Computer Intrusion Detection

Lecture 7

Evaluation, ROC Analysis, and Visualization

Xiangyang Li

Outline

- Performance Measures
- Evaluating Analysis Engines
- Receiver Operator Characteristic Curve (ROC) and Confusion Matrix
- Information Visualization

Partially based on D. Marchette's book and M. Bishop's

Performance Measures

- Processing Speed
- Scalability
- Accuracy
- Security
- Interoperability
- …

Outline

- Performance Measures
- Evaluating Analysis Engines
- Receiver Operator Characteristic Curve (ROC) and Confusion Matrix
- Information Visualization

Computational Cost

Data volume

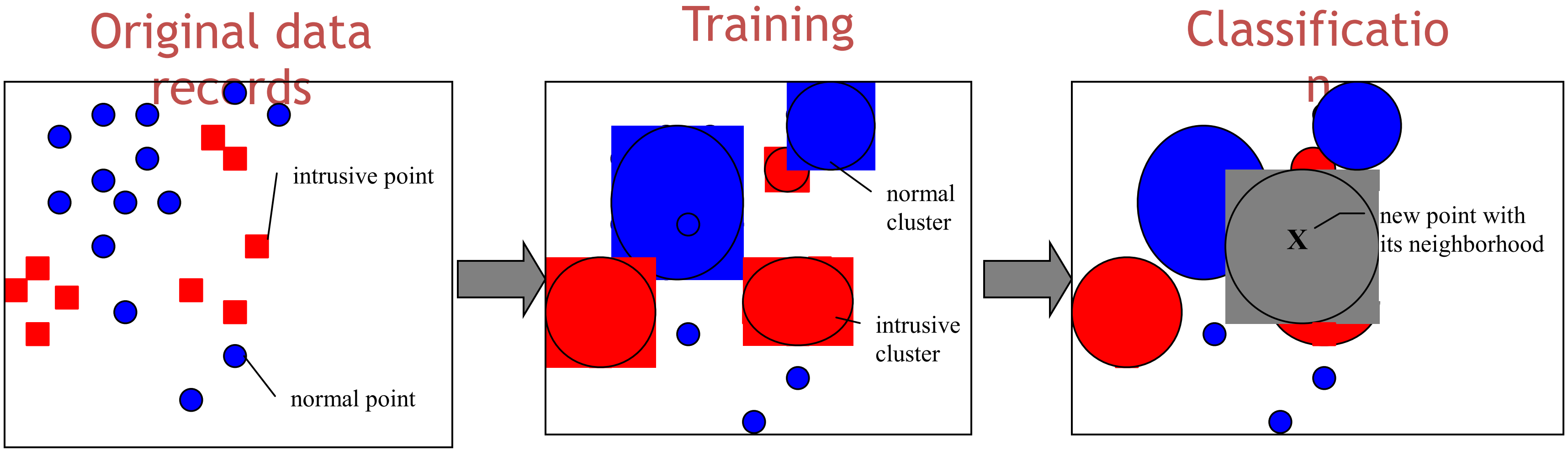
- Partitioning / filtering event streams
- N events or data records

Training

- Ideally, the computation cost is linear or polynomial of the number of training events (N).
- O(N) or O(poly(N))

Testing

- O(N) or O(poly(N))

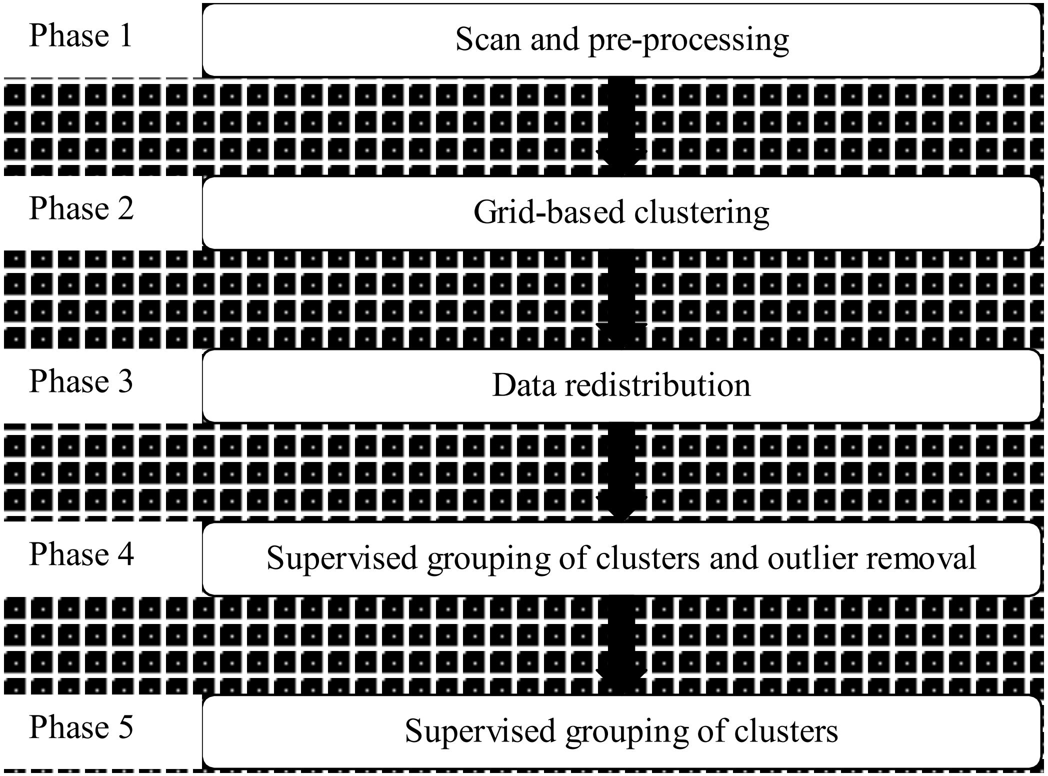


Original data records

Training

Classification

Clustering and Classification Algorithm – Supervised (CCAS)



CCAS Procedures

Computation Cost

In the CCAS model, let the number of points in training data be *N*. The number of numeric predictors is *p* and the number of nominal predictors is *m*. The number of output clusters after each phase is *M*.

- Training - Incremental clustering and redistribution:

$$O((p+m)NM)$$

Use some technique, this cost can be:

$$O((p+m)N)$$

The special hierarchical clustering

$$O(\frac{M_I(M_I-1)}{2})$$

Computation Cost (cont.)

