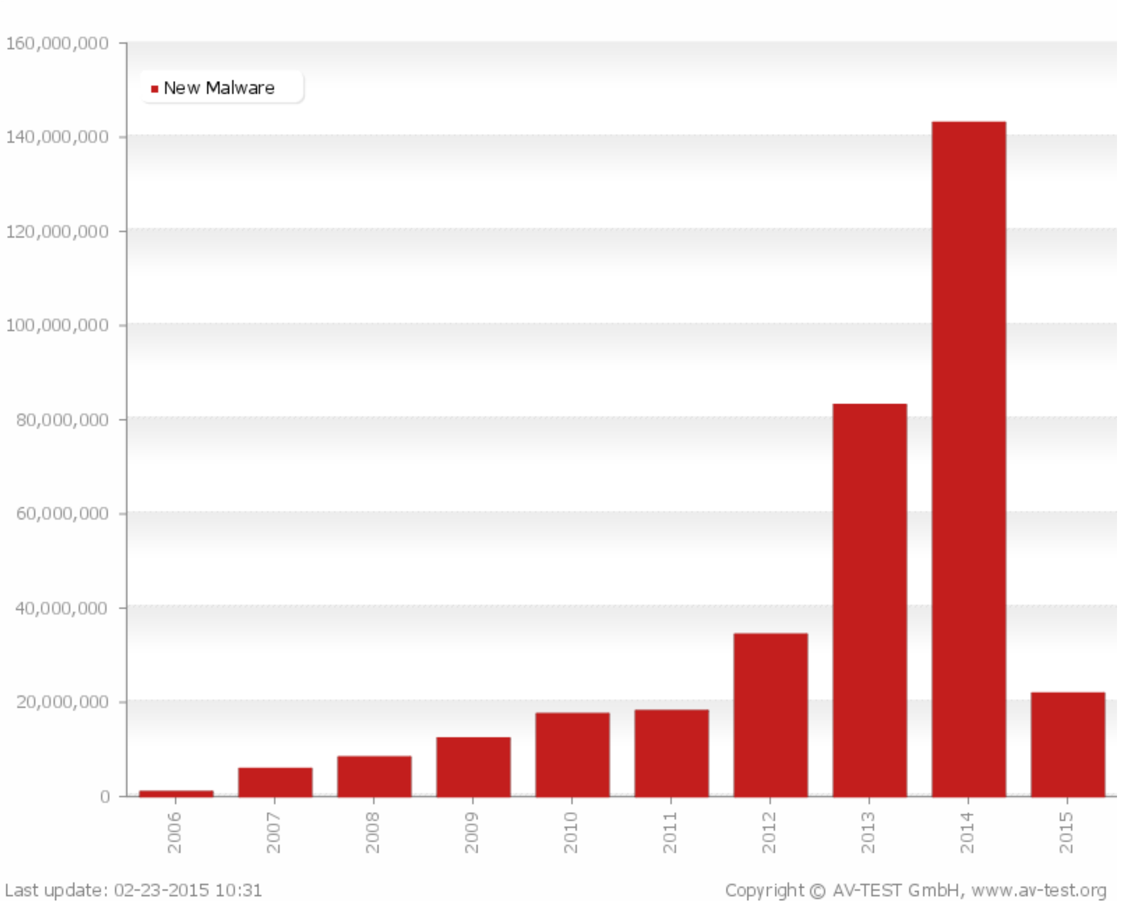
```
00762DC3  .  33C0             XOR EAX,EAX
00762DC5  .  55               PUSH EBP
00762DC6  .  68 282E7600      PUSH sample.00762E28
00762DCB  .  64:FF30          PUSH DWORD PTR FS:[EAX]
00762DCE  .  64:8920          MOV DWORD PTR FS:[EAX],ESP            break only if condition
00762DD1  .  832D 8C437800    SUB DWORD PTR DS:[78438C],1
00762DD8  .v73 40             JNB SHORT sample.00762E1A
00762DDA  .  E8 ED67E7FF      CALL sample.005D95CC
00762DDF  .  68 342E7600      PUSH sample.00762E34                  RsrcName = "TEE_CURSOR_HAND"
00762DE4  .  A1 FCA77700      MOV EAX,DWORD PTR DS:[77A7FC]
00762DE9  .  50               PUSH EAX                              hInst => NULL
00762DEA  .  E8 7169CAFF      CALL <JMP.&user32.LoadCursorA>        LoadCursorA
00762DEF  .  8BC8             MOV ECX,EAX
00762DF1  .  A1 E8757700      MOV EAX,DWORD PTR DS:[7775E8]
00762DF6  .  8B00             MOV EAX,DWORD PTR DS:[EAX]
00762DF8  .  BA E4070000      MOV EDX,7E4
00762DFD  .  E8 B6BDD0FF      CALL sample.0046EBB8
00762E02     66:B9 0500       MOV CX,5                              modified code !
00762E06  .  66:BA 0100       MOV DX,1
00762E0A  .  66:B8 6C07       MOV AX,76C
00762E0E  .  E8 7DE6CAFF      CALL sample.00411490
00762E13  .  DD1D 7CF07600    FSTP QWORD PTR DS:[76F07C]
00762E19  .  9B               WAIT
00762E1A  >  33C0             XOR EAX,EAX                           break always here
00762E1C  .  5A               POP EDX
00762E1D  .  59               POP ECX
00762E1E  .  59               POP ECX
00762E1F  .  64:8910          MOV DWORD PTR FS:[EAX],EDX
00762E22  .  68 2F2E7600      PUSH sample.00762E2F
00762E27  >  C3               RETN                                  RET used as a jump to 00762E2F
00762E28  .-E9 D324CAFF       JMP sample.00405300
00762E2D  .^EB F8             JMP SHORT sample.00762E27
00762E2F  >  5D               POP EBP
00762E30  .  C3               RETN
00762E31     00               DB 00
00762E32     00               DB 00
00762E33     00               DB 00
00762E34  .  54 45 45 5F 4:   ASCII "TEE_CURSOR_HAND",0
00762E44  .  55               PUSH EBP
00762E45  .  8BEC             MOV EBP,ESP
00762E47  .  83C4 F0          ADD ESP,-10
```

