In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import itertools
import random

# model imports
from sklearn.tree import DecisionTreeClassifier
from sklearn.linear_model import LogisticRegression

# processing imports
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_val_score
from sklearn.metrics import mean_absolute_error
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
features= [
    'duration',
    'protocol_type',
    'service',
    'flag',
    'src_bytes',
    'dst_bytes',
    'land',
    'wrong_fragment',
    'urgent',
    'hot',
    'num_failed_logins',
    'logged_in',
    'num_compromised',
    'root_shell',
    'su_attempted',
    'num_root',
    'num_file_creations',
    'num_shells',
    'num_access_files',
    'num_outbound_cmds',
    'is_host_login',
    'is_guest_login',
    'count',
    'srv_count',
    'serror_rate',
    'srv_serror_rate',
    'rerror_rate',
    'srv_rerror_rate',
    'same_srv_rate',
    'diff_srv_rate',
    'srv_diff_host_rate',
    'dst_host_count',
    'dst_host_srv_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_srv_serror_rate',
    'dst_host_rerror_rate',
    'dst_host_srv_rerror_rate',
    'intrusion_type'
]
from keras.utils.data_utils import get_file
try:
    path = get_file('kddcup.data_10_percent_corrected', origin=
    'https://raw.githubusercontent.com/JeevanSandhu/Intrusion-Detection/master/dataset/kddcup.data_10_percent_corrected')
except:
    print('Error downloading')
    raise
```

In [2]:

```
data = pd.read_csv(path,names=features ,header=None)
print(data)
output = data['intrusion_type'].values
labels = set(output)
print('The different type of output labels are:',labels)
print('='*125)
print('No. of different output labels are:', len(labels))
print('Null values in dataset are',len(data[data.isnull().any(1)]))
data.drop_duplicates(subset=features, keep='first', inplace = True)
data.shape
data.to_pickle('data.pkl')
plt.figure(figsize=(20,15))
class_distribution = data['intrusion_type'].value_counts()
class_distribution.plot(kind='bar')
plt.xlabel('Class')
plt.ylabel('Data points per Class')
plt.title('Distribution of yi in train data')
plt.grid()
plt.show()
# ref: arg sort https://docs.scipy.org/doc/numpy/reference/generated/numpy.argsort.html
# -(train_class_distribution.values): the minus sign will give us in decreasing order
sorted_yi = np.argsort(-class_distribution.values)
for i in sorted_yi:
    print('Number of data points in class', i+1,':', class_distribution.values[i], '(', np.round((class_distribution.values[i]/data.shape[0]*100), 3), '%)')
```

```
        duration protocol_type service flag  src_bytes  dst_bytes  land  \
0              0           tcp    http   SF        181       5450     0
1              0           tcp    http   SF        239        486     0
2              0           tcp    http   SF        235       1337     0
3              0           tcp    http   SF        219       1337     0
4              0           tcp    http   SF        217       2032     0
5              0           tcp    http   SF        217       2032     0
6              0           tcp    http   SF        212       1940     0
7              0           tcp    http   SF        159       4087     0
8              0           tcp    http   SF        210        151     0
9              0           tcp    http   SF        212        786     0
10             0           tcp    http   SF        210        624     0
11             0           tcp    http   SF        177       1985     0
12             0           tcp    http   SF        222        773     0
13             0           tcp    http   SF        256       1169     0
14             0           tcp    http   SF        241        259     0
15             0           tcp    http   SF        260       1837     0
16             0           tcp    http   SF        241        261     0
17             0           tcp    http   SF        257        818     0
18             0           tcp    http   SF        233        255     0
19             0           tcp    http   SF        233        504     0
20             0           tcp    http   SF        256       1273     0
21             0           tcp    http   SF        234        255     0
22             0           tcp    http   SF        241        259     0
23             0           tcp    http   SF        239        968     0
24             0           tcp    http   SF        245       1919     0
25             0           tcp    http   SF        248       2129     0
26             0           tcp    http   SF        354       1752     0
27             0           tcp    http   SF        193       3991     0
28             0           tcp    http   SF        214      14959     0
29             0           tcp    http   SF        212       1309     0
...          ...           ...     ...  ...        ...        ...   ...
493991         0           tcp    http   SF        296        617     0
493992         0           tcp    http   SF        294      29288     0
493993         0           tcp    http   SF        285      34557     0
493994         0           tcp    http   SF        316       3665     0
493995         0           tcp    http   SF        335      10440     0
493996         0           tcp    http   SF        284      10592     0
493997         0           tcp    http   SF        242       7066     0
493998         0           tcp    http   SF        223       3707     0
493999         0           tcp    http   SF        204       1731     0
494000         0           tcp    http   SF        148       1122     0
494001         0           tcp    http   S0          0          0     0
494002         0           tcp    http   SF        215       2649     0
494003         0           tcp    http   SF        341        326     0
494004         0           tcp    http   SF        341       1943     0
494005         0           tcp    http   SF        341       1663     0
494006         0           tcp    http   SF        235        501     0
494007         0           tcp    http   SF        320      13828     0
494008         0           tcp    http   SF        319       1435     0
494009         0           tcp    http   SF        335       3435     0
494010         0           tcp    http   SF        291        236     0
494011         0           tcp    http   SF        308        662     0
494012         0           tcp    http   SF        291       1862     0
494013         0           tcp    http   SF        289        244     0
494014         0           tcp    http   SF        306        662     0
494015         0           tcp    http   SF        289       1862     0
```