- Classifying a data point:

$$O((p+m)M)$$

Again, use new technique:

$$O(p+m)$$

Incrementa l Learning

- Automatically learning from large amounts of activity data is done in an incremental manner, to suit the constantly evolvement of normal traffic and computer attacks.
- Each update of the pattern/profile is based on only the existing pattern/profile and the new available data.
- Previous training data is not needed anymore (ideally).

Incremental Learning Examples

- Anomaly detection
  - Chi-squared distance: $X^2 = \sum_{i=1}^{284} \frac{\left(X_i - \overline{X_i}\right)^2}{\overline{X_i}}$
- Misuse detection
  - Clustering and Classification Algorithm – Supervised (CCAS)

Error Types

- In statistical hypothesis testing, Type I and Type II errors refer to incorrect conclusions that can be drawn.
- An example of a patient being tested for HIV
  - The null hypothesis is that the patient does not have the disease; the alternative hypothesis is that HIV is present.
  - If the null hypothesis is rejected when it is in fact true (the patient tests positive for infection when the patient is well), this is a Type I error or "false positive."
  - If the null hypothesis is not rejected when it is in fact false (the patient tests negative when the patient is infected), this is a Type II error or "false negative".

Error Types in Intrusion Detection

- Different types of error may incur different costs in response and management.
  - Consequences: false alarm vs. false negative
- Strict requirements of almost 100% detection rate and near-zero false alarm rate fail many intrusion detection models in practical environments.
- There is study that assigns different weights for different types of error in selecting detection parameters.
  - Use an integrated cost function in optimization

Challenge to Intrusion Detection

- False positive rates (vs. True negative rates)
  - Particularly troublesome for anomaly detection systems
  - "Base Rate Fallacy" (Axelsson, 1998) or Bayes dilemma
- False negative rates (vs. True positive rates)
  - Failure to detect an attack event

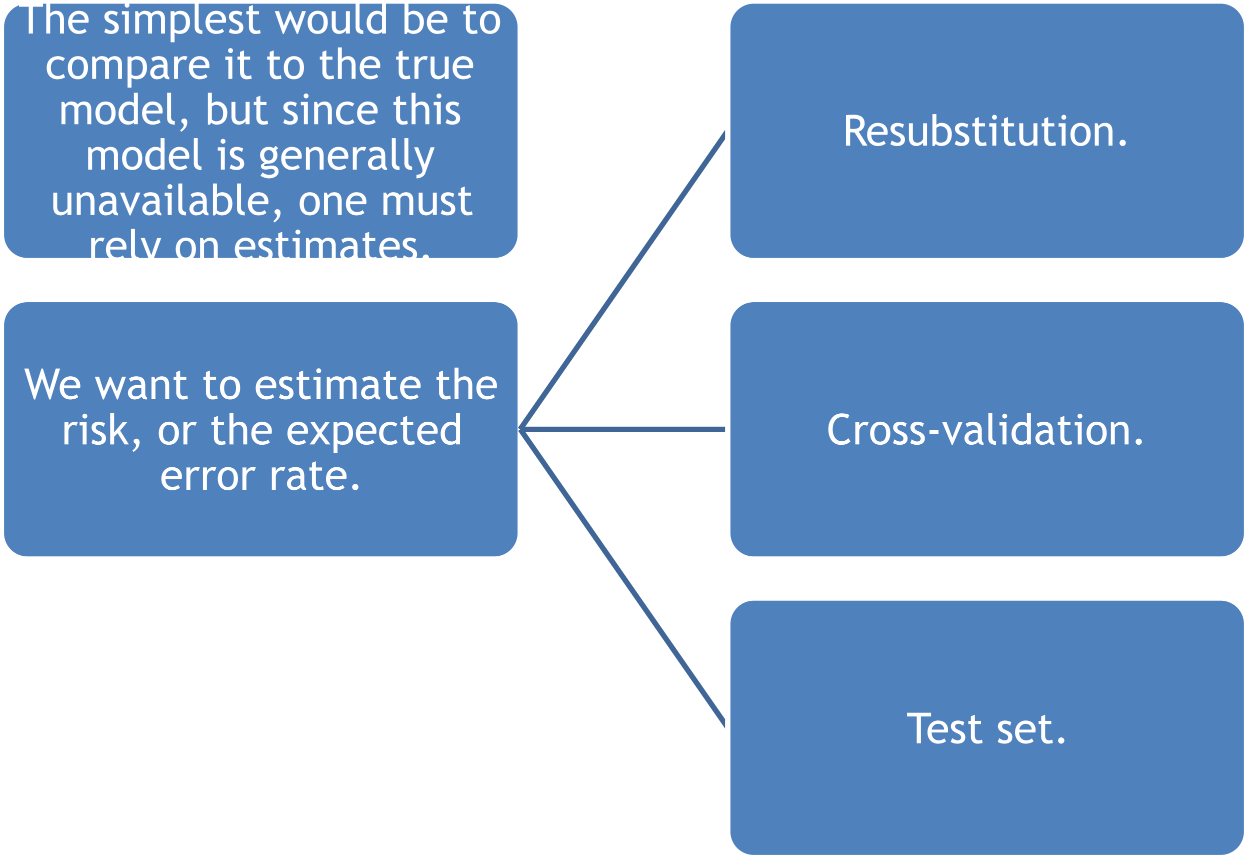
Bayes Dilemma

- Benign computer operations dominate the data generated in a computer system.
    - Assume that 99.9% of all data records are from normal activities, which is not uncommon in real world.
- 99.9% detection rate and 0.1% false alarm rate is seemingly very good performance.
- Counter-intuitively, according to the Bayes theorem we will only be 50% confident that an alarm raised by this system is indeed caused by an attack.

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}.$$

Factors Impacting Accuracy

- A classifier is a mapping *C* which maps observations to class labels (or to a probability vector with one entry per class).
- One generally constructs a classifier from data (training data) for which the true classification is known.
- Thus, technically, *C* is a function both of the new observation and the training data.



The simplest would be to compare it to the true model, but since this model is generally unavailable, one must rely on estimates.

We want to estimate the risk, or the expected error rate.

Resubstitution.

Cross-validation.

Test set.

Evaluating Classifiers

Resubstitution

- Given observation set *X* one builds (estimates) a classifier .
- A simple estimate for the error rate is to evaluate the classifier on *X* and count the number of errors.
- This will generally provide a biased estimate of the error (biased low).

Problems of Resubstitution

Sometimes, for example when the classifier has few parameters to estimate and is large, the bias may not be large and this can be a useful estimate.

Sometimes, for example with a nearest neighbor classifier, this estimate is very poor (for this classifier the re-substitution error is always 0).

There is also a so called over-fitting problem.

