Analysis of FRAUD network ACTIONS; rules and models for detecting fraud activities

Eren Golge

FRAUD ?

- HACKERS !!
  - *DoS:* Denial of service
  - R2L: Unauth. Access
  - **U2R:** Root access to Local Machine.
  - **Probing:** Survallience.
  - ....

    First PC Viruses: Prophets of PC Viruses

- Our Case : **Fraud = Anomally**

EXPECTATIONs ?

- concise data ?

  (Feature Weightining and Selection)
- model for classification?

  (Model Extration)
- what indicates fraud ?

  (Rule extraction)

Constraints

DATA

- HUGE
- UNBALANCED
- HIGH DIMESNIONAL
- NOMINAL+NUMERICAL FEATs
- DIVERGENT CASES

MACHINE

- MEMORY
- TIME

DATA

- **KDD'99** Data
  - TCP dump on LAN -> Lincoln Labs-MIT **
  - Peppered with attacks.
  - 4 million records (reduction need)
  - 42 Features (selection need)
  - 16 Attact Type (merge as anomaly)

*Deficient for detecting content based attackts.**

* Kayacık, H. G., Zincir-heywood, A. N., & Heywood, M. I. (n.d.). Selecting Features for Intrusion Detection : A Feature Relevance Analysis on KDD 99 Intrusion Detection Datasets, 3–8.

** http://www.ll.mit.edu/

Instances

- **Instance** – 2 sec snapshots of connection.
  - **Connection** – TCP packages between same two IPs
- Each instance ~ 80 bytes

43 features

- **Categories**
  - Host
  - Service (Telnet, Voip)
  - Content
    - Exp. Root access, Shell activation
  - TCP stats
  - Destination host

  - 4 nominal vs 39 numeric features

TCP stats

feature name	*type*
duration	continuous
protocol_type	discrete
service	discrete
src_bytes	continuous
dst_bytes	continuous
flag	discrete
land	discrete
wrong_fragment	continuous
urgent	continuous

Content Features

feature name	*type*
hot	continuous
num_failed_logins	continuous
logged_in	discrete
num_compromised	continuous
root_shell	discrete
su_attempted	discrete
num_root	continuous
num_file_creations	continuous
num_shells	continuous
num_access_files	continuous
num_outbound_cmds	continuous
is_hot_login	discrete
is_guest_login	discrete

Same Host & Same Service

feature name	*type*
count	continuous
serror_rate	continuous
rerror_rate	continuous
same_srv_rate	continuous
diff_srv_rate	continuous
srv_count	continuous
srv_serror_rate	continuous
srv_rerror_rate	continuous
srv_diff_host_rate	continuous

Destination Host

@attribute '**dst_host_count**' real

@attribute '**dst_host_srv_count**' real

@attribute '**dst_host_same_srv_rate**' real

@attribute '**dst_host_diff_srv_rate**' real

@attribute '**dst_host_same_src_port_rate**' real

@attribute **'dst_host_srv_diff_host_rate**' real

@attribute '**dst_host_serror_rate**' real

@attribute '**dst_host_srv_serror_rate**' real

@attribute '**dst_host_rerror_rate**' real

@attribute **'dst_host_srv_rerror_rate**' real

All List

- duration: continuous.
- protocol_type: symbolic.
- service: symbolic.
- flag: symbolic.
- src_bytes: continuous.
- dst_bytes: continuous.
- land: symbolic.
- wrong_fragment: continuous.
- urgent: continuous.
- hot: continuous.
- num_failed_logins: continuous.
- logged_in: symbolic.
- num_compromised: continuous.
- root_shell: continuous.
- su_attempted: continuous.
- num_root: continuous.
- num_file_creations: continuous.
- num_shells: continuous.
- num_access_files: continuous.
- num_outbound_cmds: continuous.
- is_host_login: symbolic.
- is_guest_login: symbolic.
- count: continuous.
- srv_count: continuous.
- serror_rate: continuous.
- srv_serror_rate: continuous.
- rerror_rate: continuous.
- srv_rerror_rate: continuous.
- same_srv_rate: continuous.
- diff_srv_rate: continuous.
- srv_diff_host_rate: continuous.
- dst_host_count: continuous.
- dst_host_srv_count: continuous.
- dst_host_same_srv_rate: continuous.
- dst_host_diff_srv_rate: continuous.
- dst_host_same_src_port_rate: continuous.
- dst_host_srv_diff_host_rate: continuous.
- dst_host_serror_rate: continuous.
- dst_host_srv_serror_rate: continuous.
- dst_host_rerror_rate: continuous.
- dst_host_srv_rerror_rate: continuous.
- class : symbolic

4 symbolic (nominal) + 39 numeric

Path to Gotcha Moment

Packages used:

- Rapidminer
- Knime
- Weka
- R

Best performance on slide

- **Step1:** Preprocessing
  - Normalization – Scaling
  - Class merging
  - Data Filtering
- **Step2**: Feat. Selection
  - Genetic Algorithm (Out of curiosity)
  - Backward Selection
  - Supervision
- **Step3**: Model Learning
  - Xval. Parameter optimizing
    - **SVM**
    - **Neural Net.**
    - **Naive Bayes**
    - **Decision Trees**
- **Step4**: Rule Learning
  - Decision Trees
  - Naive Bayes
- **Step5**: Eval. Of Anomaly detection
- **Get Useless Results by LOF (not included)**
  - Compare with supervised method

General Work Iteration
Run the code
Go out, waste time
Come back

Step1: Clear Data

- **Remove redundant data**
  - **4,898,431 -> 1,074,991** (742MB -> 18.7MB)
    - Attack : 3,925,650 -> 262,178 (cause deficinecy ?)
    - Normal : 972,781 -> 812,814
      - Reduce majority effect
- **Random Selection**
  - **1,074,991 -> 125,973** (57,666 – Ano. Vs 67,388 - Normal)
    - Keep amount of fraud types same
    - Decrease payload of Normal instances (cost of Anomaly is bigger)
- **Merge class values**
  - FraudActivities (dos,probe) -> “anomaly”