```
494016         0           tcp    http   SF        310       1881     0
494017         0           tcp    http   SF        282       2286     0
494018         0           tcp    http   SF        203       1200     0
494019         0           tcp    http   SF        291       1200     0
494020         0           tcp    http   SF        219       1234     0

        wrong_fragment  urgent  hot  ...  dst_host_srv_count  \
0                    0       0    0  ...                   9   
1                    0       0    0  ...                  19   
2                    0       0    0  ...                  29   
3                    0       0    0  ...                  39   
4                    0       0    0  ...                  49   
5                    0       0    0  ...                  59   
6                    0       0    0  ...                  69   
7                    0       0    0  ...                  79   
8                    0       0    0  ...                  89   
9                    0       0    1  ...                  99   
10                   0       0    0  ...                 109   
11                   0       0    0  ...                 119   
12                   0       0    0  ...                 129   
13                   0       0    0  ...                 139   
14                   0       0    0  ...                 149   
15                   0       0    0  ...                 159   
16                   0       0    0  ...                 169   
17                   0       0    0  ...                 179   
18                   0       0    0  ...                 189   
19                   0       0    0  ...                 199   
20                   0       0    0  ...                 209   
21                   0       0    0  ...                 219   
22                   0       0    0  ...                 229   
23                   0       0    0  ...                 239   
24                   0       0    0  ...                 249   
25                   0       0    0  ...                 255   
26                   0       0    0  ...                 255   
27                   0       0    0  ...                 255   
28                   0       0    0  ...                 255   
29                   0       0    0  ...                 255   
...                ...     ...  ...  ...                 ...   
493991               0       0    0  ...                 255   
493992               0       0    0  ...                 255   
493993               0       0    0  ...                 255   
493994               0       0    0  ...                 255   
493995               0       0    0  ...                 255   
493996               0       0    0  ...                 255   
493997               0       0    0  ...                 255   
493998               0       0    0  ...                 255   
493999               0       0    0  ...                 255   
494000               0       0    0  ...                 255   
494001               0       0    0  ...                 255   
494002               0       0    0  ...                 255   
494003               0       0    0  ...                 255   
494004               0       0    0  ...                 255   
494005               0       0    0  ...                 255   
494006               0       0    0  ...                 255   
494007               0       0    0  ...                 255   
494008               0       0    0  ...                 255   
494009               0       0    0  ...                 255   
```

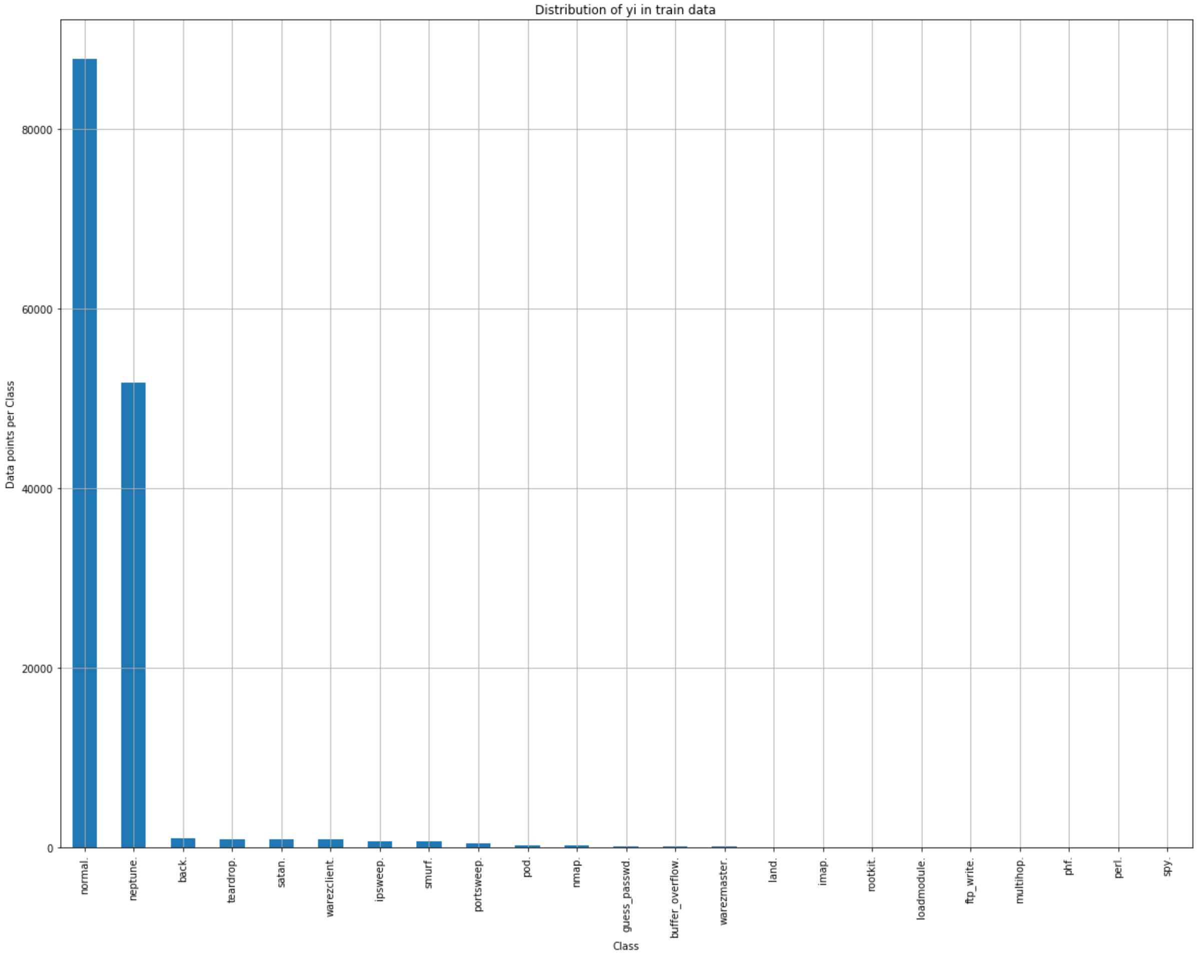
```
494010               0       0    0  ...                 255
494011               0       0    0  ...                 255
494012               0       0    0  ...                 255
494013               0       0    0  ...                 255
494014               0       0    0  ...                 255
494015               0       0    0  ...                 255
494016               0       0    0  ...                 255
494017               0       0    0  ...                 255
494018               0       0    0  ...                 255
494019               0       0    0  ...                 255
494020               0       0    0  ...                 255