Cross-validation

- The idea of Cross-validation is to build the classifier only on data not used for evaluation.
- An extreme is "leave-one-out" cross-validation:
  - Let C_i be the classifier built on all the data except x_i.
  - Evaluate C_i on x_i.
  - Repeat for all the data records.
- The total error is the proportion of all the incorrectly classified x_i.

k-fold Cross-validation

ONE SIMILAR APPROACH IS TO DIVIDE THE DATA INTO *K* SETS OF EQUAL SIZE.

THE CLASSIFIER IS TRAINED ON ALL BUT ONE OF THE SETS, AND TESTED ON THE LEFT-OUT SET.

THIS IS REPEATED FOR ALL THE SETS AND THE OVERALL PERFORMANCE IS THE AVERAGE OF THE *K* CLASSIFIERS.

Cross-validation Notes

- A set of different classifiers are evaluated.
- Cross-validation removes bias by variance in performance.
- The evaluation is more accurate (comprehensive), but computationally intensive.
  - This process can be automated.

Test Set

- The “best” method for evaluation of a classifier is to use an independent and sufficient test set.
- This assumes that sufficient data is available both to estimate the classifier and to test it.
- Various bootstrap or sampling versions are also possible.

Outline

- Performance Measures
- Evaluating Analysis Engines
- Receiver Operator Characteristic Curve (ROC) and Confusion Matrix
- Information Visualization

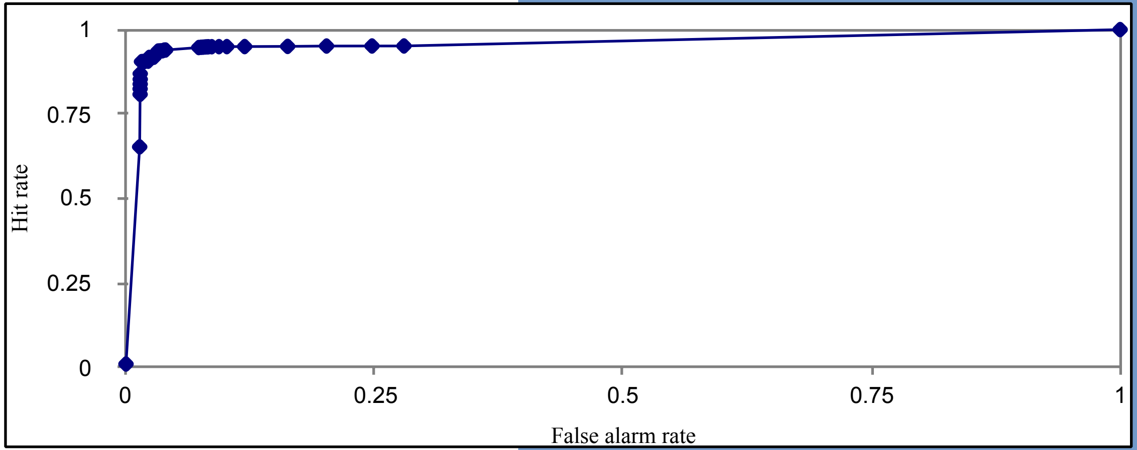
PD/PFA Trade-off

- Probability of detection (PD) or detection (hit) rate, i.e., true positive rate (TPR), is the probability of correctly identifying an attack.
- Probability of false alarm (PFA) or false alarm rate, i.e., false positive rate (PPR) is the probability of alerting on a non-attack.
- You (generally) cannot simultaneously get PD=1 and PFA=0.
- You can get either one: PD=1 or PFA=0.

Receiver Operating Characteristic (ROC) Curves

- In signal detection theory, a receiver operating characteristic, or simply ROC curve, is a graphical plot of the sensitivity for a binary classifier system as its discrimination threshold is varied.
  - Set a parameter: for example the threshold in IDES detection model.
  - For each value of the parameter, plot PD vs. PFA.

ROC Example



Use of ROC

Find the parameter value producing the optimal trade-off.

- For example, the cost function to minimize can be $\alpha*(1-PD)+(1-\alpha)*PFA$.

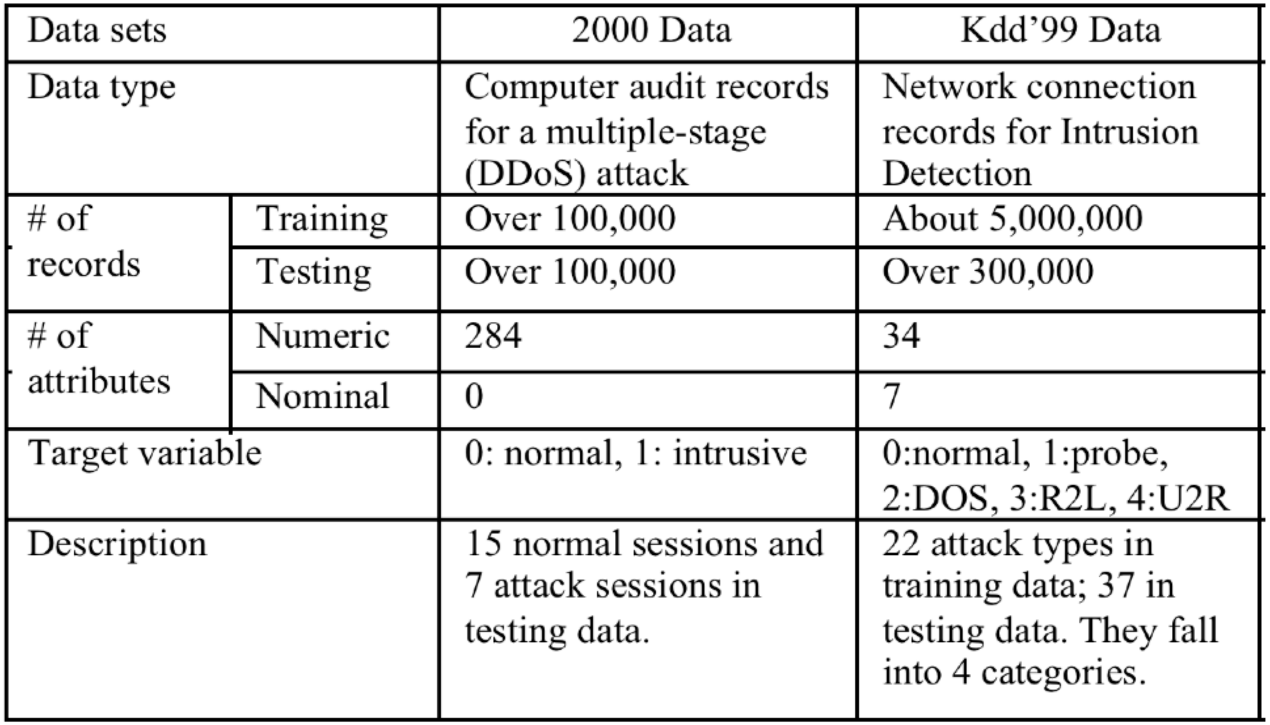
Area Under the Curve (AUC) for ROC

- This is a commonly used measure.
- However it has its own weakness.