Step2: Future Selection

Genetic Algorithms
vs
Backward Selection

+

ROC curves
Correlation Matrix

Step2: Feature Selection

- **Genetic algorithm :** **42 feats => 12 feats**

  - NaiveBayes( ) { returns fittness; }

PerformanceVector:
accuracy: **84.67%**
ConfusionMatrix:

True:	anomaly	normal
anomaly:	10912	1536
normal:	1921	8175

Without feature selection

PerformanceVector:
accuracy: **77.58%**
ConfusionMatrix:

True:	anomaly	normal
anomaly:	8540	761
normal:	4293	8950

Step2

- **Backward elemination** BETTER but LONGER !
  - with Naive Bayes **42 feats => 30 feats**
    - PerformanceVector:
      accuracy: **89.46%**
      ConfusionMatrix:
      True: anomaly normal
      anomaly: 12257 1800
      normal: 576 7911

IMPROVEMENT !

PerformanceVector:
accuracy: **77.58%**
ConfusionMatrix:
True: anomaly normal
anomaly:8540 761
normal: 4293 8950

Without feature selection

GENETIC ALGO. İs good if you have time constraint!
%84 Accuracy

Step2: After unsupervised selection

42 features => 31 features

SELECTED FEATURES

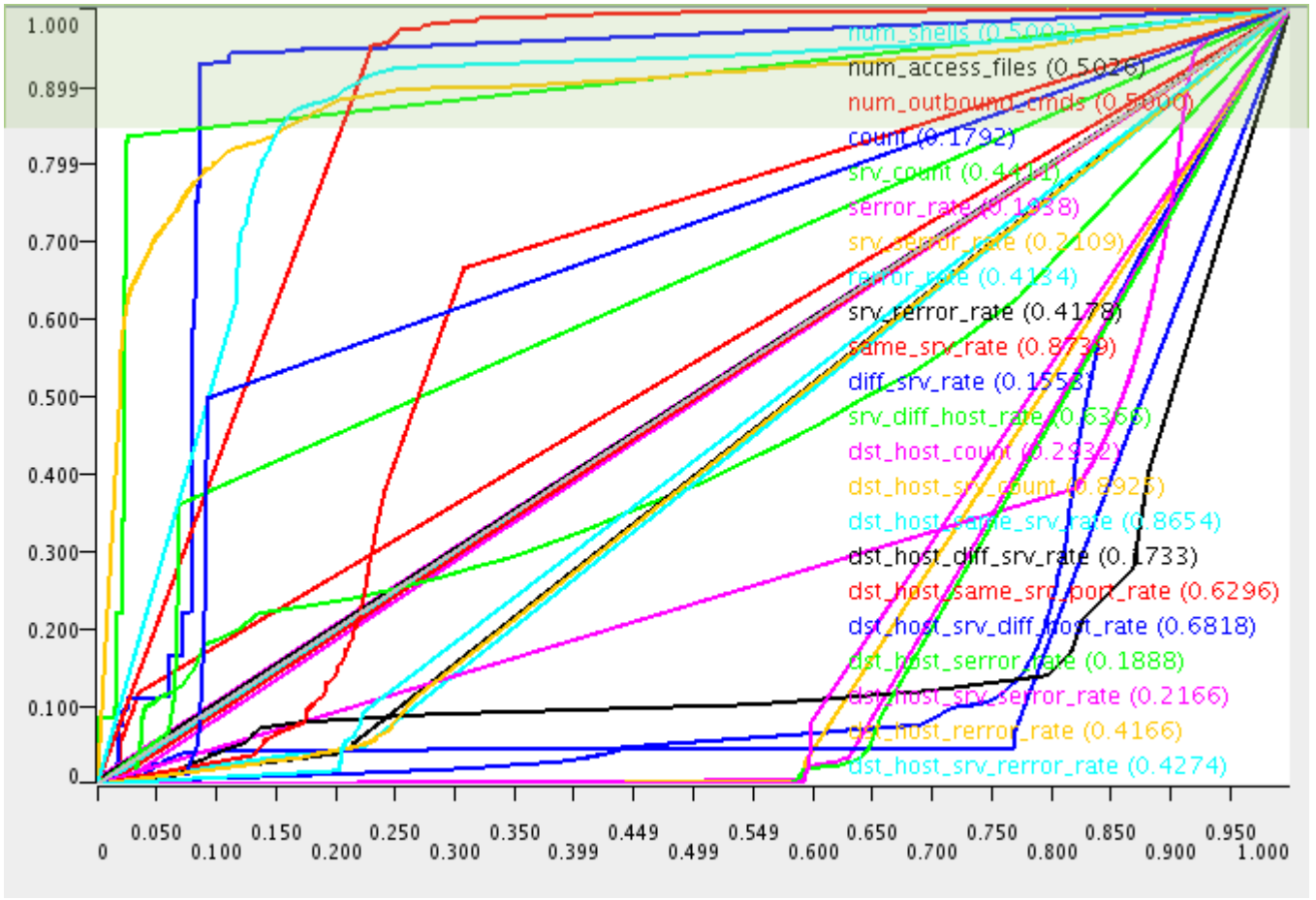
protocol_type
service
flag
land
urgent
hot
num_compromised
root_shell
su_attempted
num_root
num_file_creations
num_shells
num_access_files
num_outbound_cmds
is_guest_login
srv_count
serror_rate
srv_serror_rate
rerror_rate
srv_rerror_rate
diff_srv_rate
srv_diff_host_rate
dst_host_count
dst_host_same_srv_rate
dst_host_diff_srv_rate
dst_host_same_src_port_rate
dst_host_srv_diff_host_rate
dst_host_serror_rate
dst_host_rerror_rate
dst_host_srv_rerror_rate