AUTOMATION

For malware analysis

WHY AUTOMATION?

- The amount of malware that security companies need to handle is continuously growing
  - Impossible to manually analyze all of the received samples
- Optimize operations by presenting the human analysts only those samples that are worthy of attention
- Guarantee operational efficiency 24/7



BENEFITS OF AUTOMATION

For a Data Security company, it facilitates the whole stack of a Security Response Unit:

- Prioritization & insight for analysts
- Knowledge extraction for Data Scientists
- Automated malware classification systems

WHAT HAPPENS IN YOUR HEAD WHEN YOU ANALYZE A SAMPLE

- Intuitively the analyst performs a process in which evidence is collected until there is enough certainty that the item under analysis is malware or not
- The evidence can be of many kinds:
  - Strings
  - Pieces of code
  - Memory contents
  - Behavioral aspects
  - OSINT information
  - Meta-data
  - …

```
loc_4D3FA3:                             ; CODE XREF: xor_string+4C↓j
                mov     eax, ebp
                call    @UniqueStringA
                movzx   edx, byte ptr [esp]
                xor     dl, [edi+ebx-1]
                mov     [eax+ebx-1], dl
                inc     ebx
                dec     esi
                jnz     short loc_4D3FA3
```

DIFFERENT POINTS OF VIEW

- A machine is different from a human
  - what might be interesting and useful for a human to determine if something is malware might not be as beneficial to a machine
- Machine Learning comes to the rescue
  - No need to re-invent the wheel!
    - A number of algorithms and approaches are available from literature
  - The application of Machine Learning concepts and techniques might not be straightforward

FEATURES

What they are and how to select them

FEATURE EXTRACTION

Feature (machine learning)

From Wikipedia, the free encyclopedia

In machine learning and pattern recognition, a **feature** is an individual measurable property of a phenomenon being observed.[1] Choosing informative, discriminating and independent features is a crucial step for effective algorithms in pattern recognition, classification and regression. Features are usually numeric, but structural features such as strings and graphs are used in syntactic pattern recognition. The concept of "feature" is related to that of explanatory variable used in statistical techniques such as linear regression.