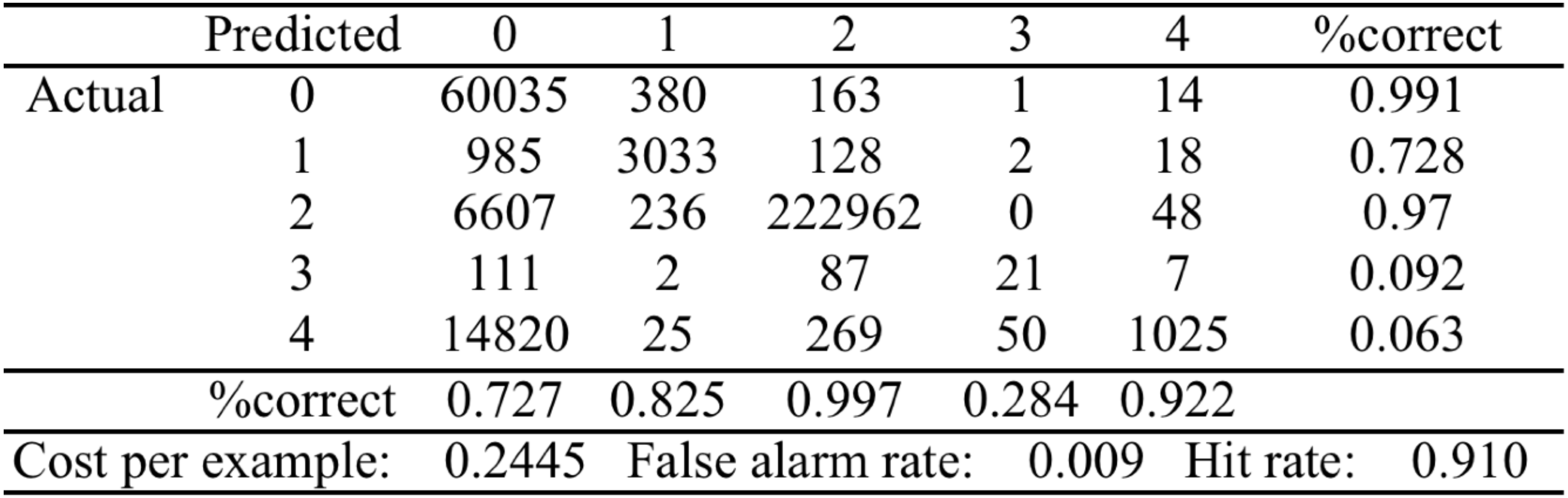
Challenge to ROC

- What is we have multiple classification classes?

DARPA/MITLL Testbed Data



Data sets		2000 Data	Kdd’99 Data
Data type		Computer audit records for a multiple-stage (DDoS) attack	Network connection records for Intrusion Detection
# of records	Training	Over 100,000	About 5,000,000
	Testing	Over 100,000	Over 300,000
# of attributes	Numeric	284	34
	Nominal	0	7
Target variable		0: normal, 1: intrusive	0:normal, 1:probe, 2:DOS, 3:R2L, 4:U2R
Description		15 normal sessions and 7 attack sessions in testing data.	22 attack types in training data; 37 in testing data. They fall into 4 categories.



	Predicted	0	1	2	3	4	%correct
Actual	0	60035	380	163	1	14	0.991
	1	985	3033	128	2	18	0.728
	2	6607	236	222962	0	48	0.97
	3	111	2	87	21	7	0.092
	4	14820	25	269	50	1025	0.063
	%correct	0.727	0.825	0.997	0.284	0.922	
Cost per example:		0.2445	False alarm rate:		0.009	Hit rate:	0.910

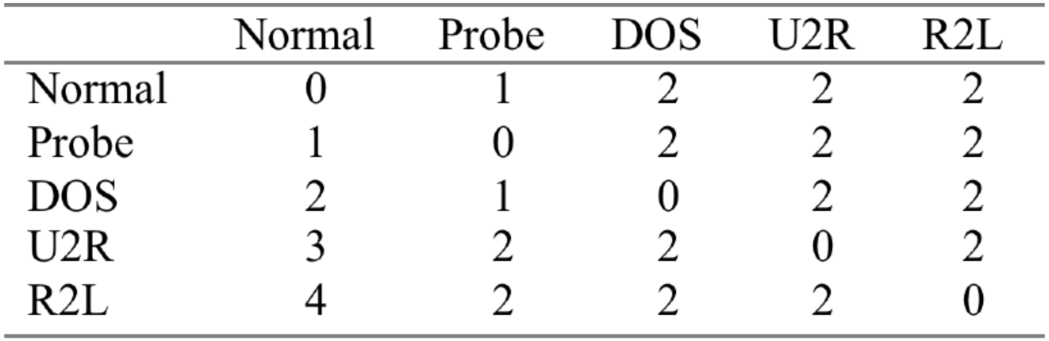
Confusion Matrix

The Results of KDD'99 Context

- From confusion matrix of predicted and actual classes, error rate could be calculated.
- The false alarm rate and hit rate of the winning decision tree technique in KDD Cup 1999 contest are 0.5% and 91.8% respectively.

Cost Matrix

- There is a cost matrix for scoring the techniques in KDD’99 contest. Based on the confusion matrix and this cost matrix, the average cost per test example is calculated. The smaller this cost, the better the performance.
- The best 17 participants have an average cost per test example ranging from 0.2331 to 0.2684.



	Normal	Probe	DOS	U2R	R2L
Normal	0	1	2	2	2
Probe	1	0	2	2	2
DOS	2	1	0	2	2
U2R	3	2	2	0	2
R2L	4	2	2	2	0

Outline

Performance Measures

Evaluating Analysis Engines

Receiver Operator Characteristic Curve (ROC) and Confusion Matrix

Information Visualization

Information Visualizatio n

- The main problem with intrusion detection is the large amount of data that the user is faced with.
- Information visualization can be utilized together with automated anomaly detection and misuse detection.
- Visualization can use the original data or the detection result.

