Data Mining and Predictive Analytics : Project 2

Dr. Eitel Lauria

November 11th, 2020

Shashank Kala

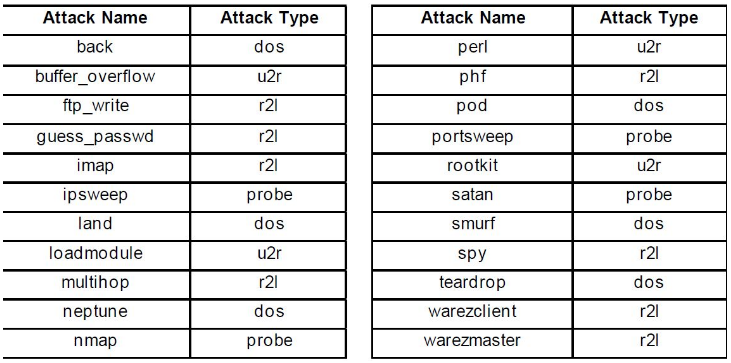
Sindhoori Kotapati

Table of Contents:

Executive Summary:

Network Intrusions are unwarranted penetrations to a system that can be either be identified as passive or active. These intrusions can be malicious in nature, and may extract confidential company information or resources for other uses. Typically, the passive intrusions interfere with the system without detection, and the active intrusions cause noticeable alterations to the system. Therefore, it is important to implement a solid network intrusion system that can detect unauthorized access by identifying signs of suspicious activity that can be handled by the system administrator. [1] The 4 main attacks that we will be focusing on are denial-of-service (DOS) attacks, R2L attacks which is an unauthorized access from a remote machine, U2R attacks which is unauthorized access to local superuser privileges and lastly probing which is the surveillance and other probing techniques.

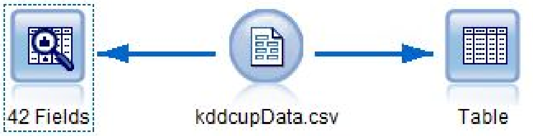
The objective of this analysis is to understand and apply classification methods using SPSS that will be used to discriminate between “normal” connections” and network intrusions or “attacks" and measure predictive importance. We have used a reduced subset of the kddcup data set which was explored using a data audit node to visualize outliers and missing data. The target variable, *connection type* was reclassified into “normal” and various “attacks' ' through SPSS reclassify node, which are shown below. The classification methods we have selected to implement and analyze include the Bayesian Network, Logistic Regression, Artificial Neural Network (ANN) and Decision Tree. The accuracy of all of these models were calculated in order to indicate which one of these classification models is the most appropriate to utilize.

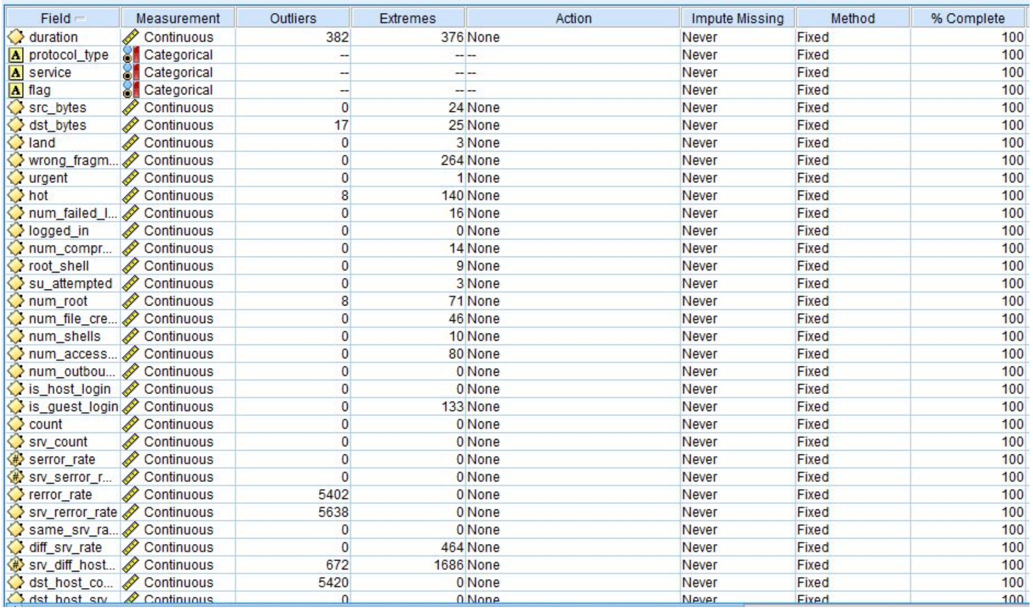


Attack Name	Attack Type
back	dos
buffer_overflow	u2r
ftp_write	r2l
guess_passwd	r2l
imap	r2l
ipsweep	probe
land	dos
loadmodule	u2r
multihop	r2l
neptune	dos
nmap	probe
perl	u2r
phf	r2l
pod	dos
portsweep	probe
rootkit	u2r
satan	probe
smurf	dos
spy	r2l
teardrop	dos
warezclient	r2l
warezmaster	r2l

Data Exploration:

Given the kddcupdata, we have explored the data set by attaching a Table Node, as well as a Data Audit Node shown below. The Data Audit node indicates that there are 42 fields.