Step2 with my hands

- **Support of unsupervised selection**
- **What are my constraints**
  - Correlation
    - Correlation = same information
  - ROC curves

ROC curves

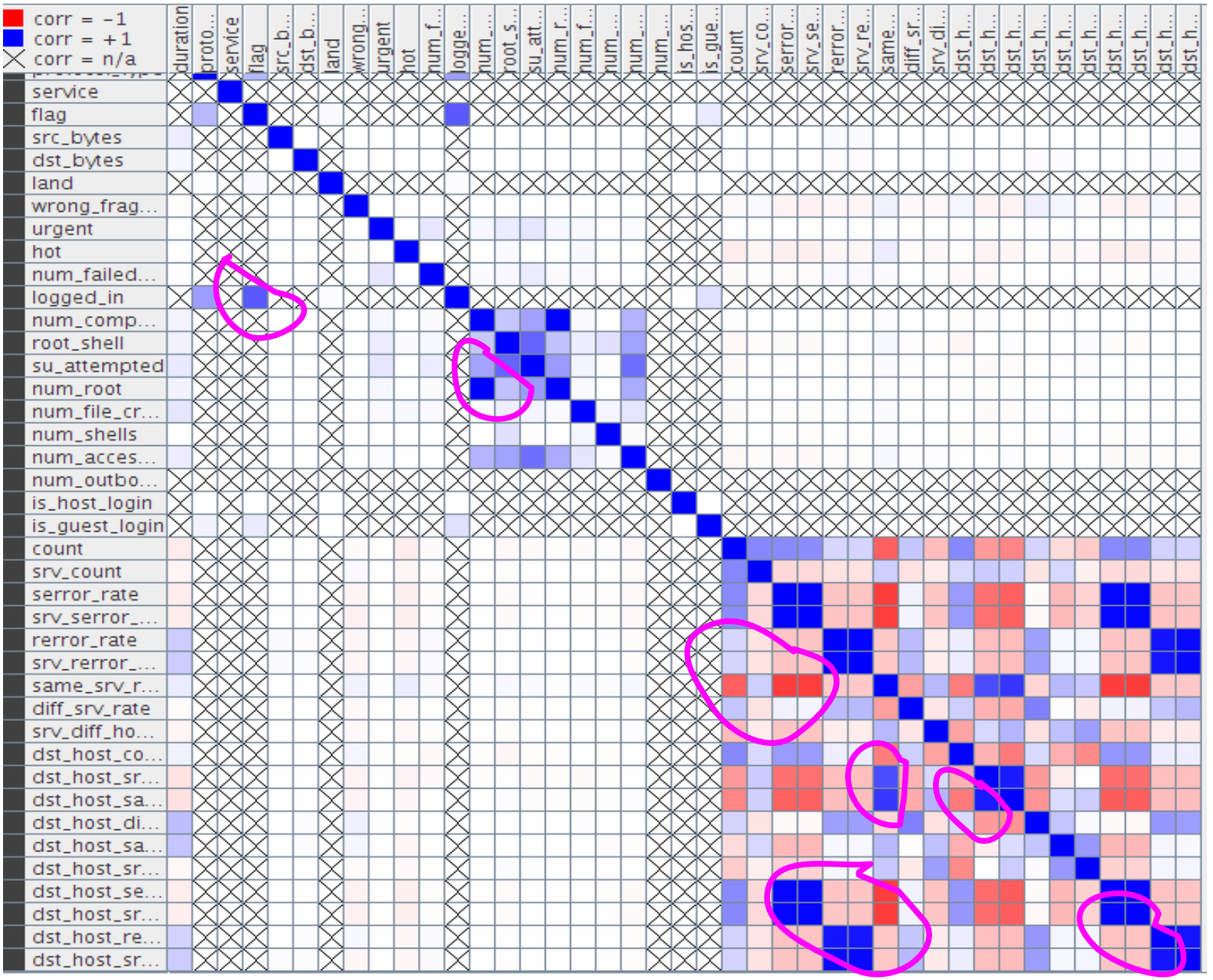
Straight Lines = Uninformative



Row ID	D Area U...
duration	0.541
src_bytes	0.899
dst_bytes	0.9
wrong_frag...	0.491
urgent	0.5
hot	0.497
num_failed...	0.5
num_comp...	0.495
root_shell	0.501
su_attempted	0.501
num_root	0.505
num_file_cr...	0.502
num_shells	0.5
num_acces...	0.503
num_outbo...	0.5
count	0.179
srv_count	0.441
serror_rate	0.194
srv_serror_...	0.211
rerror_rate	0.413
srv_rerror_...	0.418
same_srv_r...	0.874
diff_srv_rate	0.155
srv_diff_ho...	0.637
dst_host_co...	0.293
dst_host_sr...	0.892
dst_host_sa...	0.865
dst_host_di...	0.173
dst_host_sa...	0.63
dst_host_sr...	0.682
dst_host_se...	0.189
dst_host_sr...	0.217
dst_host_re...	0.417
dst_host_sr...	0.427