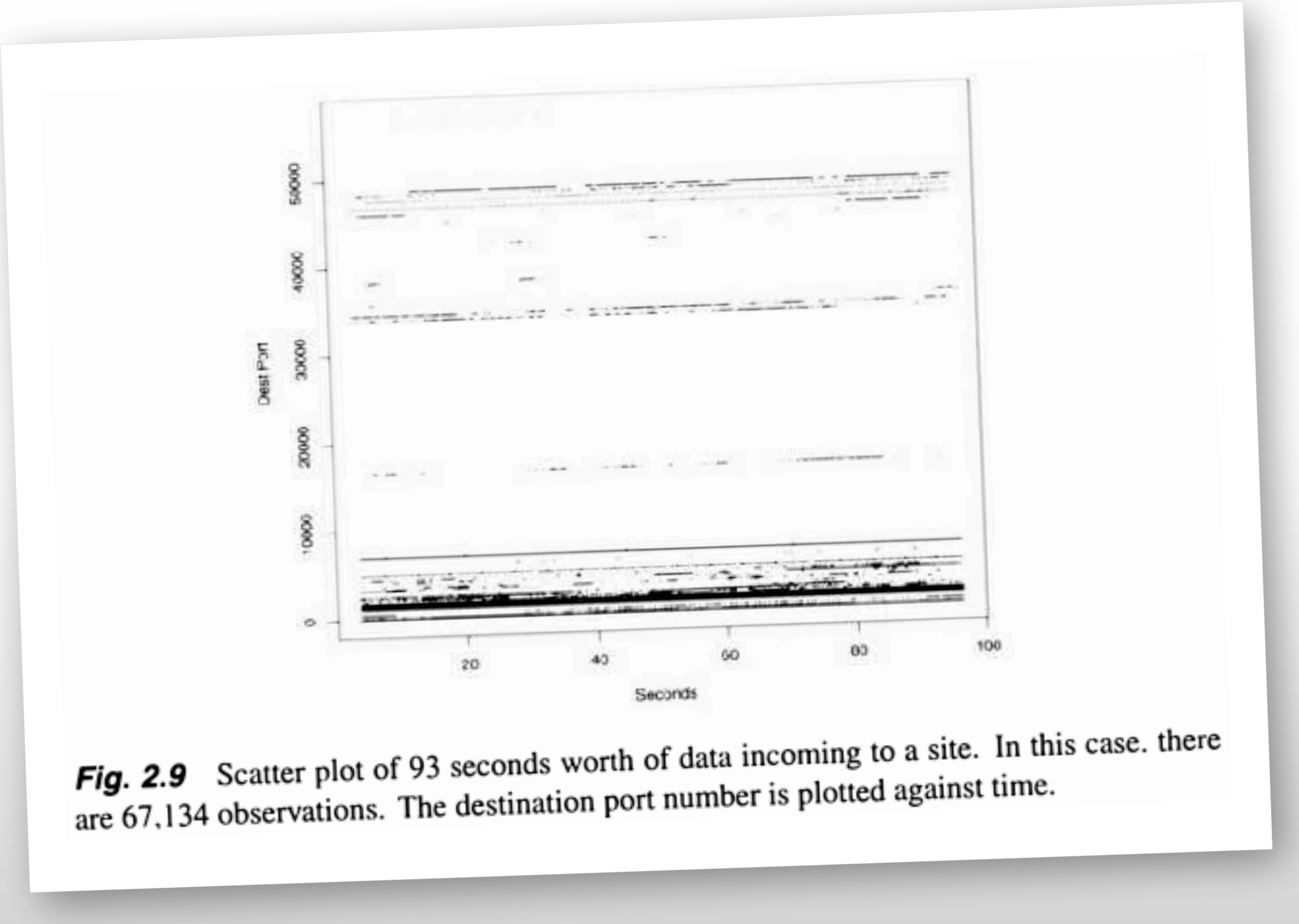


Fig. 2.9 Scatter plot of 93 seconds worth of data incoming to a site. In this case. there are 67,134 observations. The destination port number is plotted against time.