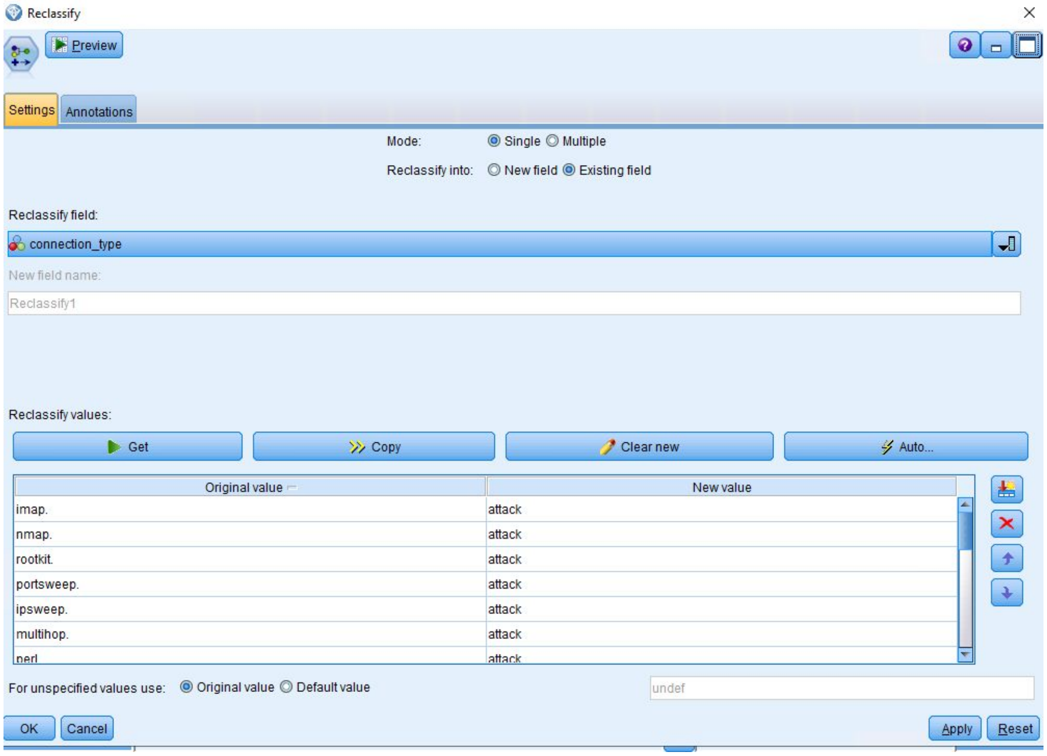


Field	Measurement	Outliers	Extremes	Action	Impute Missing	Method	% Complete
duration	Continuous	382	376	None	Never	Fixed	100
protocol_type	Categorical	--	--	--	Never	Fixed	100
service	Categorical	--	--	--	Never	Fixed	100
flag	Categorical	--	--	--	Never	Fixed	100
src_bytes	Continuous	0	24	None	Never	Fixed	100
dst_bytes	Continuous	17	25	None	Never	Fixed	100
land	Continuous	0	3	None	Never	Fixed	100
wrong_fragm...	Continuous	0	264	None	Never	Fixed	100
urgent	Continuous	0	1	None	Never	Fixed	100
hot	Continuous	8	140	None	Never	Fixed	100
num_failed_l...	Continuous	0	16	None	Never	Fixed	100
logged_in	Continuous	0	0	None	Never	Fixed	100
num_compr...	Continuous	0	14	None	Never	Fixed	100
root_shell	Continuous	0	9	None	Never	Fixed	100
su_attempted	Continuous	0	3	None	Never	Fixed	100
num_root	Continuous	8	71	None	Never	Fixed	100
num_file_cre...	Continuous	0	46	None	Never	Fixed	100
num_shells	Continuous	0	10	None	Never	Fixed	100
num_access...	Continuous	0	80	None	Never	Fixed	100
num_outbou...	Continuous	0	0	None	Never	Fixed	100
is_host_login	Continuous	0	0	None	Never	Fixed	100
is_guest_login	Continuous	0	133	None	Never	Fixed	100
count	Continuous	0	0	None	Never	Fixed	100
srv_count	Continuous	0	0	None	Never	Fixed	100
serror_rate	Continuous	0	0	None	Never	Fixed	100
srv_serror_r...	Continuous	0	0	None	Never	Fixed	100
rerror_rate	Continuous	5402	0	None	Never	Fixed	100
srv_rerror_rate	Continuous	5638	0	None	Never	Fixed	100
same_srv_ra...	Continuous	0	0	None	Never	Fixed	100
diff_srv_rate	Continuous	0	464	None	Never	Fixed	100
srv_diff_host...	Continuous	672	1686	None	Never	Fixed	100
dst_host_co...	Continuous	5420	0	None	Never	Fixed	100
dst_host_srv	Continuous	0	0	None	Never	Fixed	100

From the above table, it’s clear that the data set does not contain any missing values as all the values are at 100% completion. However there are several outliers including; duration(382),dst_bytes(17),hot(8),num_root(8),rerror_rate(5402),srv_rerror_rate(5638),srv_diff_host_rate(672),dst_host_count(5420),dst_host_srv_diff_host_rate(548),dst_host_rerror_rate(5206),dst_host_srv_rerror_rate(5123).

Reclassification:

Shown below is the re-coding of the target variable or connection type via the Reclassify Node into attack/normal:



Mode: ● Single ○ Multiple

Reclassify into: ○ New field ● Existing field

Reclassify field: connection_type

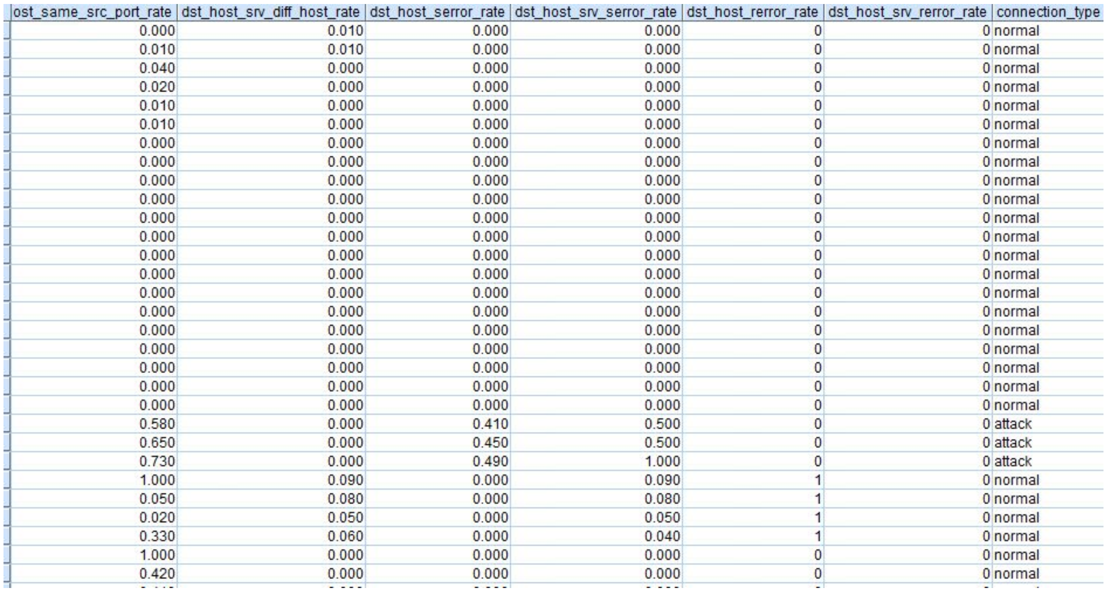
New field name: Reclassify1

Reclassify values:

Original value	New value
imap.	attack
nmap.	attack
rootkit.	attack
portsweep.	attack
ipsweep.	attack
multihop.	attack
perl	attack