FEATURE EXTRACTION

- For the purposes of automatic malware analysis, features can be extracted in two main ways:
  - Statically (includes automated OSINT)
  - Dynamically

STATIC FEATURES

- We call static features all of those attributes that can be extracted or derived from a file by means of static processing
  - This means that we do not execute the file
- Advantages of static extraction
  - Relaxed operational requirements
  - Fast
- Disadvantages
  - Protection, packing, obfuscation
  - Cannot access artifacts (code, data) generated at run-time

DEMO

Example extraction of static features

DYNAMIC FEATURES

- We call dynamic features all of those attributes that can be extracted or derived from the file by dynamically monitoring and/or instrumenting its execution in a controlled environment
- Advantages of dynamic extraction
  - Not hindered by obfuscation, protection and packing
  - Availability of environmental information
  - Can access artifacts generated at run-time
- Disadvantages
  - Strict operational security requirements
  - Requires extensive tuning
  - We have access only to the flow of execution

HOW LONG TO WAIT?

```
read x;
While (x>1):
  if (x%2>0):
    x = 3*x + 1;
  else:
    x=x/2;
```

OPEN SOURCE TOOLS FOR DYNAMIC ANALYSIS

- Bochs emulator
- Qemu
- VirtualBox
- Cuckoo Sandbox in combination with
  - VirtualBox
  - Vmware
  - …
- …

DEMO

Example extraction of dynamic features

FEATURE SELECTION

Feature (machine learning)

From Wikipedia, the free encyclopedia

The initial set of raw features can be redundant and too large to be managed. Therefore, a preliminary step in many applications of machine learning and pattern recognition consists of selecting a subset of features, or constructing a new and reduced set of features to facilitate learning, and to improve generalization and interpretability.

Extracting or selecting features is a combination of art and science. It requires the experimentation of multiple possibilities and the combination of automated techniques with the intuition and knowledge of the domain expert.

Cluster analysis

From Wikipedia, the free encyclopedia

(Redirected from Data clustering)

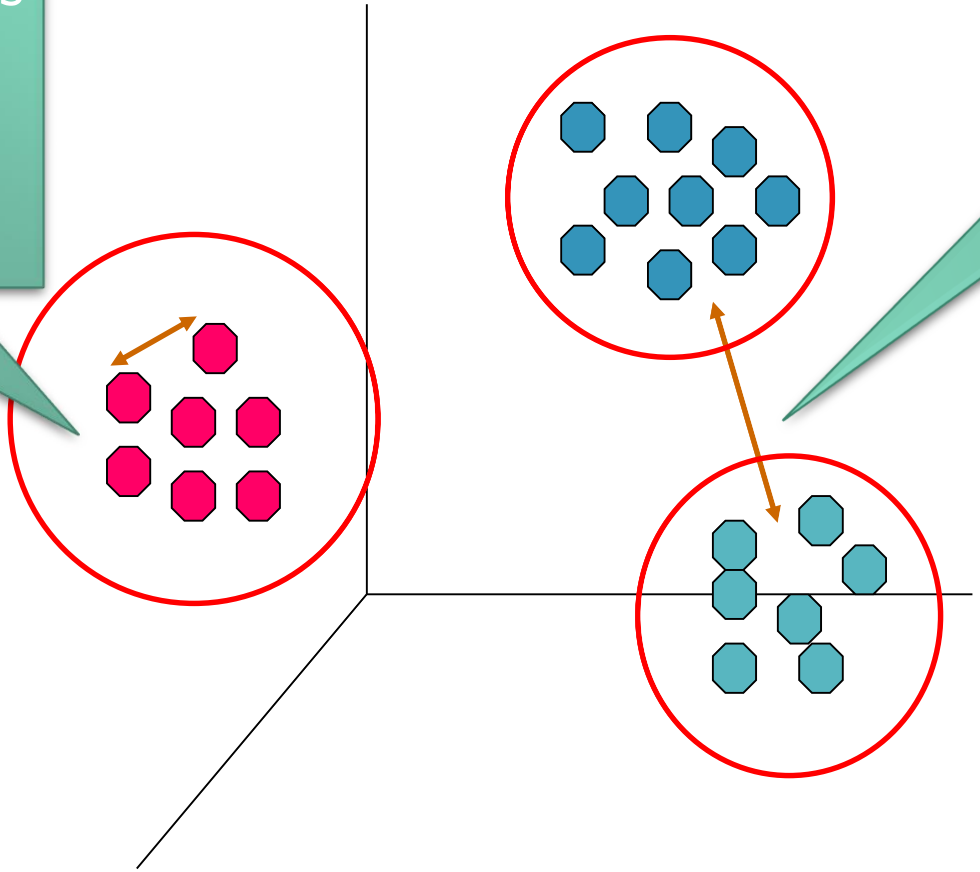
For the supervised learning approach, see Statistical classification.