        dst_host_same_srv_rate  dst_host_diff_srv_rate  \
0                          1.0                     0.0   
1                          1.0                     0.0   
2                          1.0                     0.0   
3                          1.0                     0.0   
4                          1.0                     0.0   
5                          1.0                     0.0   
6                          1.0                     0.0   
7                          1.0                     0.0   
8                          1.0                     0.0   
9                          1.0                     0.0   
10                         1.0                     0.0   
11                         1.0                     0.0   
12                         1.0                     0.0   
13                         1.0                     0.0   
14                         1.0                     0.0   
15                         1.0                     0.0   
16                         1.0                     0.0   
17                         1.0                     0.0   
18                         1.0                     0.0   
19                         1.0                     0.0   
20                         1.0                     0.0   
21                         1.0                     0.0   
22                         1.0                     0.0   
23                         1.0                     0.0   
24                         1.0                     0.0   
25                         1.0                     0.0   
26                         1.0                     0.0   
27                         1.0                     0.0   
28                         1.0                     0.0   
29                         1.0                     0.0   
...                        ...                     ...   
493991                     1.0                     0.0   
493992                     1.0                     0.0   
493993                     1.0                     0.0   
493994                     1.0                     0.0   
493995                     1.0                     0.0   
493996                     1.0                     0.0   
493997                     1.0                     0.0   
493998                     1.0                     0.0   
493999                     1.0                     0.0   
494000                     1.0                     0.0   
494001                     1.0                     0.0   
494002                     1.0                     0.0   
494003                     1.0                     0.0   
```

```
494004                     1.0                     0.0
494005                     1.0                     0.0
494006                     1.0                     0.0
494007                     1.0                     0.0
494008                     1.0                     0.0
494009                     1.0                     0.0
494010                     1.0                     0.0
494011                     1.0                     0.0
494012                     1.0                     0.0
494013                     1.0                     0.0
494014                     1.0                     0.0
494015                     1.0                     0.0
494016                     1.0                     0.0
494017                     1.0                     0.0
494018                     1.0                     0.0
494019                     1.0                     0.0
494020                     1.0                     0.0

        dst_host_same_src_port_rate  dst_host_srv_diff_host_rate  \
0                              0.11                         0.00   
1                              0.05                         0.00   
2                              0.03                         0.00   
3                              0.03                         0.00   
4                              0.02                         0.00   
5                              0.02                         0.00   
6                              1.00                         0.04   
7                              0.09                         0.04   
8                              0.12                         0.04   
9                              0.12                         0.05   
10                             0.06                         0.05   
11                             0.04                         0.04   
12                             0.03                         0.04   
13                             0.25                         0.04   
14                             0.07                         0.04   
15                             0.04                         0.04   
16                             0.03                         0.04   
17                             0.02                         0.03   
18                             0.02                         0.03   
19                             0.02                         0.03   
20                             0.01                         0.03   
21                             0.01                         0.03   
22                             0.01                         0.03   
23                             0.33                         0.03   
24                             0.08                         0.03   
25                             0.04                         0.03   
26                             0.20                         0.04   
27                             1.00                         0.05   
28                             0.09                         0.05   
29                             0.05                         0.05   
...                             ...                          ...   
493991                         0.04                         0.10   
493992                         0.03                         0.10   
493993                         0.14                         0.10   
493994                         0.06                         0.09   
493995                         0.04                         0.08   
493996                         0.03                         0.07   
493997                         0.02                         0.05   
```

```
493998                         0.11                         0.05
493999                         0.05                         0.04
494000                         0.33                         0.04
494001                         0.15                         0.04
494002                         0.04                         0.04
494003                         1.00                         0.05
494004                         0.09                         0.05
494005                         0.05                         0.05
494006                         0.50                         0.05
494007                         0.10                         0.05
494008                         0.17                         0.07
494009                         0.06                         0.07
494010                         0.04                         0.06
494011                         0.03                         0.06
494012                         0.02                         0.05
494013                         0.02                         0.05
494014                         0.02                         0.05
494015                         0.01                         0.05
494016                         0.01                         0.05
494017                         0.17                         0.05
494018                         0.06                         0.05
494019                         0.04                         0.05
494020                         0.17                         0.05

        dst_host_serror_rate  dst_host_srv_serror_rate  dst_host_rerror_rate
\
0                       0.00                      0.00                  0.00
1                       0.00                      0.00                  0.00
2                       0.00                      0.00                  0.00
3                       0.00                      0.00                  0.00
4                       0.00                      0.00                  0.00
5                       0.00                      0.00                  0.00
6                       0.00                      0.00                  0.00
7                       0.00                      0.00                  0.00
8                       0.00                      0.00                  0.00
9                       0.00                      0.00                  0.00
10                      0.00                      0.00                  0.00
11                      0.00                      0.00                  0.00
12                      0.00                      0.00                  0.00
13                      0.00                      0.00                  0.00
14                      0.00                      0.00                  0.00
15                      0.00                      0.00                  0.00
16                      0.00                      0.00                  0.00
17                      0.00                      0.00                  0.00
18                      0.00                      0.00                  0.00
19                      0.00                      0.00                  0.00
20                      0.00                      0.00                  0.00
21                      0.00                      0.00                  0.00
22                      0.00                      0.00                  0.00
23                      0.00                      0.00                  0.00
24                      0.00                      0.00                  0.00
25                      0.00                      0.00                  0.00
26                      0.00                      0.00                  0.00
27                      0.00                      0.00                  0.00
28                      0.00                      0.00                  0.00
29                      0.00                      0.00                  0.00
...                      ...                       ...                   ...
```