Correlations over 0.85

Correlation Matrix



After ROC and Correlation

42 to 35 features

Selected features ⟺

@ATTRIBUTE duration REAL
@ATTRIBUTE protocol_type REAL
@ATTRIBUTE service REAL
@ATTRIBUTE flag REAL
@ATTRIBUTE src_bytes REAL
@ATTRIBUTE dst_bytes REAL
@ATTRIBUTE land REAL
@ATTRIBUTE wrong_fragment REAL
@ATTRIBUTE urgent REAL
@ATTRIBUTE hotREAL
@ATTRIBUTE num_failed_logins REAL
@ATTRIBUTE logged_in REAL
@ATTRIBUTE num_compromised REAL
@ATTRIBUTE root_shell REAL
@ATTRIBUTE su_attempted REAL
@ATTRIBUTE num_file_creations REAL
@ATTRIBUTE num_shells REAL
@ATTRIBUTE num_access_files REAL
@ATTRIBUTE num_outbound_cmdsREAL
@ATTRIBUTE is_host_login REAL
@ATTRIBUTE is_guest_login REAL
@ATTRIBUTE count REAL
@ATTRIBUTE srv_count REAL
@ATTRIBUTE rerror_rate REAL
@ATTRIBUTE same_srv_rate REAL
@ATTRIBUTE diff_srv_rate REAL
@ATTRIBUTE srv_diff_host_rate REAL
@ATTRIBUTE dst_host_countREAL
@ATTRIBUTE dst_host_diff_srv_rate REAL
@ATTRIBUTE dst_host_same_src_port_rate REAL
@ATTRIBUTE dst_host_srv_diff_host_rateREAL

Naive Bayes after Supervised Filtering

=== Cross-Val Results ===

Correctly Classified Instances	117594	**93.3486 %**
Incorrectly Classified Instances	8379	6.6514 %

a	b	<-- classified as
64341	3002	a = normal
5377	53253	b = anomaly

Clues of divergency of test datas !

Overfitting!

=== Test Set Results ===

Correctly Classified Instances	16811	**74.5697 %**
Incorrectly Classified Instances	5733	25.4303 %

a	b	<-- classified as
9040	671	a = normal
5062	7771	b = anomaly

Final Atributes

42 to 29 features

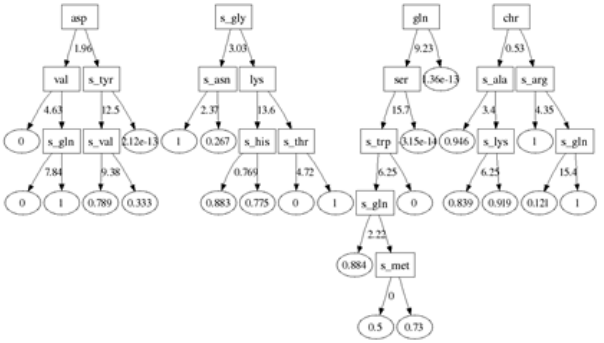
Backward selec: -11
ROC+Corre. Mat: -2

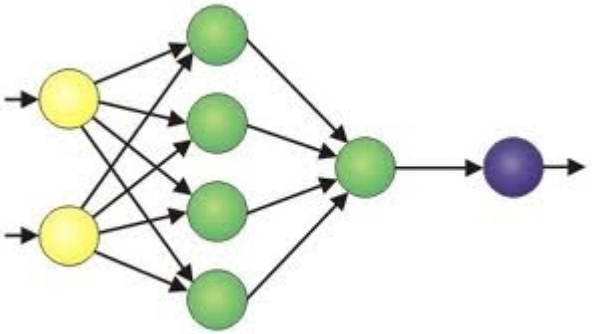
Selected features ⟺

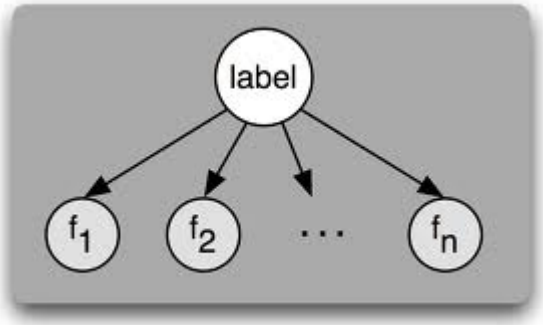
protocol_type
service
flag
land
urgent
hot
num_compromised
root_shell
su_attempted
num_root
num_file_creations
num_shells
num_access_files
num_outbound_cmds
is_guest_login
srv_count
serror_rate
srv_rerror_rate
diff_srv_rate
srv_diff_host_rate
dst_host_count
dst_host_same_srv_rate
dst_host_diff_srv_rate
dst_host_same_src_port_rate
dst_host_srv_diff_host_rate
dst_host_serror_rate
dst_host_rerror_rate
dst_host_srv_rerror_rate

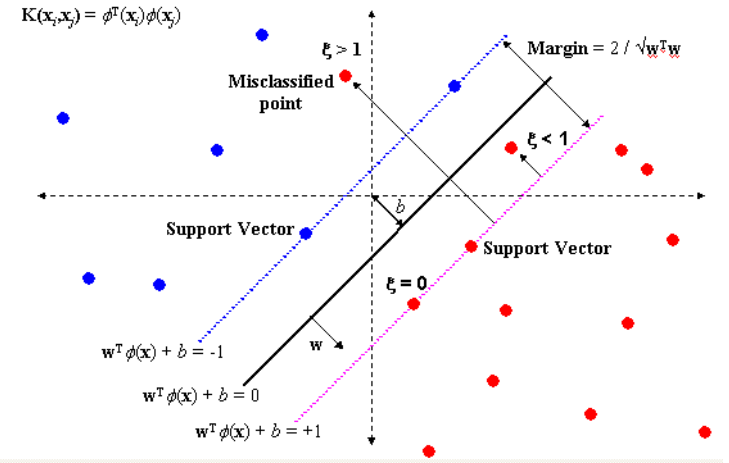
Step3: Model Learning

- Random Forests
- Neural Nets
- SVM with RBF
- Naive Bayes









Step3: Model Learning

- Random Forest
  - Enhanced to overfitting
  - No X-Val requirement

```
=== Summary ===

Correctly Classified Instances       18503               82.0751 %
Incorrectly Classified Instances      4041               17.9249 %

=== Confusion Matrix ===

   a    b   <-- classified as
 9022  689 |    a = normal
 3352 9481 |    b = anomal
```

Caruana, R., & Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. Proceedings of the 23rd international conference on Machine learning - ICML '06, 161–168. doi:10.1145/1143844.1143865

Step3: Model Learning

- Neural Nets
  - 10% of train data
  - (# classes+# attributes)/2 = # hidden units
  - Optimized Parameters
    - Momentum = 0.1
    - Learning rate = 0.2
    - Decay
  - Long, so long to train....