Cluster analysis or **clustering** is the task of grouping a set of objects in such a way that objects in the same group (called a **cluster**) are more similar (in some sense or another) to each other than to those in other groups (clusters). It is a main task of exploratory data mining, and a common technique for statistical data analysis, used in many fields, including machine learning, pattern recognition, image analysis, information retrieval, and bioinformatics.

CLUSTER ANALYSIS

Inter-cluster distances are maximized

Intra-cluster distances are minimized



CLUSTER ANALYSIS

There are two fundamental ways to apply partitional clustering

1. By selecting X features, we map binary files to an X-dimensional space. Distances between objects are determined geometrically
2. Defining directly a distance function between objects. This requires choosing a suitable view/abstraction of the binaries.

Statistical classification

From Wikipedia, the free encyclopedia

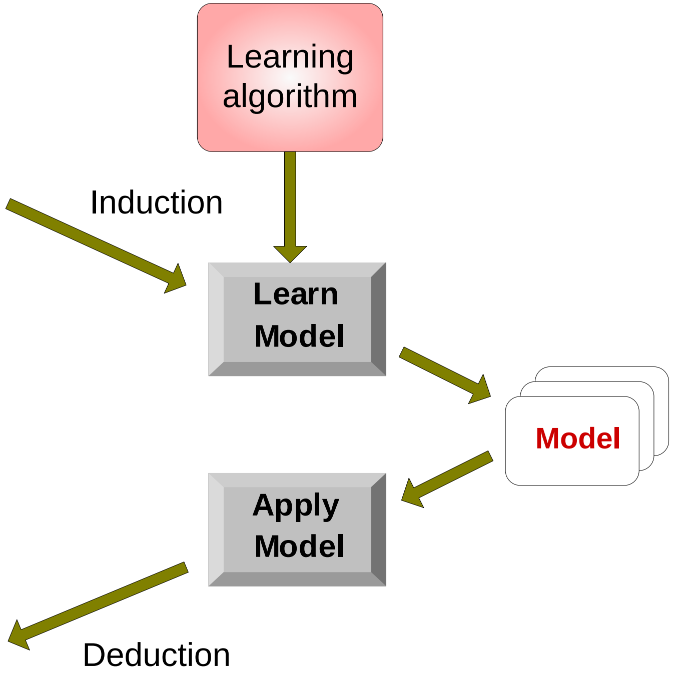
For the unsupervised learning approach, see Cluster analysis.

In machine learning and statistics, **classification** is the problem of identifying to which of a set of categories (sub-populations) a new observation belongs, on the basis of a training set of data containing observations (or instances) whose category membership is known. An example would be assigning a given email into "spam" or "non-spam" classes or assigning a diagnosis to a given patient as described by observed characteristics of the patient (gender, blood pressure, presence or absence of certain symptoms, etc.).

GENERALIZED PROCESS FOR CREATING A SUPERVISED ML CLASSIFIER

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set



Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Unknown Samples

CLASSIFICATION

- The choice of the algorithm for a classifier must be made only after careful consideration of what features will be available to the classifier
  - Certain algorithms are more suited to certain features than others
- A few examples of commonly used algorithms
  - Decision trees
  - Naive Bayes classifier
  - K-nearest neighbors classification
  - Random forests
  - Support Vector Machines (SVM)