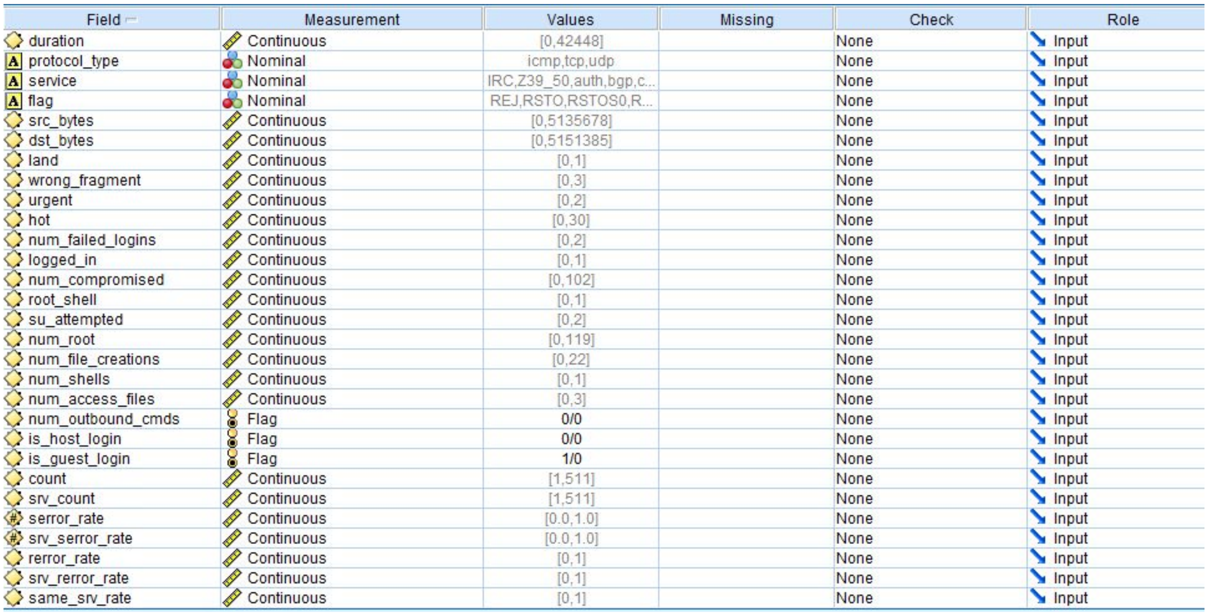
For unspecified values use: ● Original value ○ Default value undef

Shown below is the dataset after it has been reclassified as normal and attack:



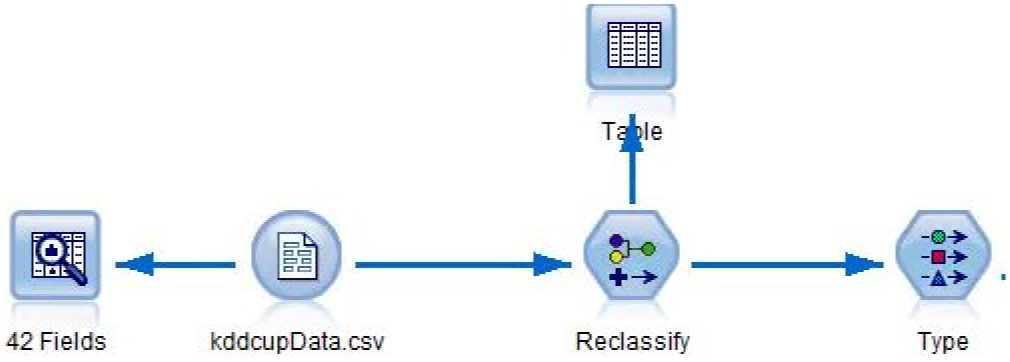
ost_same_src_port_rate	dst_host_srv_diff_host_rate	dst_host_serror_rate	dst_host_srv_serror_rate	dst_host_rerror_rate	dst_host_srv_rerror_rate	connection_type
0.000	0.010	0.000	0.000	0	0	normal
0.010	0.010	0.000	0.000	0	0	normal
0.040	0.000	0.000	0.000	0	0	normal
0.020	0.000	0.000	0.000	0	0	normal
0.010	0.000	0.000	0.000	0	0	normal
0.010	0.000	0.000	0.000	0	0	normal
0.000	0.000	0.000	0.000	0	0	normal
0.000	0.000	0.000	0.000	0	0	normal
0.000	0.000	0.000	0.000	0	0	normal
0.000	0.000	0.000	0.000	0	0	normal
0.000	0.000	0.000	0.000	0	0	normal
0.000	0.000	0.000	0.000	0	0	normal
0.000	0.000	0.000	0.000	0	0	normal
0.000	0.000	0.000	0.000	0	0	normal
0.000	0.000	0.000	0.000	0	0	normal
0.000	0.000	0.000	0.000	0	0	normal
0.000	0.000	0.000	0.000	0	0	normal
0.000	0.000	0.000	0.000	0	0	normal
0.000	0.000	0.000	0.000	0	0	normal
0.000	0.000	0.000	0.000	0	0	normal
0.000	0.000	0.000	0.000	0	0	normal
0.580	0.000	0.410	0.500	0	0	attack
0.650	0.000	0.450	0.500	0	0	attack
0.730	0.000	0.490	1.000	0	0	attack
1.000	0.090	0.000	0.090	1	0	normal
0.050	0.080	0.000	0.080	1	0	normal
0.020	0.050	0.000	0.050	1	0	normal
0.330	0.060	0.000	0.040	1	0	normal
1.000	0.000	0.000	0.000	0	0	normal
0.420	0.000	0.000	0.000	0	0	normal

Shown below is the changing of the data to a proper format using the type node:

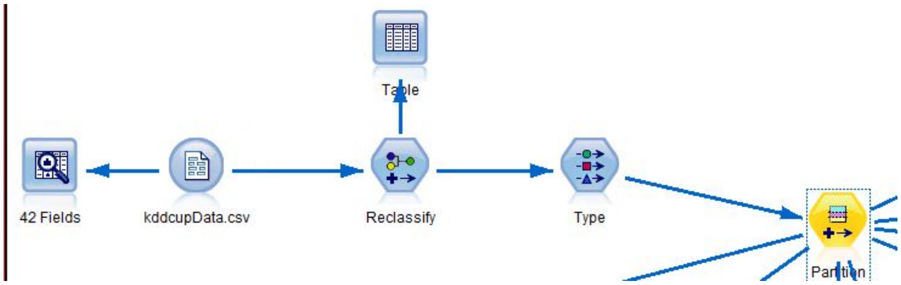


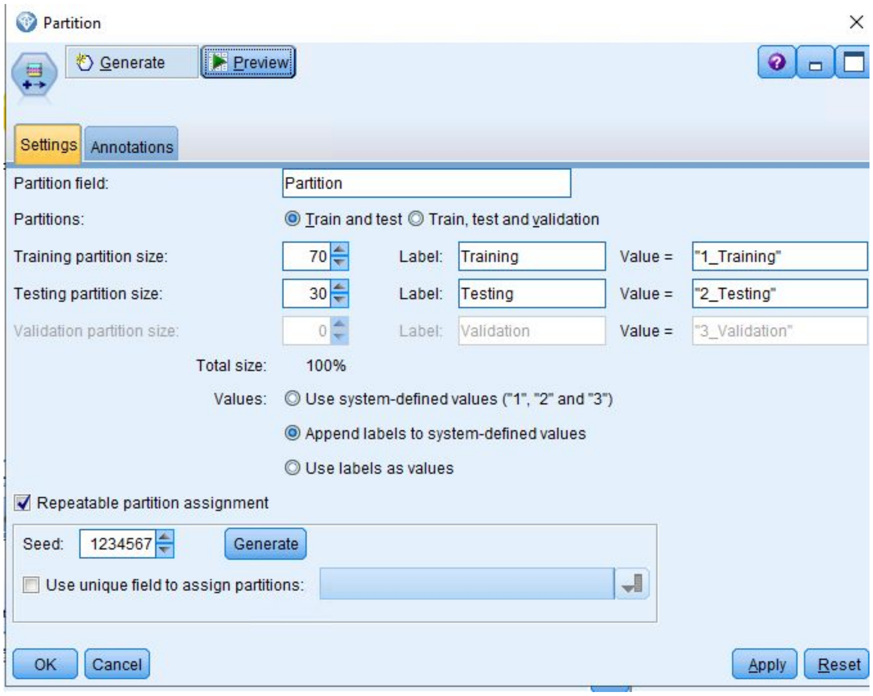
Field	Measurement	Values	Missing	Check	Role
duration	Continuous	[0,42448]		None	Input
protocol_type	Nominal	icmp,tcp,udp		None	Input
service	Nominal	IRC,Z39_50,auth,bgp,c...		None	Input
flag	Nominal	REJ,RSTO,RSTOS0,R...		None	Input
src_bytes	Continuous	[0,5135678]		None	Input
dst_bytes	Continuous	[0,5151385]		None	Input
land	Continuous	[0,1]		None	Input
wrong_fragment	Continuous	[0,3]		None	Input
urgent	Continuous	[0,2]		None	Input
hot	Continuous	[0,30]		None	Input
num_failed_logins	Continuous	[0,2]		None	Input
logged_in	Continuous	[0,1]		None	Input
num_compromised	Continuous	[0,102]		None	Input
root_shell	Continuous	[0,1]		None	Input
su_attempted	Continuous	[0,2]		None	Input
num_root	Continuous	[0,119]		None	Input
num_file_creations	Continuous	[0,22]		None	Input
num_shells	Continuous	[0,1]		None	Input
num_access_files	Continuous	[0,3]		None	Input
num_outbound_cmds	Flag	0/0		None	Input
is_host_login	Flag	0/0		None	Input
is_guest_login	Flag	1/0		None	Input
count	Continuous	[1,511]		None	Input
srv_count	Continuous	[1,511]		None	Input
serror_rate	Continuous	[0.0,1.0]		None	Input
srv_serror_rate	Continuous	[0.0,1.0]		None	Input
rerror_rate	Continuous	[0,1]		None	Input
srv_rerror_rate	Continuous	[0,1]		None	Input
same_srv_rate	Continuous	[0,1]		None	Input

Shown below is the stream used to reclassify the dataset into normal and attack variables.



Described below is the partitioning of the data set into 70/30 training and testing using the Partition Node:





Partition field: Partition

Partitions: ● Train and test ○ Train, test and validation

Training partition size:	70	Label:	Training	Value =	"1_Training"
Testing partition size:	30	Label:	Testing	Value =	"2_Testing"
Validation partition size:	0	Label:	Validation	Value =	"3_Validation"

Total size: 100%

Values: ○ Use system-defined values ("1", "2" and "3")

● Append labels to system-defined values

○ Use labels as values

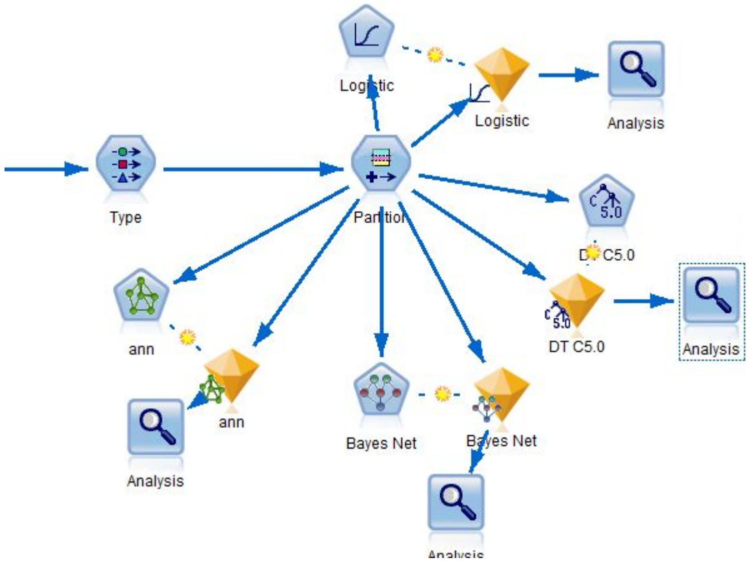
☑ Repeatable partition assignment

Seed: 1234567 Generate

□ Use unique field to assign partitions:

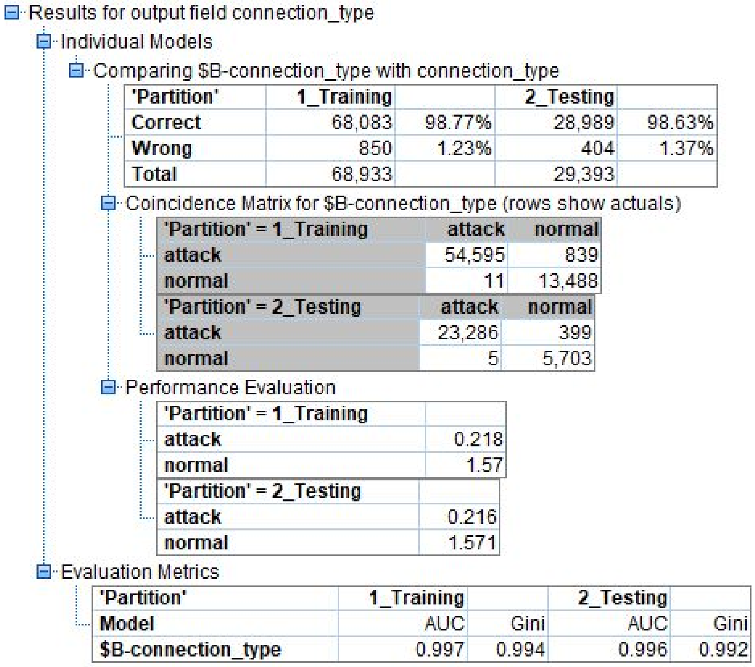
The following techniques will be used for classification:

1. Bayesian Network.
2. Logistic Regression
3. Artificial Neural Network
4. Decision Tree



Bayesian Network:

Let’s connect the partition to theBayesian Net and analyze the results.