Accuracy: **84.08%**

Step3: Model Learning

- **SVM – libSVM with RBF kernel**
  - Numeratize nominal values.
  - Long training time

PerformanceVector:
accuracy: **75.78%**
ConfusionMatrix:

True:	anomaly	normal
anomaly:	10342	2969
normal:	2491	6742

Hsu, C., Chang, C., & Lin, C. (2003). A practical guide to support vector classification, 1(1), 1–16. Retrieved from https://www.cs.sfu.ca/people/Faculty/teaching/726/spring11/svmguide.pdf

Step3

- **Naive Bayes**
  - Probabilistic approach
  - Analitical results
  - Also gives probabilistic rules

GOTCHA!

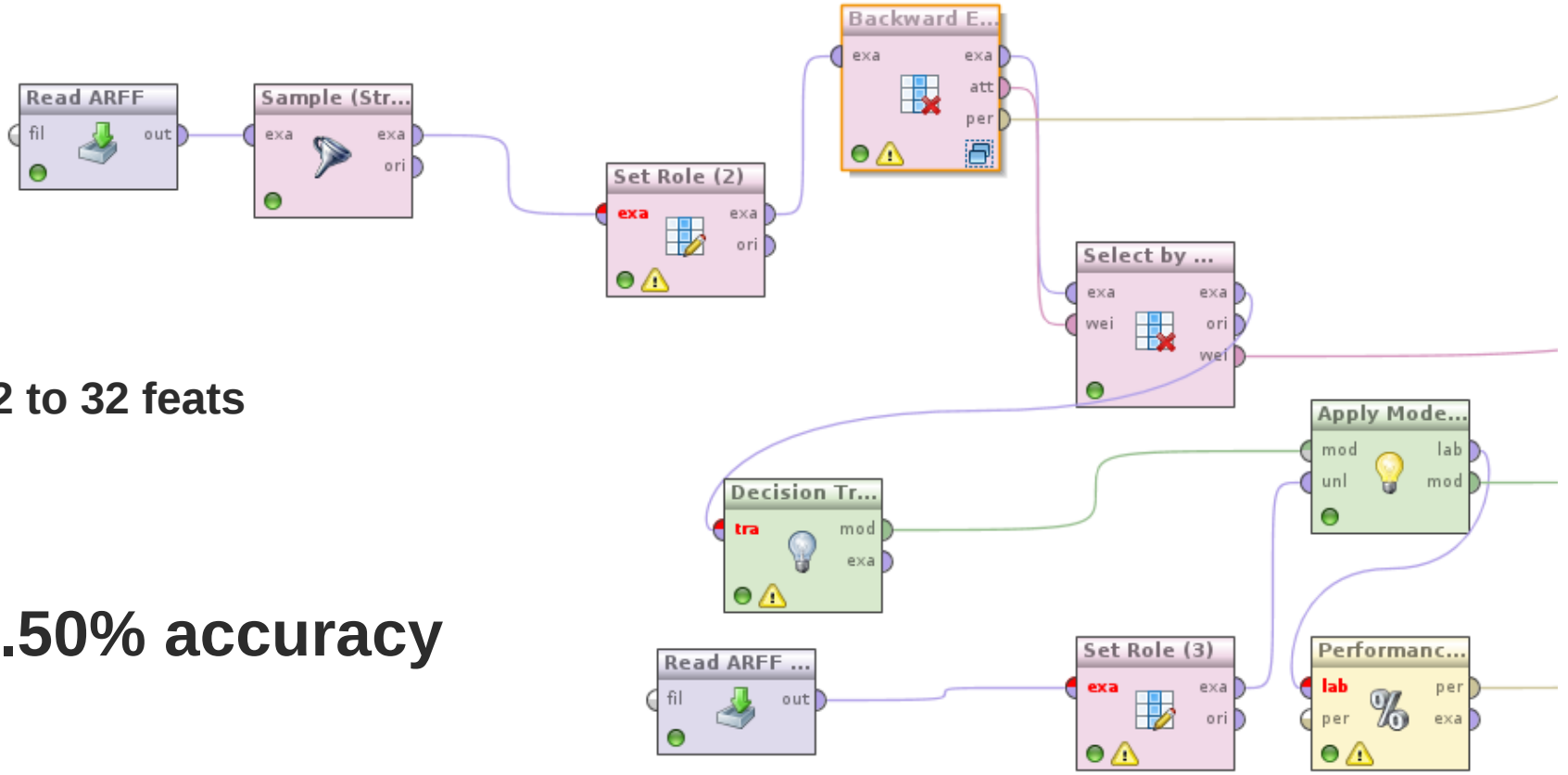
PerformanceVector:
accuracy: **89.46%**
ConfusionMatrix:

True:	anomaly	normal
anomaly:	12257	1800
normal:	576	7911

Step4: Rule Induction

Decision Tree + Naive Bayes

Step4: Backward Selection + Grid Search+Decision Tree



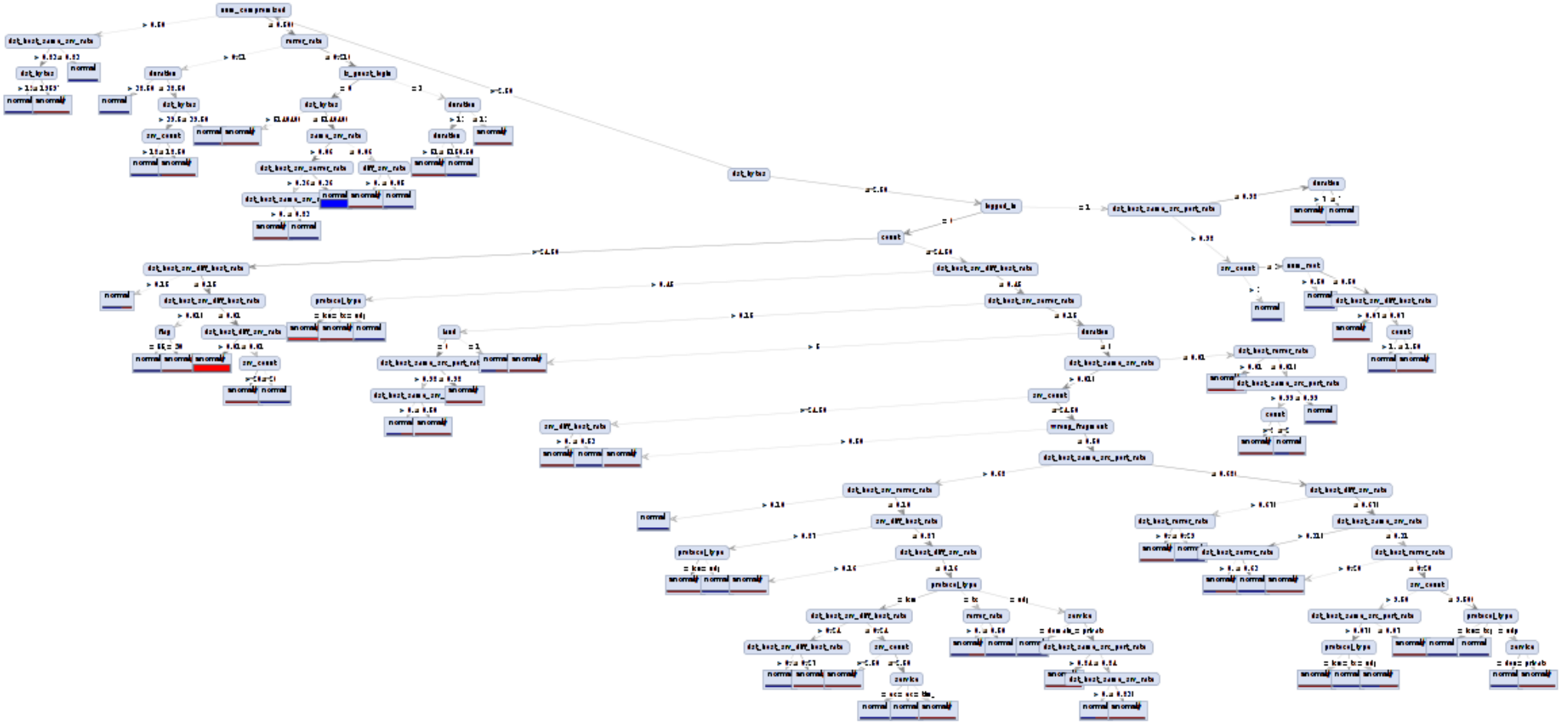
=> 42 to 32 feats

85.50% accuracy

accuracy: 85.50%

	true anomaly	true normal	class precision
pred. anomaly	9866	303	97.02%
pred. normal	2967	9408	76.02%
class recall	76.88%	96.88%	

Tree Over Overview



Top Nodes

```
is_host_login = 0
|   flag = OTH
|   |   protocol_type = tcp
|   flag = REJ
|   |   protocol_type = tcp
|   |   |   land = 0
|   flag = RSTOS0
|   |   protocol_type = tcp
|   |   |   land = 0\
|   flag = RSTOS0: anomaly {normal=0, anomaly=10}
|   flag = RSTR
|   |   protocol_type = tcp
|   |   |   land = 0
|   flag = S0
|   |   protocol_type = tcp
|   |   |   logged_in = 0
|   flag = S1
|   |   protocol_type = tcp
|   |   |   land = 0
|   |   |   |   logged_in = 0
|   |   |   |   logged_in = 1: normal {normal=31, anomaly=0}
|   flag = S2
|   |   protocol_type = tcp
|   |   |   land = 0
|   |   |   |   is_guest_login = 0
|   flag = S3
|   |   protocol_type = tcp
|   |   |   land = 0
|   |   |   |   logged_in = 1
|   flag = SF
|   |   land = 0
|   |   |   protocol_type = icmp
|   |   |   |   logged_in = 0
|   |   |   protocol_type = tcp
|   |   |   |   hot > 1.500
|   |   |   |   hot ≤ 1.500
|   |   |   protocol_type = udp
|   |   |   |   logged_in = 0
|   flag = SH: anomaly {normal=0, anomaly=28}
```