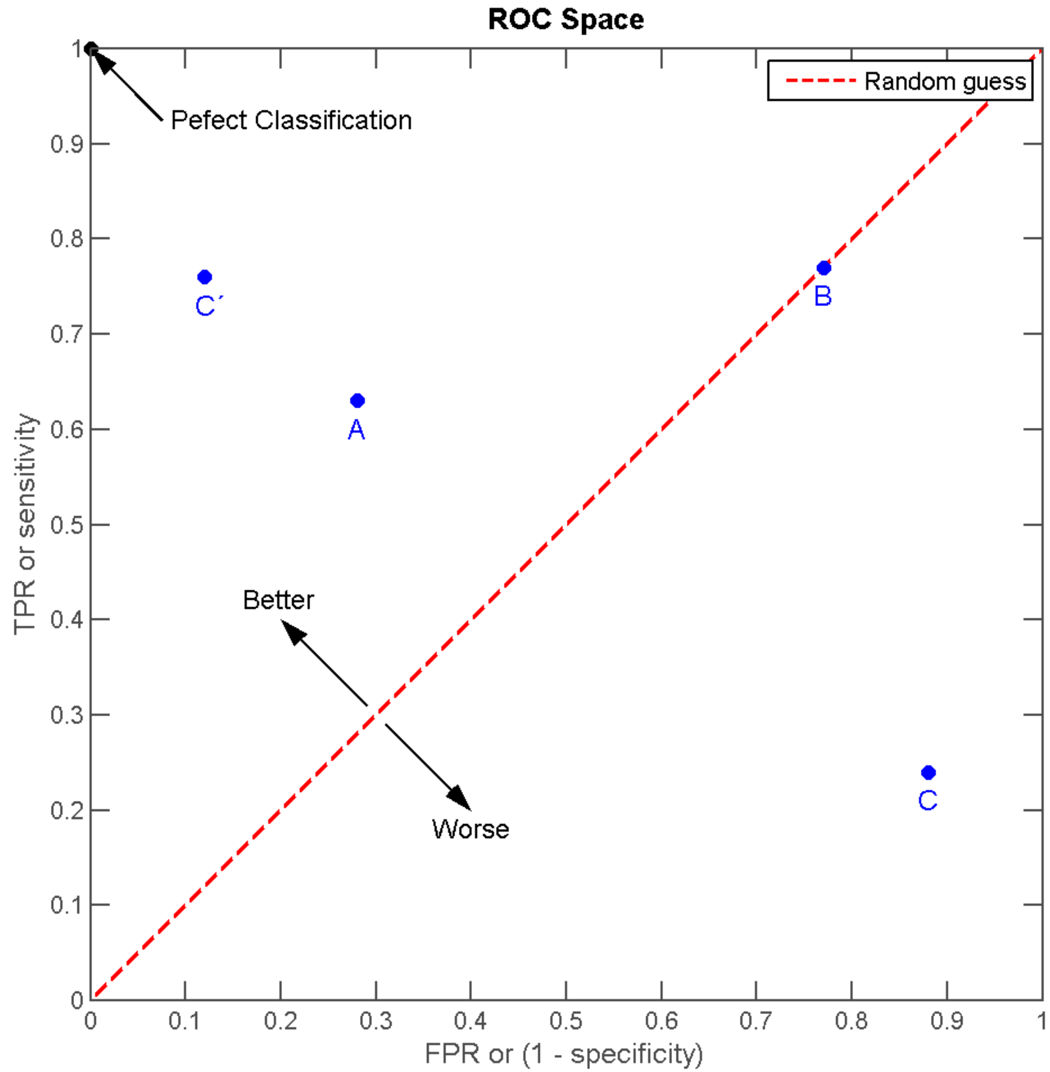
A NOTE ABOUT EVALUATION

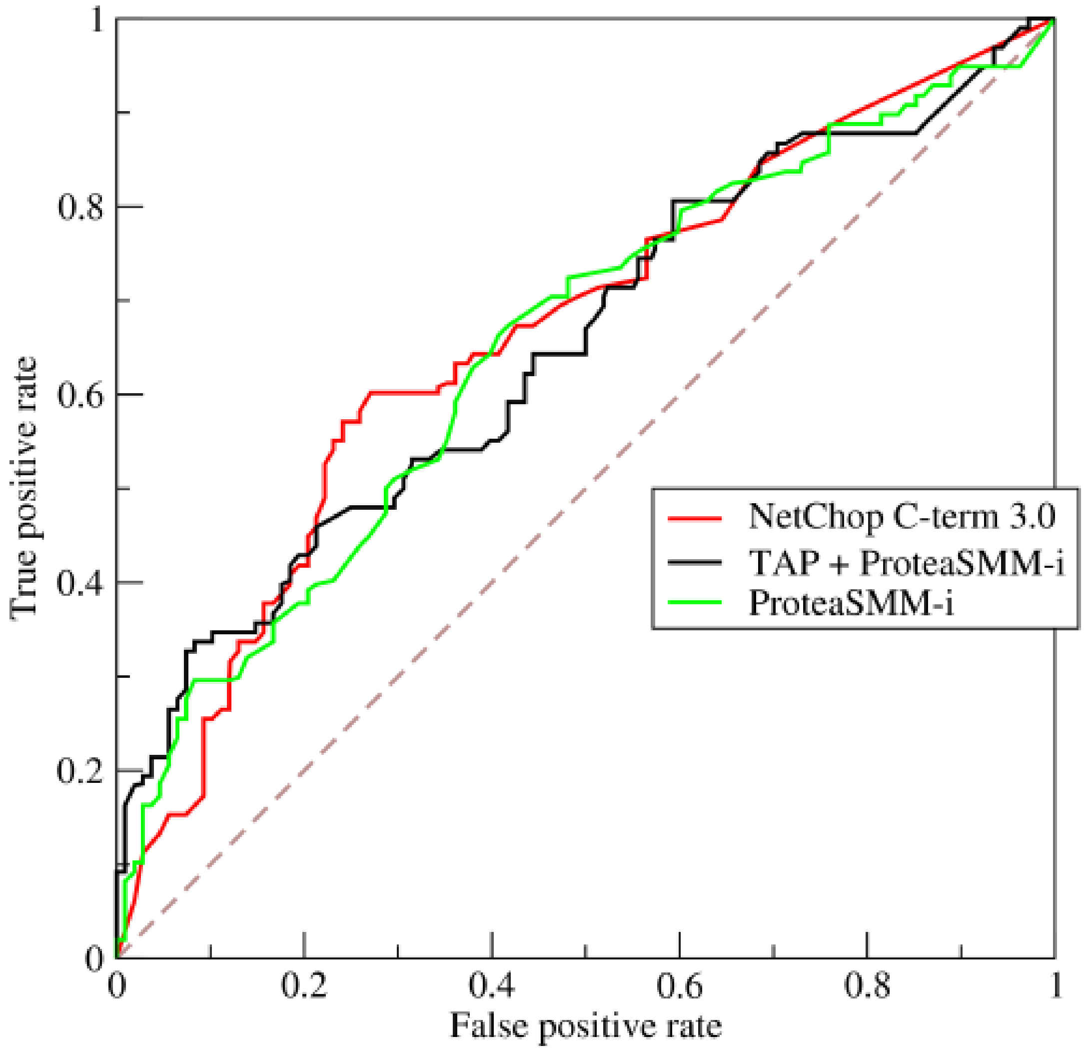
- When developing an automatic malware analysis system, it is critical to make sure that the results of the system satisfy the requirements
  - An automatic malware analysis system might end up doing automatic sample classification!
- Useful techniques to assess the quality of the automatic system
  - Cross Validation
  - ROC curve

CROSS VALIDATION

Suppose we have a model with one or more unknown parameters, and a data set to which the model can be fit (the training data set). The fitting process optimizes the model parameters to make the model fit the training data as well as possible. If we then take an independent sample of validation data from the same population as the training data, it will generally turn out that the model does not fit the validation data as well as it fits the training data. This is called overfitting, and is particularly likely to happen when the size of the training data set is small, or when the number of parameters in the model is large. Cross-validation is a way to predict the fit of a model to a hypothetical validation set when an explicit validation set is not available.

ROC CURVE


ROC Space
Random guess
Pefect Classification
C´
A
B
C
Better
Worse
TPR or sensitivity
FPR or (1 - specificity)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


NetChop C-term 3.0