```
493991                  0.00                       0.00                  0.00
493992                  0.00                       0.00                  0.00
493993                  0.00                       0.00                  0.00
493994                  0.00                       0.00                  0.00
493995                  0.00                       0.00                  0.00
493996                  0.00                       0.00                  0.00
493997                  0.00                       0.00                  0.00
493998                  0.00                       0.00                  0.00
493999                  0.00                       0.00                  0.00
494000                  0.00                       0.00                  0.00
494001                  0.08                       0.00                  0.08
494002                  0.04                       0.00                  0.04
494003                  0.00                       0.01                  0.00
494004                  0.00                       0.01                  0.00
494005                  0.00                       0.01                  0.00
494006                  0.00                       0.01                  0.00
494007                  0.00                       0.01                  0.00
494008                  0.00                       0.01                  0.00
494009                  0.00                       0.01                  0.00
494010                  0.00                       0.01                  0.00
494011                  0.00                       0.01                  0.00
494012                  0.00                       0.01                  0.00
494013                  0.00                       0.01                  0.00
494014                  0.00                       0.01                  0.00
494015                  0.00                       0.01                  0.00
494016                  0.00                       0.01                  0.00
494017                  0.00                       0.01                  0.00
494018                  0.06                       0.01                  0.00
494019                  0.04                       0.01                  0.00
494020                  0.00                       0.01                  0.00

        dst_host_srv_rerror_rate  intrusion_type
0                            0.0         normal.
1                            0.0         normal.
2                            0.0         normal.
3                            0.0         normal.
4                            0.0         normal.
5                            0.0         normal.
6                            0.0         normal.
7                            0.0         normal.
8                            0.0         normal.
9                            0.0         normal.
10                           0.0         normal.
11                           0.0         normal.
12                           0.0         normal.
13                           0.0         normal.
14                           0.0         normal.
15                           0.0         normal.
16                           0.0         normal.
17                           0.0         normal.
18                           0.0         normal.
19                           0.0         normal.
20                           0.0         normal.
21                           0.0         normal.
22                           0.0         normal.
23                           0.0         normal.
24                           0.0         normal.
```

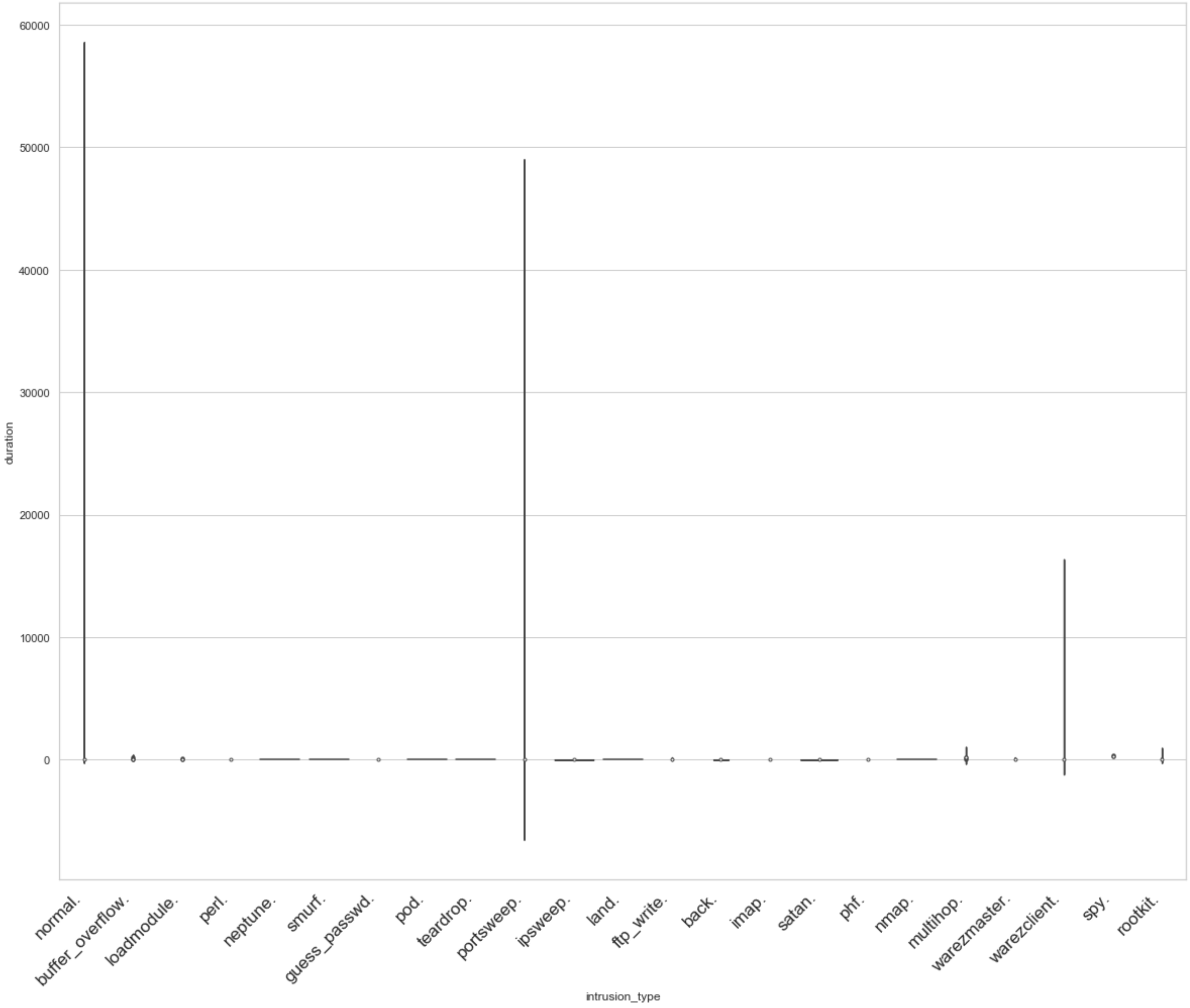
```
25                           0.0         normal.
26                           0.0         normal.
27                           0.0         normal.
28                           0.0         normal.
29                           0.0         normal.
...                          ...             ...
493991                       0.0         normal.
493992                       0.0         normal.
493993                       0.0         normal.
493994                       0.0         normal.
493995                       0.0         normal.
493996                       0.0         normal.
493997                       0.0         normal.
493998                       0.0         normal.
493999                       0.0         normal.
494000                       0.0         normal.
494001                       0.0         normal.
494002                       0.0         normal.
494003                       0.0         normal.
494004                       0.0         normal.
494005                       0.0         normal.
494006                       0.0         normal.
494007                       0.0         normal.
494008                       0.0         normal.
494009                       0.0         normal.
494010                       0.0         normal.
494011                       0.0         normal.
494012                       0.0         normal.
494013                       0.0         normal.
494014                       0.0         normal.
494015                       0.0         normal.
494016                       0.0         normal.
494017                       0.0         normal.
494018                       0.0         normal.
494019                       0.0         normal.
494020                       0.0         normal.