Most discriminative attributes

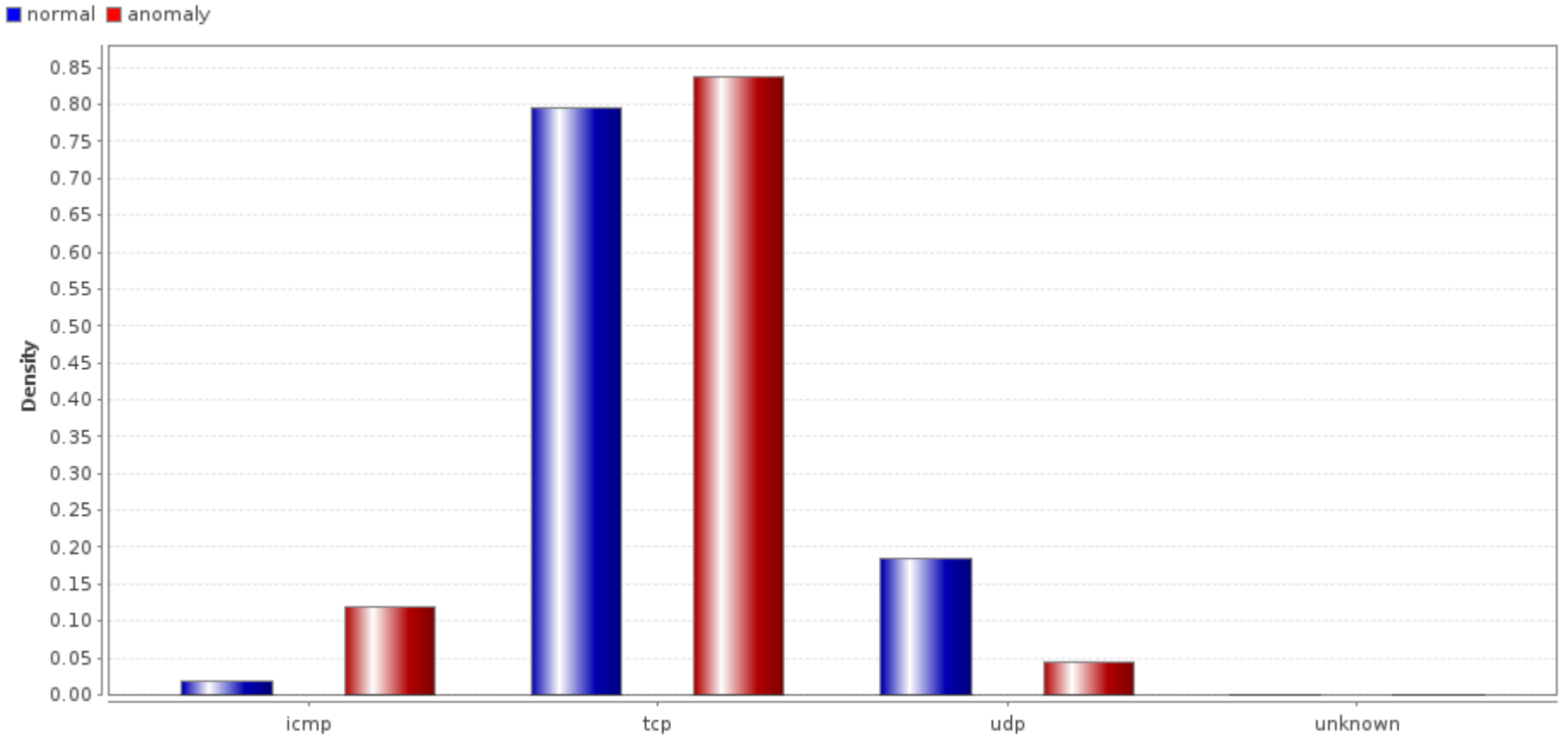
- **is_host_login**
  - ...at that particular snapshot
- **flag**
  - Package header flag
- **protocol_type**
  - Protocol of connetion