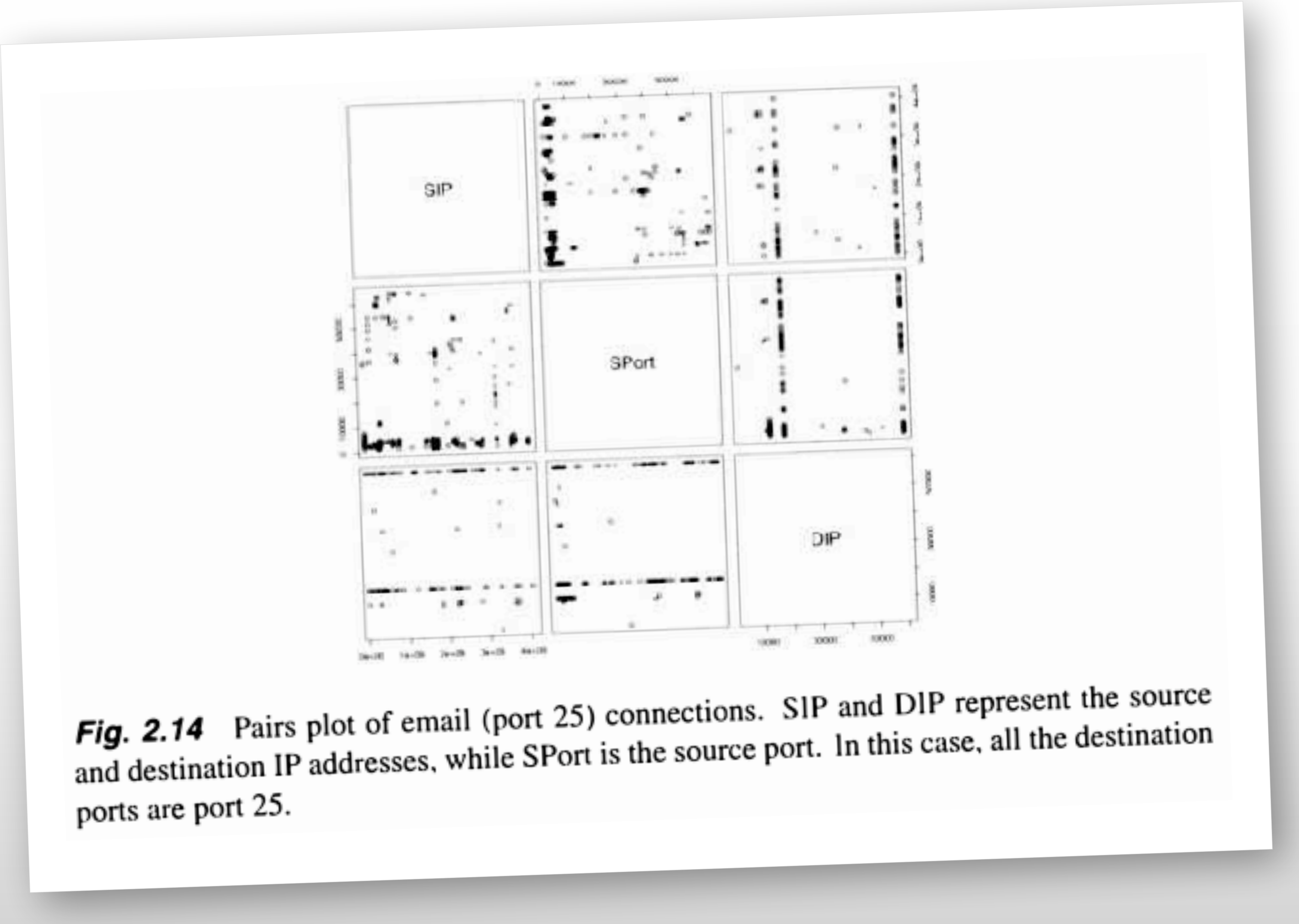


Fig. 2.14 Pairs plot of email (port 25) connections. SIP and DIP represent the source and destination IP addresses, while SPort is the source port. In this case, all the destination ports are port 25.

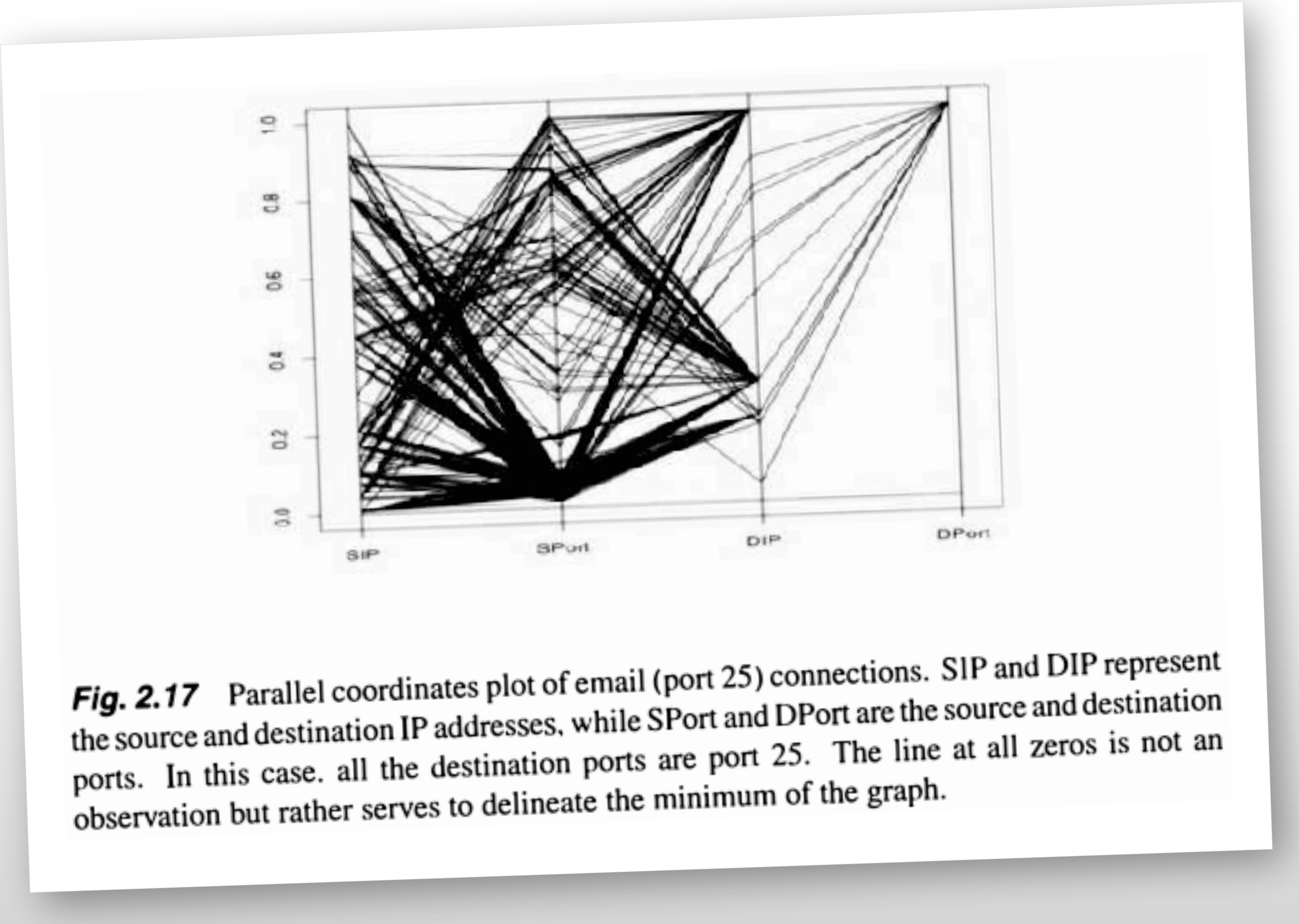


Fig. 2.17 Parallel coordinates plot of email (port 25) connections. SIP and DIP represent the source and destination IP addresses, while SPort and DPort are the source and destination ports. In this case. all the destination ports are port 25. The line at all zeros is not an observation but rather serves to delineate the minimum of the graph.