Results for output field connection_type

Individual Models

Comparing $B-connection_type with connection_type

'Partition'	1_Training		2_Testing	
Correct	68,083	98.77%	28,989	98.63%
Wrong	850	1.23%	404	1.37%
Total	68,933		29,393	

Coincidence Matrix for $B-connection_type (rows show actuals)

'Partition' = 1_Training	attack	normal
attack	54,595	839
normal	11	13,488

'Partition' = 2_Testing	attack	normal
attack	23,286	399
normal	5	5,703

Performance Evaluation

'Partition' = 1_Training	
attack	0.218
normal	1.57

'Partition' = 2_Testing	
attack	0.216
normal	1.571

Evaluation Metrics

'Partition'	1_Training		2_Testing	
Model	AUC	Gini	AUC	Gini
$B-connection_type	0.997	0.994	0.996	0.992

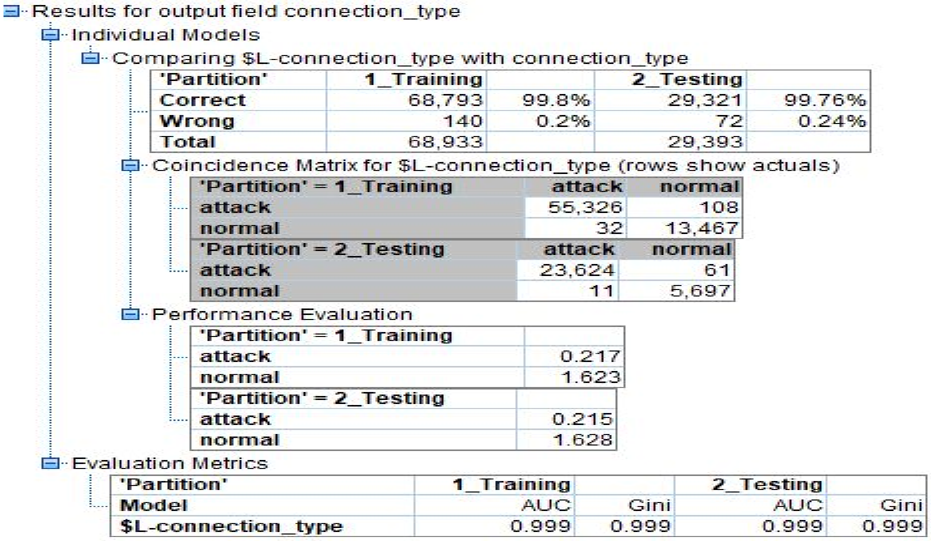
The accuracy is given by the formula below:

Accuracy (testing) = TP+TN/TP+TN+FP+FN

(23,286 + 5,703) / (23,286 + 5,703 + 399 + 5) *100 = **98.626%**

Logistic Regression:

Let’s connect the partition to the Logistic regression node and analyze the results.



Results for output field connection_type

Individual Models

Comparing $L-connection_type with connection_type

'Partition'	1_Training		2_Testing	
Correct	68,793	99.8%	29,321	99.76%
Wrong	140	0.2%	72	0.24%
Total	68,933		29,393	

Coincidence Matrix for $L-connection_type (rows show actuals)

'Partition' = 1_Training	attack	normal
attack	55,326	108
normal	32	13,467
'Partition' = 2_Testing	**attack**	**normal**
attack	23,624	61
normal	11	5,697

Performance Evaluation

'Partition' = 1_Training	
attack	0.217
normal	1.623
'Partition' = 2_Testing	
attack	0.215
normal	1.628

Evaluation Metrics

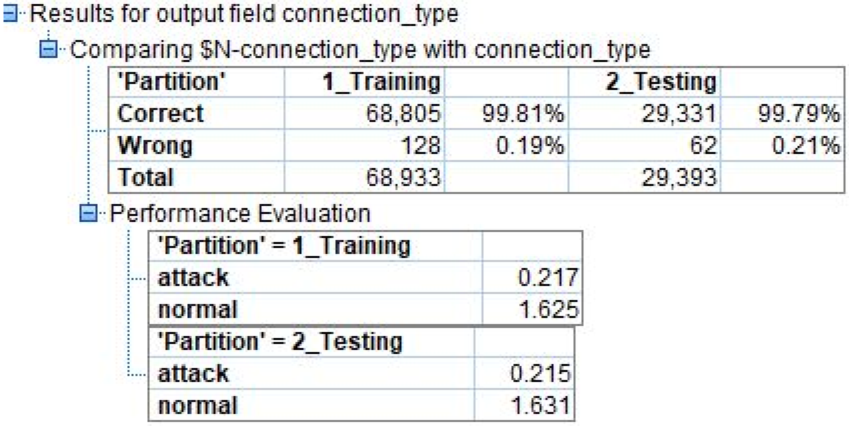
'Partition'	1_Training		2_Testing	
Model	AUC	Gini	AUC	Gini
$L-connection_type	0.999	0.999	0.999	0.999

Accuracy (testing) = TP+TN/TP+TN+FP+FN

(23,624 +56970)/ (23,624 +5,697 +61 + 11)* 100 =**99.7565**

Artificial Neural Network:

Let’s connect the partition to the Neural Network node and analyze the results.



Results for output field connection_type

Comparing $N-connection_type with connection_type

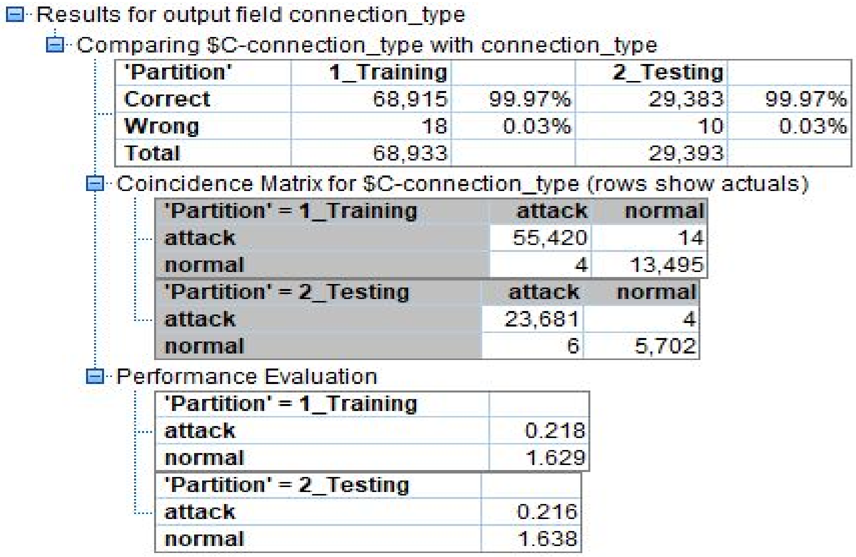
'Partition'	1_Training		2_Testing	
Correct	68,805	99.81%	29,331	99.79%
Wrong	128	0.19%	62	0.21%
Total	68,933		29,393	

Performance Evaluation

'Partition' = 1_Training	
attack	0.217
normal	1.625
'Partition' = 2_Testing	
attack	0.215
normal	1.631

Decision Tree(C5.0 2 Attacks):

Let’s connect the partition to Decision Tree and analyze the results.