Step4: Naive Bayes

Exampe Results

Accuracy 89.43%

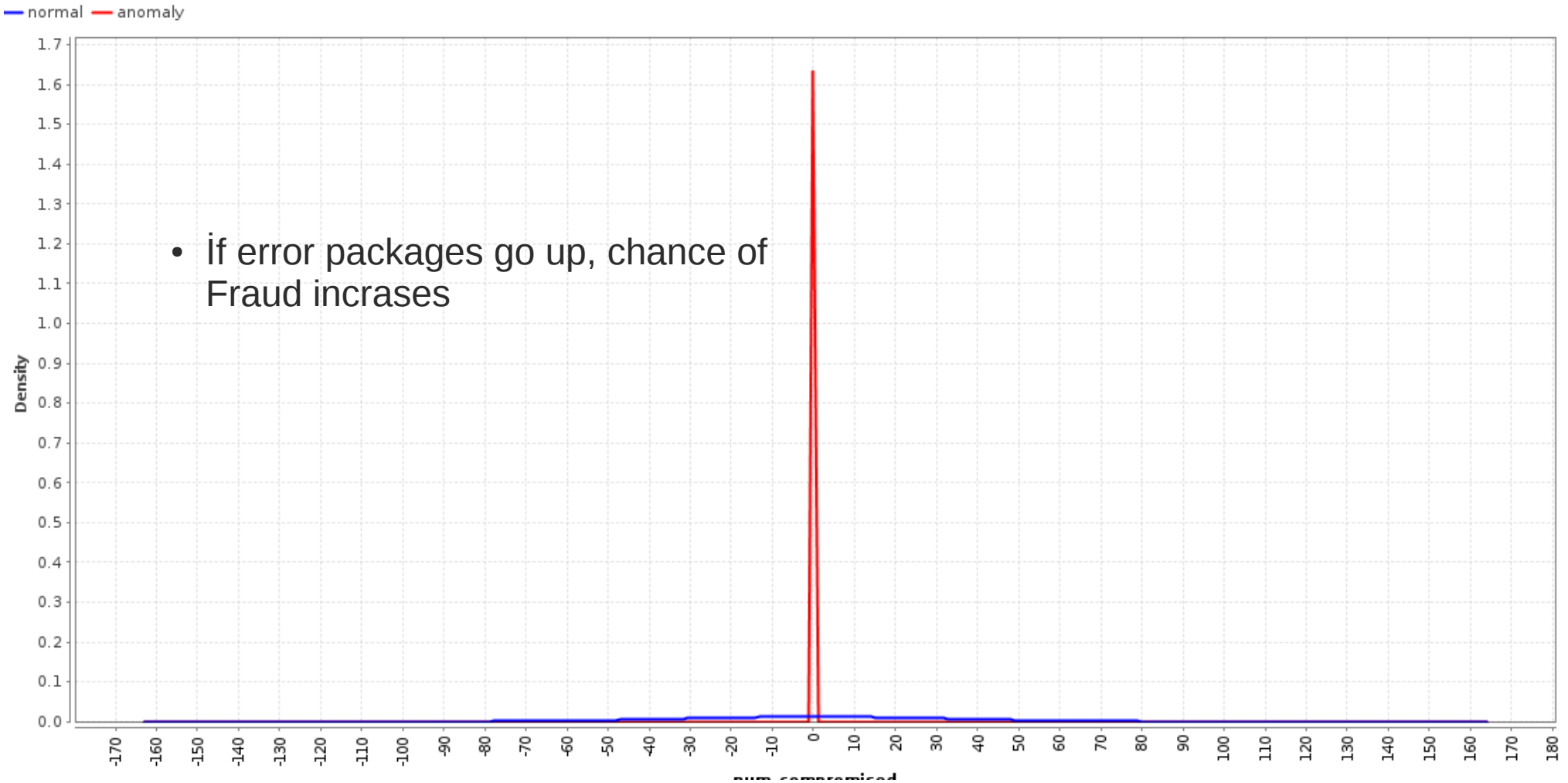
PROTOCOL EFFECT



Step4: Naive Bayes

Accuracy 89.43%

 # COMPROMISE CONDITIONS



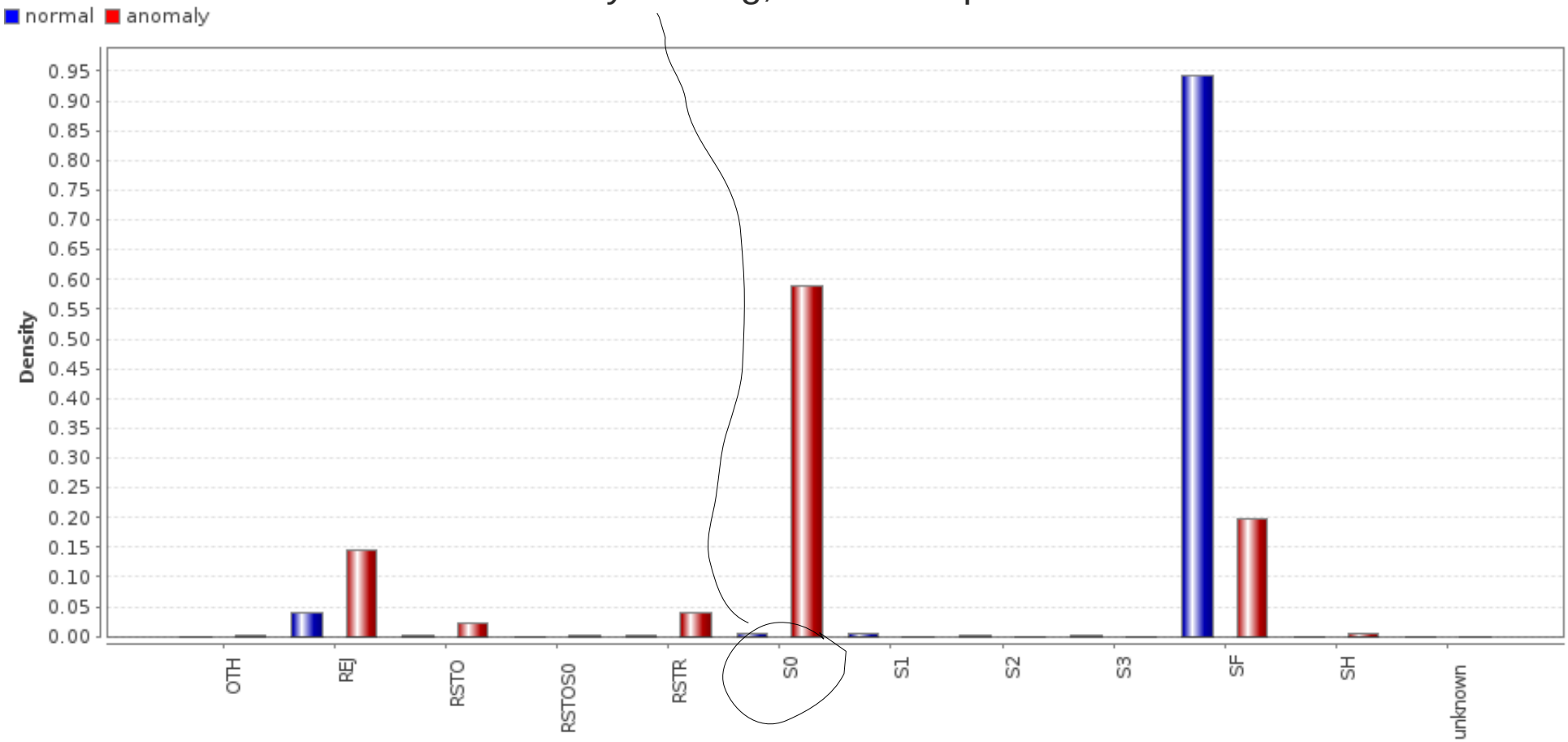
- İf error packages go up, chance of Fraud incrases

Step4: Naive Bayes

Accuracy 89.43%

FLAGS

- If so much Sync0 flag, increased prob of fraud



Gotchas

- **Naive Bayes**
  - Poor Machine + Less Effort + Naive Bayes = **Best Model**

- **Best indicators of fraud**
  - **Protocols** (monitoring icmp packages)
  - Log **compromised conditions** (larger # is secure)
  - Log **SNY** packages.

What could or will be done?

- Test selected features with other algorithms as well
- Craft a new algorithm for divergency
- Wire a software and see the real time perofrmance.
- Run a accociation algorithm.

ANY QUESTION & COMMENT & SAYING & WORD & SOMETHING?