TAP + ProteaSMM-i
ProteaSMM-i
True positive rate
False positive rate
0
0.2
0.4
0.6
0.8
1

SUMMARY

- To handle the problem of the ever-increasing number of new files that need analysis we need to automate the analysis process
- Depending on the type of items under analysis one needs to carefully consider
  - What features can properly describe the items we want to automatically analyze
  - How to collect these features
  - What algorithms are best suited for handling the type of features that have been selected
- No single automatic system can handle everything
  - In practice one ends up with a collection of automatic system
- There will be cases where a machine will not be able to reliably provide an answer

RESOURCES

- Parsing PE files with Python: https://code.google.com/p/pefile/
- Tools for dynamic analysis
  - Bochs emulator: http://bochs.sourceforge.net/
  - QEMU: http://wiki.qemu.org/Main_Page
  - VirtualBox: https://www.virtualbox.org/
  - Cuckoo Sandbox: http://cuckoosandbox.org/
- Machine learning: https://en.wikipedia.org/wiki/Machine_learning
  - Features: https://en.wikipedia.org/wiki/Feature_%28machine_learning%29
  - Clustering: https://en.wikipedia.org/wiki/Cluster_analysis
  - Statistical classification: https://en.wikipedia.org/wiki/Statistical_classification
  - ROC curve: https://en.wikipedia.org/wiki/ROC_Curve
  - Cross validation: https://en.wikipedia.org/wiki/Cross-validation_%28statistics%29