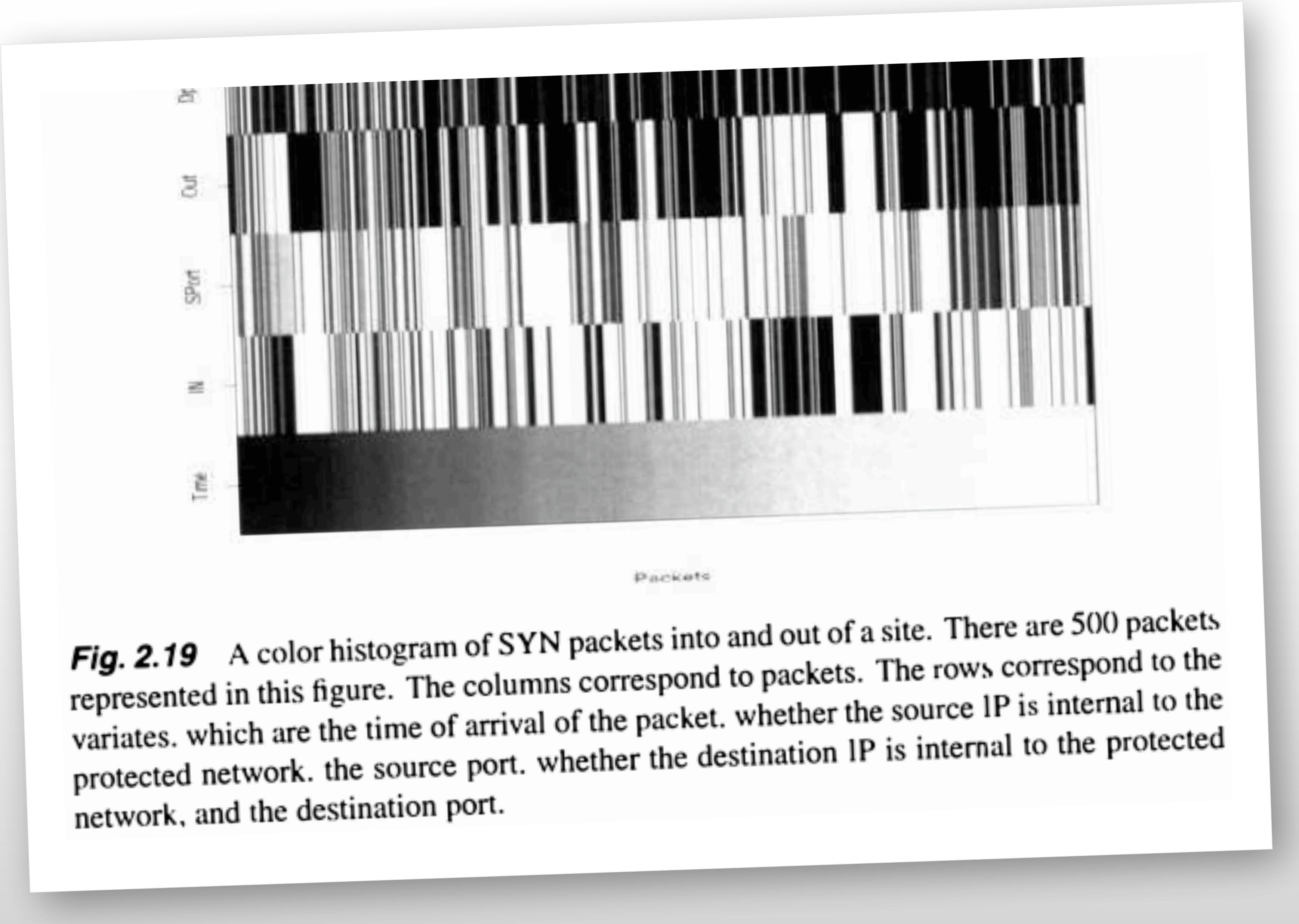
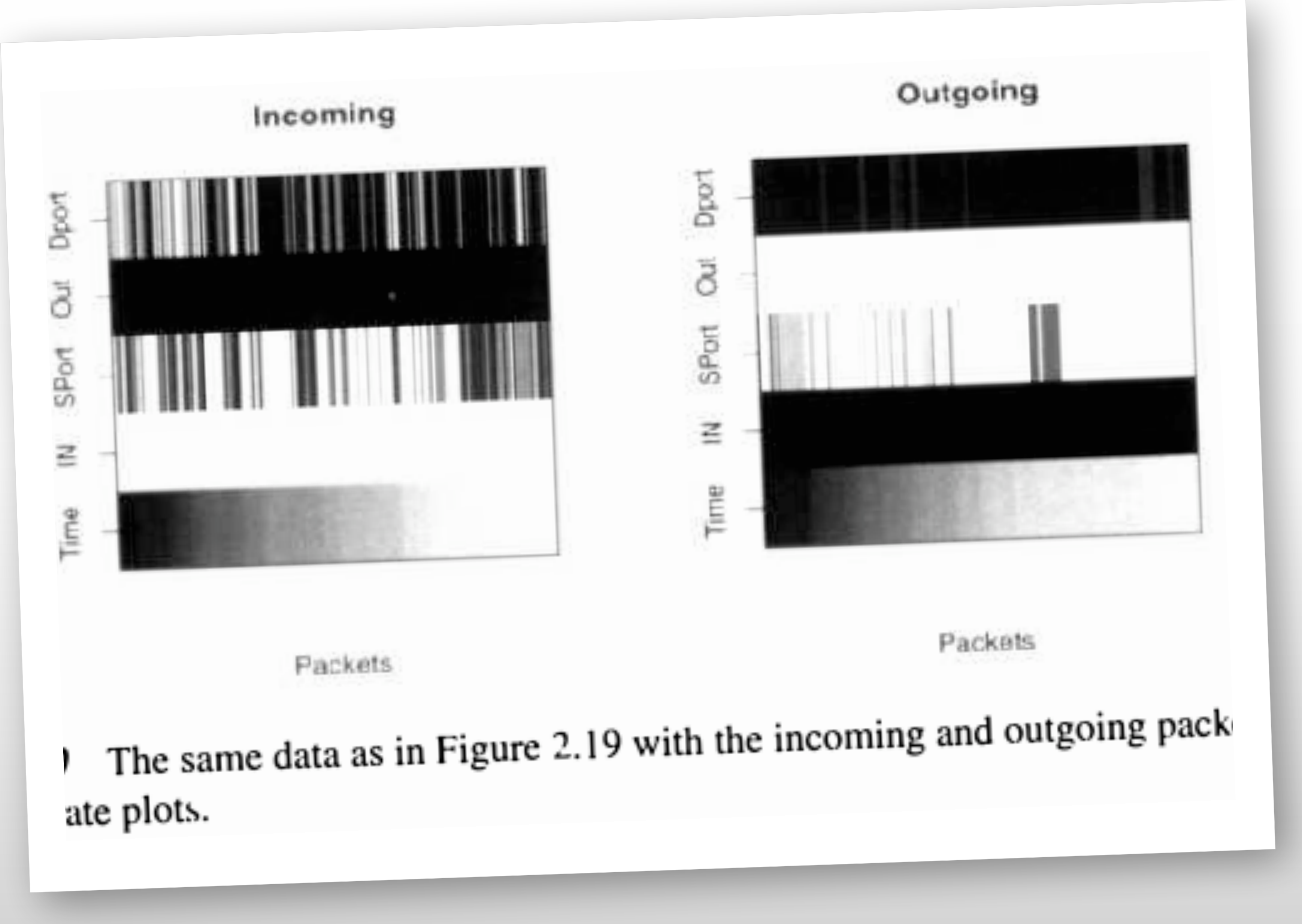
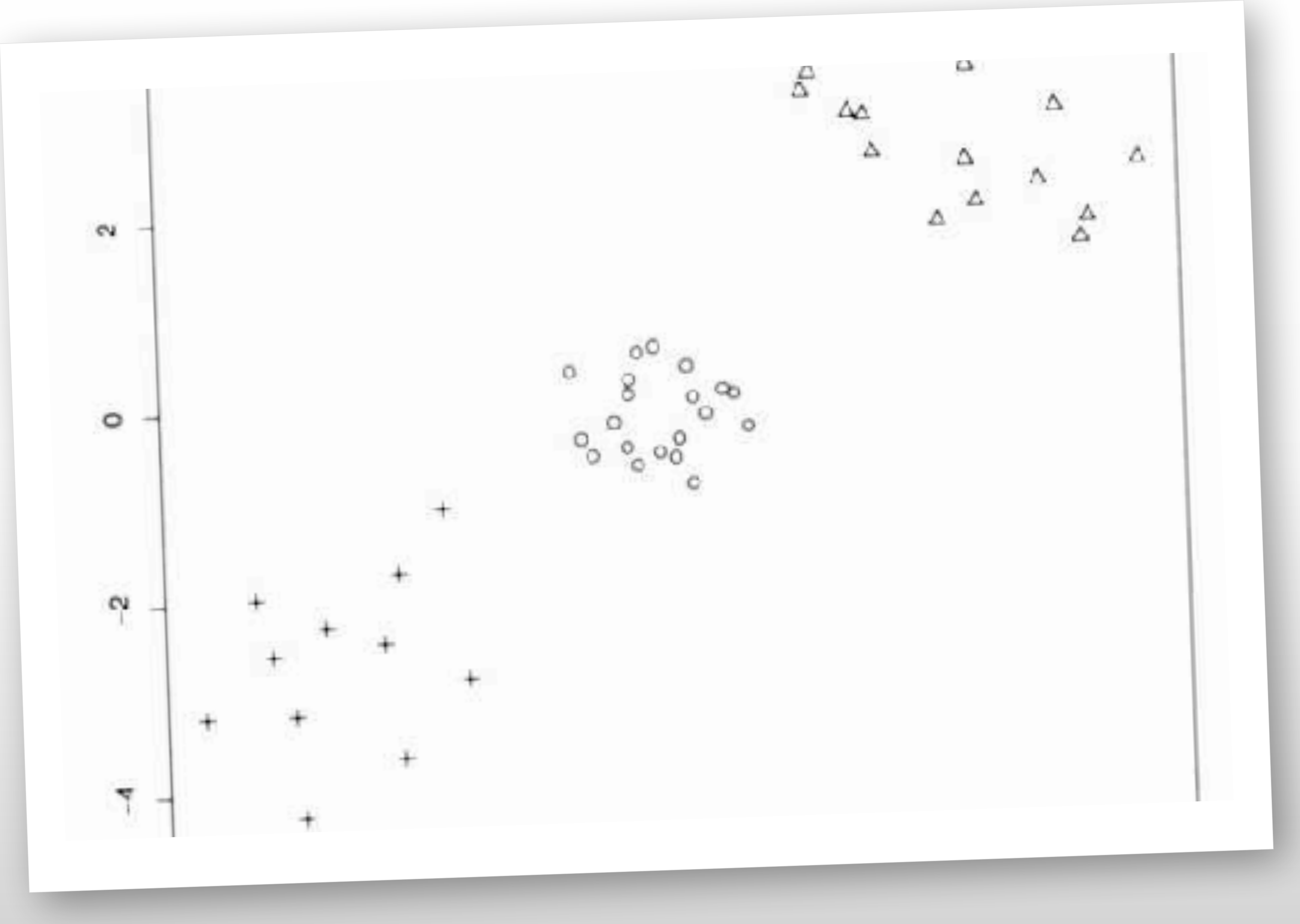