Results for output field connection_type

Comparing $C-connection_type with connection_type

'Partition'	1_Training		2_Testing	
Correct	68,915	99.97%	29,383	99.97%
Wrong	18	0.03%	10	0.03%
Total	68,933		29,393	

Coincidence Matrix for $C-connection_type (rows show actuals)

'Partition' = 1_Training	attack	normal
attack	55,420	14
normal	4	13,495

'Partition' = 2_Testing	attack	normal
attack	23,681	4
normal	6	5,702

Performance Evaluation

'Partition' = 1_Training	
attack	0.218
normal	1.629

'Partition' = 2_Testing	
attack	0.216
normal	1.638

Accuracy (testing) = TP+TN/TP+TN+FP+FN

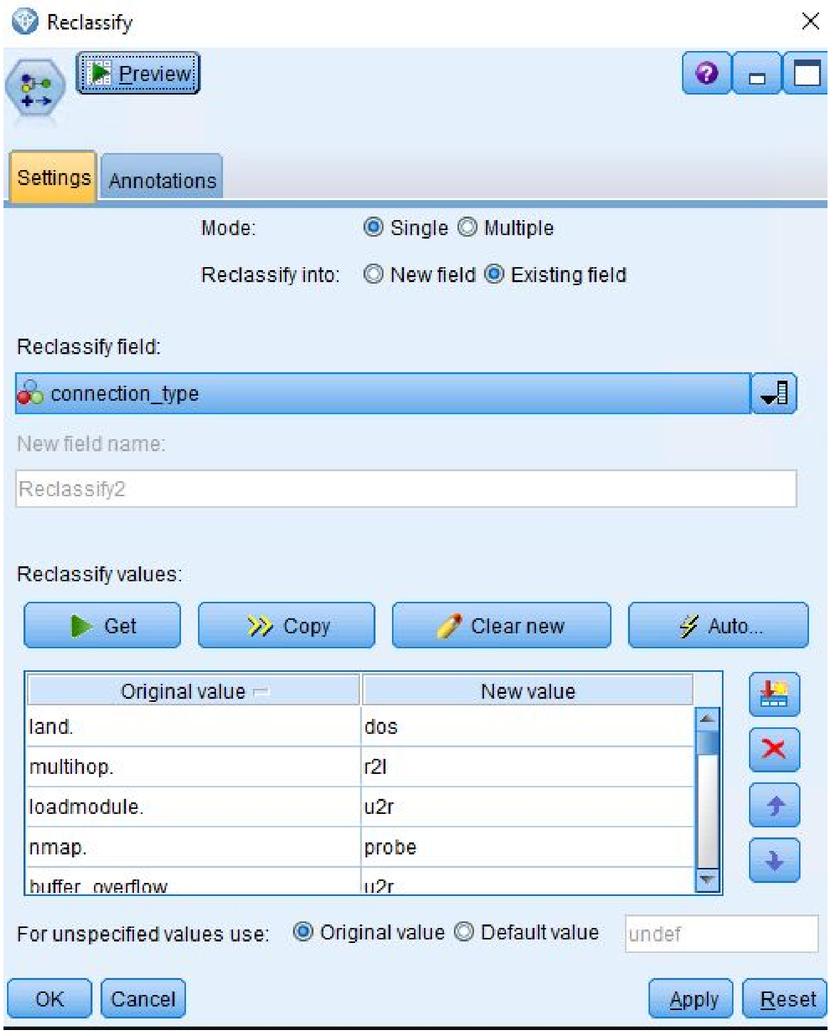
(23,681+5,702)/ (23,681+5,702+4+6)*100= **99.966%**

Refer to DT1 in the Decision Tree document.
As we can see the Decision tree gives us the best prediction accuracy of 99.97 % hence we will choose this method for the re-classification.

Decision Tree(C5.0 - 4 Attacks)

Let us create Decision tree(C5.0) by reclassifying the attacks in the 4 type a shown below :

1. DOS
2. R2L
3. U2R
4. Probe



Mode: ● Single ○ Multiple

Reclassify into: ○ New field ● Existing field

Reclassify field:

connection_type

New field name:

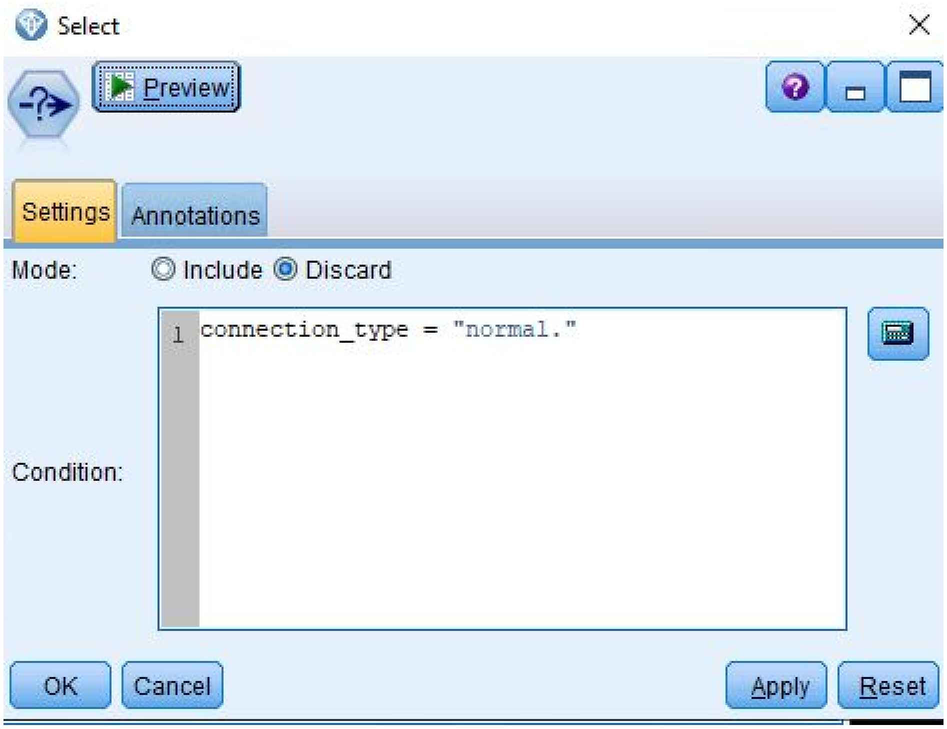
Reclassify2

Reclassify values:

Original value	New value
land.	dos
multihop.	r2l
loadmodule.	u2r
nmap.	probe
buffer_overflow	u2r

For unspecified values use: ● Original value ○ Default value undef

We don't want to consider the Normal type in our decision tree so we use Select node to skip it.