[494021 rows x 42 columns]
The different type of output labels are: {'land.', 'pod.', 'satan.', 'rootki
t.', 'neptune.', 'nmap.', 'perl.', 'loadmodule.', 'warezclient.', 'ipsweep.',
'smurf.', 'back.', 'phf.', 'normal.', 'portsweep.', 'teardrop.', 'warezmaste
r.', 'multihop.', 'ftp_write.', 'imap.', 'guess_passwd.', 'spy.', 'buffer_ove
rflow.'}
==============================================================================
================================================
No. of different output labels are: 23
Null values in dataset are 0
```



```
Number of data points in class 1 : 87832 ( 60.33 %)
Number of data points in class 2 : 51820 ( 35.594 %)
Number of data points in class 3 : 968 ( 0.665 %)
Number of data points in class 4 : 918 ( 0.631 %)
Number of data points in class 5 : 906 ( 0.622 %)
Number of data points in class 6 : 893 ( 0.613 %)
Number of data points in class 7 : 651 ( 0.447 %)
Number of data points in class 8 : 641 ( 0.44 %)
Number of data points in class 9 : 416 ( 0.286 %)
Number of data points in class 10 : 206 ( 0.141 %)
Number of data points in class 11 : 158 ( 0.109 %)
Number of data points in class 12 : 53 ( 0.036 %)
Number of data points in class 13 : 30 ( 0.021 %)
Number of data points in class 14 : 20 ( 0.014 %)
Number of data points in class 15 : 19 ( 0.013 %)
Number of data points in class 16 : 12 ( 0.008 %)
Number of data points in class 17 : 10 ( 0.007 %)
Number of data points in class 18 : 9 ( 0.006 %)
Number of data points in class 19 : 8 ( 0.005 %)
Number of data points in class 20 : 7 ( 0.005 %)
Number of data points in class 21 : 4 ( 0.003 %)
Number of data points in class 22 : 3 ( 0.002 %)
Number of data points in class 23 : 2 ( 0.001 %)
```

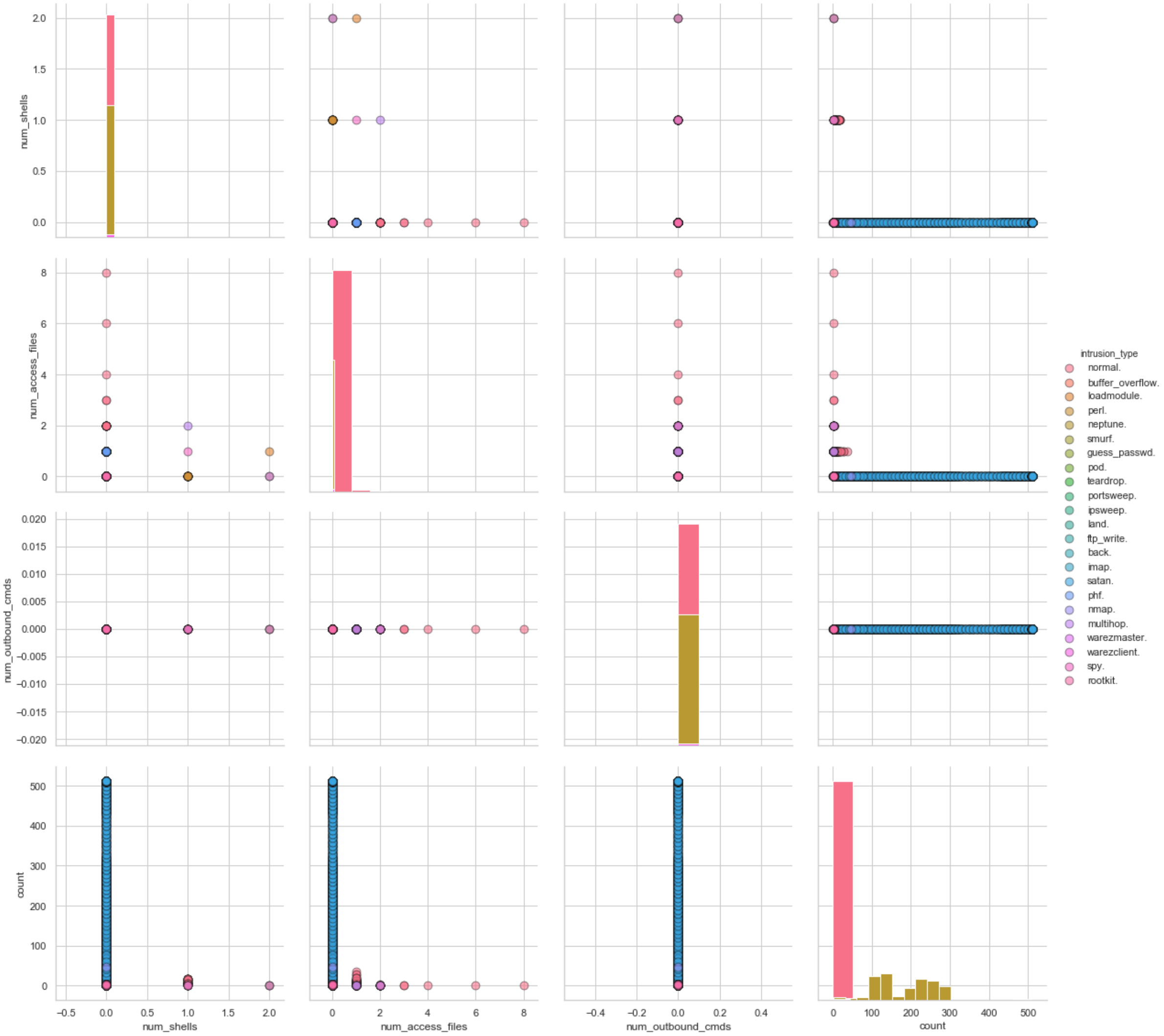
In [3]:

```
import seaborn as sns
plt.figure(figsize=(20,16))
sns.set(style="whitegrid")
ax = sns.violinplot(x="intrusion_type", y="duration", data=data, fliersize=Non
e)
plt.xticks(
    rotation=45,
    horizontalalignment='right',
    fontweight='light',
    fontsize='x-large'
)
def pairplot(data, label, features=[]):
  '''
    This function creates pairplot taking 4 features from our dataset as defau
lt parameters along with the output variable
    '''
  sns.pairplot(data, hue=label, height=4, diag_kind='hist',   vars=features, p
lot_kws={'alpha':0.6, 's':80, 'edgecolor':'k'})
```



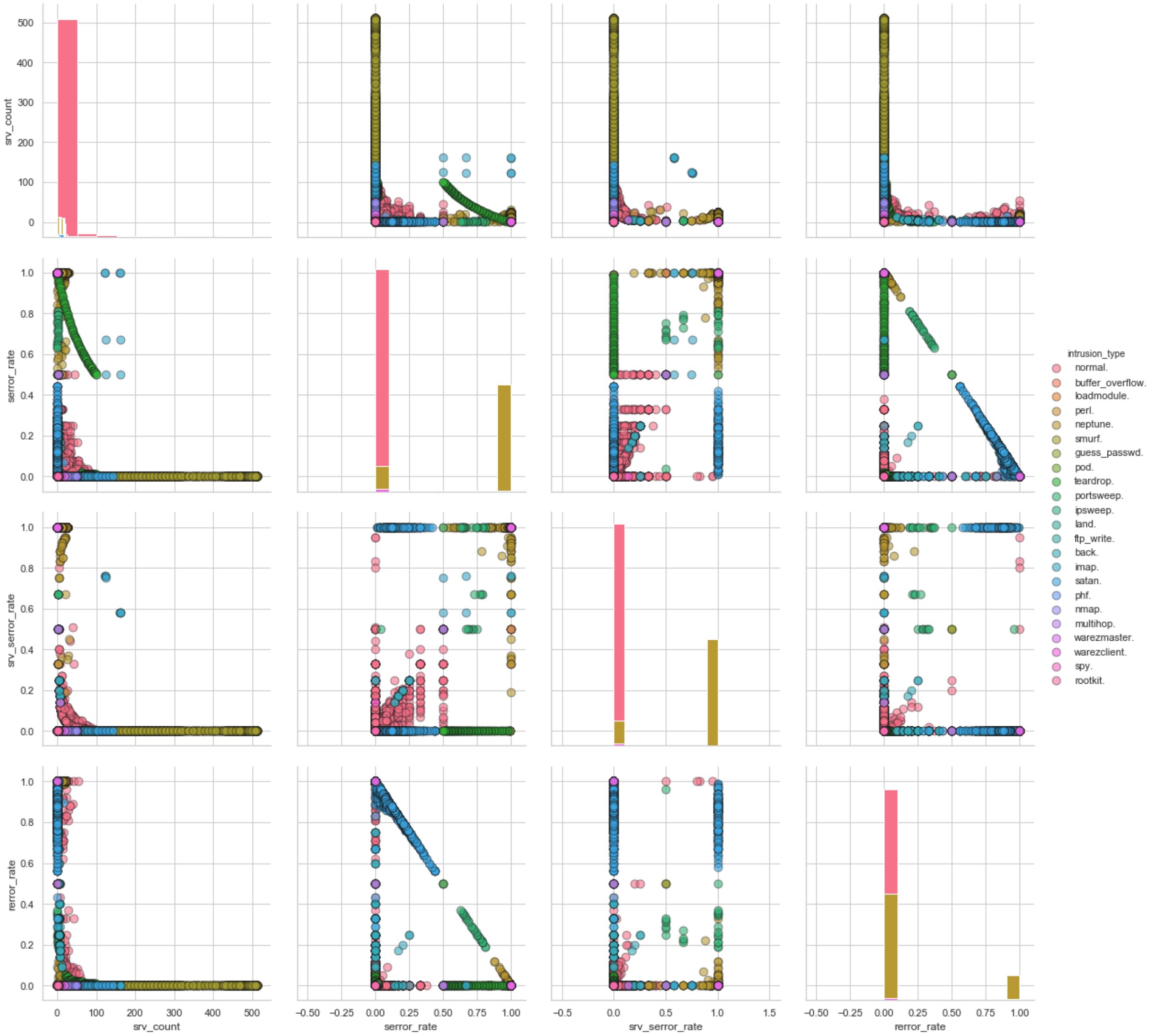
In [4]:

```
pairplot(data, 'intrusion_type', features=['num_shells','num_access_files','num_outbound_cmds','count'])
```



In [5]:

```
pairplot(data, 'intrusion_type', features=['srv_count','serror_rate','srv_serror_rate','rerror_rate'])
```



```
In [6]: def tsne_func(data, label, no_components, perplexity_value, n_iter_value):
          '''
          This function applies TSNE on the original dataset with  no_components, perp
        lexity_value, n_iter_value as the TSNE parameters
          and transforms the original dataset into TSNE transformed feature space with
        the tsne dataset containing number of features
          equal to the value specified for no_components and also plots the scatter pl
        ot of the transformed data points along with
          their class label
          '''
          print('TSNE with perplexity={} and no. of iterations={}'.format(perplexity_v
        alue, n_iter_value))
          tsne = TSNE(n_components=no_components, perplexity=perplexity_value, n_iter=
        n_iter_value)
          tsne_df1 = tsne.fit_transform(data)
          print(tsne_df1.shape)
          tsne_df1 = np.vstack((tsne_df1.T, Y)).T
          tsne_data1 = pd.DataFrame(data=tsne_df1, columns=['feature1', 'feature2', 'O
        utput'])
          sns.FacetGrid(tsne_data1, hue='Output', size=6).map(plt.scatter, 'feature1',
        'feature2').add_legend()
          plt.show()
        from sklearn.manifold import TSNE
        from sklearn.model_selection import train_test_split
        X_train, X_test, Y_train, Y_test = train_test_split(data.drop('intrusion_type'
        , axis=1), data['intrusion_type'], stratify=data['intrusion_type'], test_size=
        0.25)
        print('Train data')
        print(X_train.shape)
        print(Y_train.shape)
        print('='*20)
        print('Test data')
        print(X_test.shape)
        print(Y_test.shape)
        protocol = list(X_train['protocol_type'].values)
        protocol = list(set(protocol))
        print('Protocol types are:', protocol)
        from sklearn.feature_extraction.text import CountVectorizer
        one_hot = CountVectorizer(vocabulary=protocol, binary=True)
        train_protocol = one_hot.fit_transform(X_train['protocol_type'].values)
        test_protocol = one_hot.transform(X_test['protocol_type'].values)
        print(train_protocol[1].toarray())
        print(train_protocol.shape)
        service = list(X_train['service'].values)
        service = list(set(service))
        print('Service types are:', service)
        from sklearn.feature_extraction.text import CountVectorizer
        one_hot = CountVectorizer(vocabulary=service, binary=True)
        train_service = one_hot.fit_transform(X_train['service'].values)
        test_service = one_hot.transform(X_test['service'].values)
        print(train_service[1].toarray())
        print(train_service.shape)
        flag = list(X_train['flag'].values)
        flag = list(X_train['flag'].values)
        flag = list(set(flag))
        print('flag types are:', flag)
```

```
from sklearn.feature_extraction.text import CountVectorizer
one_hot = CountVectorizer(vocabulary=flag, binary=True)
train_flag = one_hot.fit_transform(X_train['flag'].values)
test_flag = one_hot.transform(X_test['flag'].values)
print(train_flag[1].toarray())
print(train_flag.shape)
from sklearn.preprocessing import StandardScaler
def feature_scaling(X_train, X_test, feature_name):
  '''
  This function performs standardisation on the features
  '''
  scaler = StandardScaler()
  scaler1 = scaler.fit_transform(X_train[feature_name].values.reshape(-1,1))
  scaler2 = scaler.transform(X_test[feature_name].values.reshape(-1,1))