Fig. 2.19 A color histogram of SYN packets into and out of a site. There are 500 packets represented in this figure. The columns correspond to packets. The rows correspond to the variates. which are the time of arrival of the packet. whether the source IP is internal to the protected network. the source port. whether the destination IP is internal to the protected network, and the destination port.



) The same data as in Figure 2.19 with the incoming and outgoing pack

ate plots.


2
0
-2
-4

Audit Browsing

- Goal of browser: present log information in a form easy to understand and use
- Several reasons to do this:
  - Audit mechanisms may miss problems that auditors will spot
  - Mechanisms may be unsophisticated or make invalid assumptions about log format or meaning
  - Logs usually not integrated; often different formats, syntax, etc.

Browsing Techniques

Text display

Does not indicate relationships between events

✓ Hypertext display

Indicates local relationships between events

Does not indicate global relationships clearly

Relational database browsing

DBMS performs correlations, so auditor need not know in advance what associations are of interest

Preprocessing required, and may limit the associations DBMS can make

More Browsing Techniques

Replay

Shows events occurring in order; if multiple logs, intermingles entries

Graphing

Nodes are entities, edges relationships

Often too cluttered to show everything, so graphing selects subsets of events

Slicing

Show minimum set of log events affecting object

Focuses on local relationships, not global ones

Example: Visual Audit Browser

Frame Visualizer

Generates graphical representation of logs

Movie Maker

Generates sequence of graphs, each event creating a new graph suitably modified

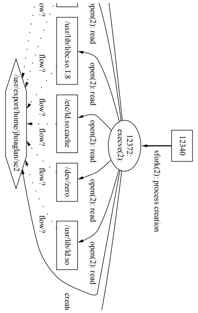
Hypertext Generator

Produces page per user, page per modified file, summary and index pages

Focused Audit Browser

Enter node name, displays node, incident edges, and nodes at end of edges

Example Use



- File changed
  - Use focused audit browser
    - Changed file is initial focus
    - Edges show which processes have altered file
  - Focus on suspicious process
    - Iterate through nodes until method used to gain access to system determined
- Question: is masquerade occurring?
  - Auditor knows audit UID of attacker

Tracking Attacker

- Use hypertext generator to get all audit records with that UID
  - Now examine them for irregular activity
  - Frame visualizer may help here
  - Once found, work forward to reconstruct activity
- For non-technical people, use movie maker to show what happened
  - Helpful for law enforcement authorities especially!