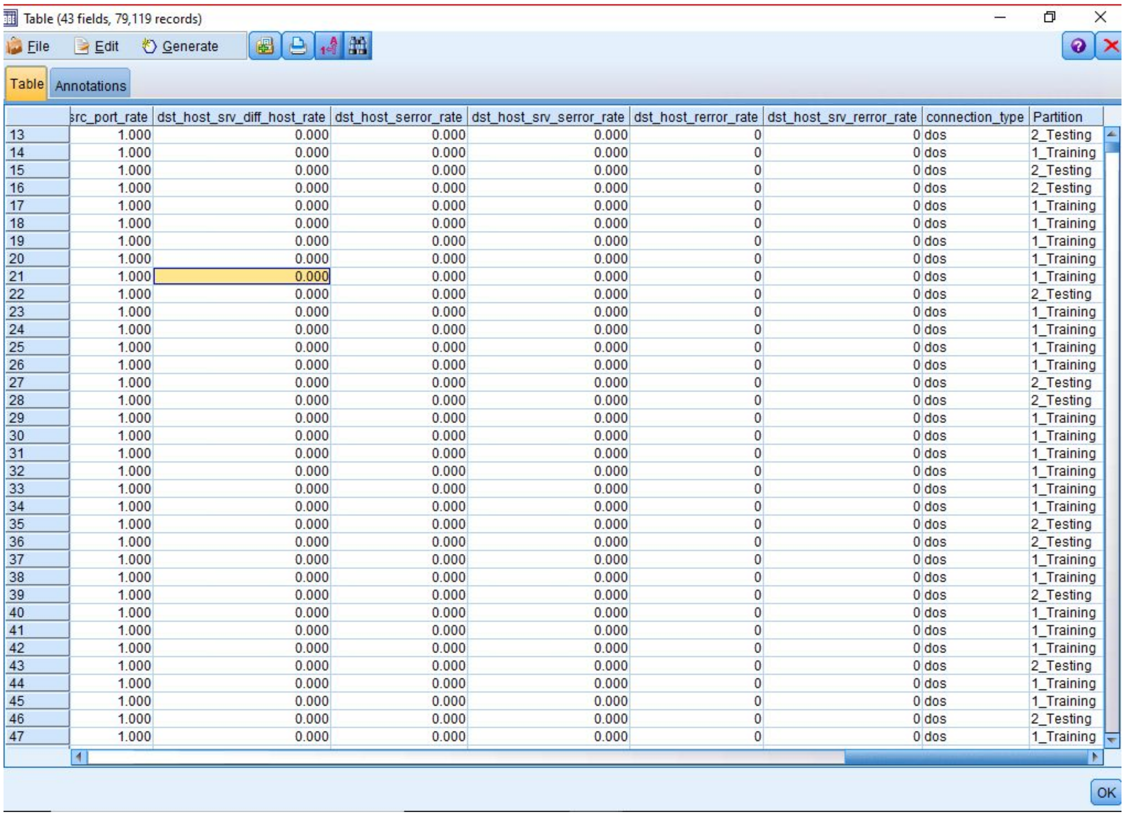


Mode: ○ Include ● Discard

Condition:

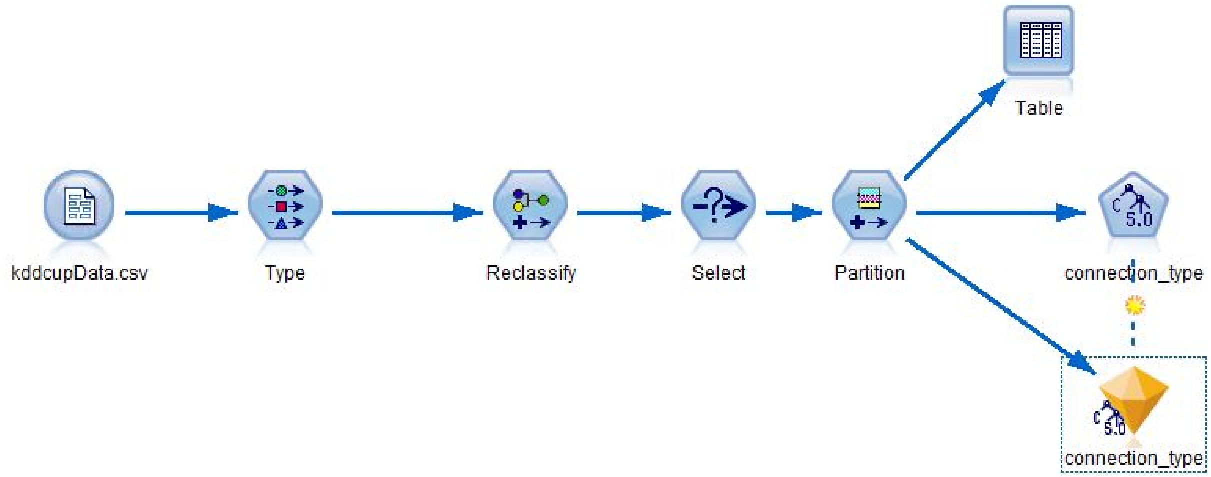
```
connection_type = "normal."
```

Partition the data in 70 (training) and 30 (testing). Let us see how data looks like after reclassification and partition.

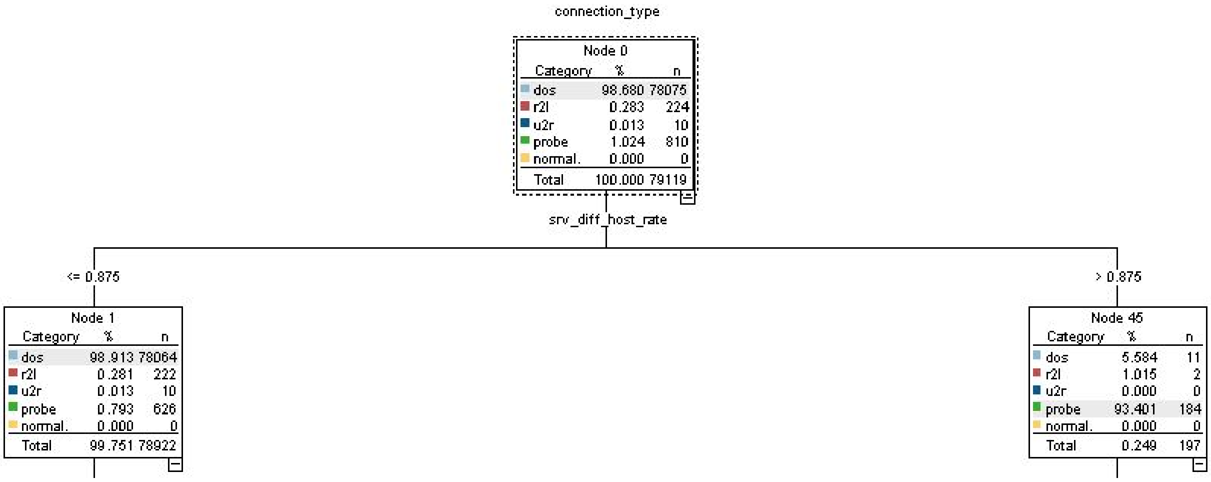


	src_port_rate	dst_host_srv_diff_host_rate	dst_host_serror_rate	dst_host_srv_serror_rate	dst_host_rerror_rate	dst_host_srv_rerror_rate	connection_type	Partition
13	1.000	0.000	0.000	0.000	0	0	dos	2_Testing
14	1.000	0.000	0.000	0.000	0	0	dos	1_Training
15	1.000	0.000	0.000	0.000	0	0	dos	2_Testing
16	1.000	0.000	0.000	0.000	0	0	dos	2_Testing
17	1.000	0.000	0.000	0.000	0	0	dos	1_Training
18	1.000	0.000	0.000	0.000	0	0	dos	1_Training
19	1.000	0.000	0.000	0.000	0	0	dos	1_Training
20	1.000	0.000	0.000	0.000	0	0	dos	1_Training
21	1.000	0.000	0.000	0.000	0	0	dos	1_Training
22	1.000	0.000	0.000	0.000	0	0	dos	2_Testing
23	1.000	0.000	0.000	0.000	0	0	dos	1_Training
24	1.000	0.000	0.000	0.000	0	0	dos	1_Training
25	1.000	0.000	0.000	0.000	0	0	dos	1_Training
26	1.000	0.000	0.000	0.000	0	0	dos	1_Training
27	1.000	0.000	0.000	0.000	0	0	dos	2_Testing
28	1.000	0.000	0.000	0.000	0	0	dos	2_Testing
29	1.000	0.000	0.000	0.000	0	0	dos	1_Training
30	1.000	0.000	0.000	0.000	0	0	dos	1_Training
31	1.000	0.000	0.000	0.000	0	0	dos	1_Training
32	1.000	0.000	0.000	0.000	0	0	dos	1_Training
33	1.000	0.000	0.000	0.000	0	0	dos	1_Training
34	1.000	0.000	0.000	0.000	0	0	dos	1_Training
35	1.000	0.000	0.000	0.000	0	0	dos	2_Testing
36	1.000	0.000	0.000	0.000	0	0	dos	2_Testing
37	1.000	0.000	0.000	0.000	0	0	dos	1_Training
38	1.000	0.000	0.000	0.000	0	0	dos	1_Training
39	1.000	0.000	0.000	0.000	0	0	dos	2_Testing
40	1.000	0.000	0.000	0.000	0	0	dos	1_Training
41	1.000	0.000	0.000	0.000	0	0	dos	1_Training
42	1.000	0.000	0.000	0.000	0	0	dos	1_Training
43	1.000	0.000	0.000	0.000	0	0	dos	2_Testing
44	1.000	0.000	0.000	0.000	0	0	dos	1_Training
45	1.000	0.000	0.000	0.000	0	0	dos	1_Training
46	1.000	0.000	0.000	0.000	0	0	dos	2_Testing
47	1.000	0.000	0.000	0.000	0	0	dos	1_Training

Now connect the C5.0 model to the partition node and run the stream.



Just to give the reference Decision Tree has assigned 0 value to the normal as we have to skip. For full decision refer to the Decision Tree doc labeled as ‘DT2’’.



Node 0

Category	%	n
dos	98.680	78075
r2l	0.283	224
u2r	0.013	10
probe	1.024	810
normal.	0.000	0
Total	100.000	79119

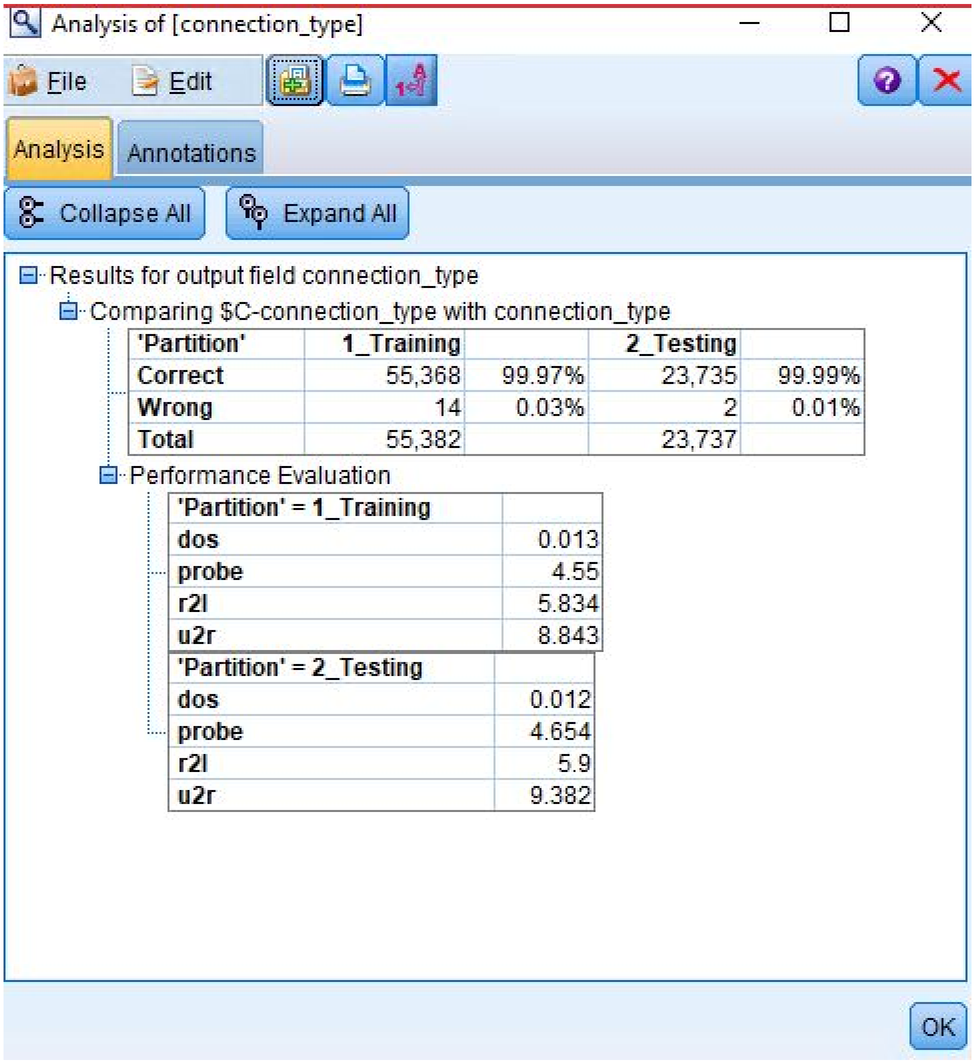
Node 1

Category	%	n
dos	98.913	78064
r2l	0.281	222
u2r	0.013	10
probe	0.793	626
normal.	0.000	0
Total	99.751	78922

Node 45

Category	%	n
dos	5.584	11
r2l	1.015	2
u2r	0.000	0
probe	93.401	184
normal.	0.000	0
Total	0.249	197

Lets, connect to the analysis node and analyze the result



Results for output field connection_type

Comparing $C-connection_type with connection_type

'Partition'	1_Training		2_Testing	
Correct	55,368	99.97%	23,735	99.99%
Wrong	14	0.03%	2	0.01%
Total	55,382		23,737	

Performance Evaluation

'Partition' = 1_Training	
dos	0.013
probe	4.55
r2l	5.834
u2r	8.843

'Partition' = 2_Testing	
dos	0.012
probe	4.654
r2l	5.9
u2r	9.382

Conclusion:

After implementing various classifiers using SPSS, we have decided that the most accurate model is the Decision Tree which has an accuracy of 99.966 % where attacks were reclassified into two. When we removed the normal attack types in the next section and reclassified into 4 attack types , accuracy was improved making it to nearly 99.99 %.

References:

1. https://www.sciencedirect.com/topics/computer-science/network-intrusion