  return scaler1, scaler2
duration1, duration2 = feature_scaling(X_train, X_test, 'duration')
print(duration1[1])
src_bytes1, src_bytes2 = feature_scaling(X_train, X_test, 'src_bytes')
print(src_bytes1[1])
dst_bytes1, dst_bytes2 = feature_scaling(X_train, X_test, 'dst_bytes')
print(dst_bytes1[1])
land1, land2 = feature_scaling(X_train, X_test, 'land')
wrong_fragment1, wrong_fragment2 = feature_scaling(X_train, X_test, 'wrong_fragment')
urgent1,urgent2 = feature_scaling(X_train, X_test, 'urgent')
hot1, hot2 = feature_scaling(X_train, X_test, 'hot')
num_failed_logins1, num_failed_logins2 = feature_scaling(X_train, X_test, 'num_failed_logins')
logged_in1, logged_in2 = feature_scaling(X_train, X_test, 'logged_in')
num_compromised1, num_compromised2 = feature_scaling(X_train, X_test, 'num_compromised')
root_shell1, root_shell2 = feature_scaling(X_train, X_test, 'root_shell')
su_attempted1, su_attempted2 = feature_scaling(X_train, X_test, 'su_attempted')
num_root1, num_root2 = feature_scaling(X_train, X_test, 'num_root')
num_shells1, num_shells2 = feature_scaling(X_train, X_test, 'num_shells')
num_file_creations1, num_file_creations2 = feature_scaling(X_train, X_test, 'num_file_creations')
num_access_files1, num_access_files2 = feature_scaling(X_train, X_test, 'num_access_files')
is_host_login1, is_host_login2 = feature_scaling(X_train, X_test, 'is_host_login')
is_guest_login1, is_guest_login2 = feature_scaling(X_train, X_test, 'is_guest_login')
count1, count2 = feature_scaling(X_train, X_test, 'count')
srv_count1, srv_count2 = feature_scaling(X_train, X_test, 'srv_count')
serror_rate1, serror_rate2 = feature_scaling(X_train, X_test, 'serror_rate')
srv_serror_rate1, srv_serror_rate2 = feature_scaling(X_train, X_test, 'srv_serror_rate')
rerror_rate1, rerror_rate2 = feature_scaling(X_train, X_test, 'rerror_rate')
srv_rerror_rate1, srv_rerror_rate2 = feature_scaling(X_train, X_test, 'srv_rerror_rate')
same_srv_rate1, same_srv_rate2 = feature_scaling(X_train, X_test, 'same_srv_rate')
diff_srv_rate1, diff_srv_rate2 = feature_scaling(X_train, X_test, 'diff_srv_rate')
```

```
srv_diff_host_rate1, srv_diff_host_rate2 = feature_scaling(X_train, X_test, 'srv_diff_host_rate')
dst_host_count1, dst_host_count2 = feature_scaling(X_train, X_test, 'dst_host_count')
dst_host_srv_count1, dst_host_srv_count2 = feature_scaling(X_train, X_test, 'dst_host_srv_count')
dst_host_same_srv_rate1, dst_host_same_srv_rate2 = feature_scaling(X_train, X_test, 'dst_host_same_srv_rate')
dst_host_diff_srv_rate1, dst_host_diff_srv_rate2 = feature_scaling(X_train, X_test, 'dst_host_diff_srv_rate')
dst_host_same_src_port_rate1, dst_host_same_src_port_rate2 = feature_scaling(X_train, X_test, 'dst_host_same_src_port_rate')
dst_host_srv_diff_host_rate1, dst_host_srv_diff_host_rate2 = feature_scaling(X_train, X_test, 'dst_host_srv_diff_host_rate')
dst_host_serror_rate1, dst_host_serror_rate2 = feature_scaling(X_train, X_test, 'dst_host_serror_rate')
dst_host_srv_serror_rate1, dst_host_srv_serror_rate2 = feature_scaling(X_train, X_test, 'dst_host_srv_serror_rate')
dst_host_rerror_rate1, dst_host_rerror_rate2 = feature_scaling(X_train, X_test, 'dst_host_rerror_rate')
dst_host_srv_rerror_rate1, dst_host_srv_rerror_rate2 = feature_scaling(X_train, X_test, 'dst_host_srv_rerror_rate')
from scipy.sparse import hstack
X_train_1 = hstack((duration1, train_protocol, train_service, train_flag, src_bytes1, dst_bytes1, land1, wrong_fragment1, urgent1, hot1, num_failed_logins1, logged_in1, num_compromised1, root_shell1, su_attempted1, num_root1, num_file_creations1, num_shells1, num_access_files1, is_host_login1, is_guest_login1, count1, srv_count1, serror_rate1, srv_serror_rate1, rerror_rate1, srv_rerror_rate1, same_srv_rate1, diff_srv_rate1, srv_diff_host_rate1, dst_host_count1, dst_host_srv_count1, dst_host_same_srv_rate1, dst_host_diff_srv_rate1, dst_host_same_src_port_rate1, dst_host_srv_diff_host_rate1, dst_host_serror_rate1, dst_host_srv_serror_rate1, dst_host_rerror_rate1, dst_host_srv_rerror_rate1))
X_test_1 = hstack((duration2, test_protocol, test_service, test_flag, src_bytes2, dst_bytes2, land2, wrong_fragment2, urgent2, hot2, num_failed_logins2, logged_in2, num_compromised2, root_shell2, su_attempted2, num_root2, num_file_creations2, num_shells2, num_access_files2, is_host_login2, is_guest_login2, count2, srv_count2, serror_rate2, srv_serror_rate2, rerror_rate2, srv_rerror_rate2, same_srv_rate2, diff_srv_rate2, srv_diff_host_rate2, dst_host_count2, dst_host_srv_count2, dst_host_same_srv_rate2, dst_host_diff_srv_rate2, dst_host_same_src_port_rate2, dst_host_srv_diff_host_rate2, dst_host_serror_rate2, dst_host_srv_serror_rate2, dst_host_rerror_rate2, dst_host_srv_rerror_rate2))
```

```
Train data
(109189, 41)
(109189,)
====================
Test data
(36397, 41)
(36397,)
Protocol types are: ['tcp', 'icmp', 'udp']
[[1 0 0]]
(109189, 3)
Service types are: ['netbios_dgm', 'klogin', 'csnet_ns', 'telnet', 'systat',
'iso_tsap', 'sunrpc', 'remote_job', 'name', 'ftp', 'ecr_i', 'smtp', 'eco_i',
'whois', 'netbios_ns', 'hostnames', 'mtp', 'tim_i', 'domain_u', 'vmnet', 'bg
p', 'courier', 'sql_net', 'domain', 'supdup', 'link', 'ssh', 'nnsp', 'urp_i',
'efs', 'time', 'pop_3', 'ldap', 'ntp_u', 'netstat', 'http', 'kshell', 'ctf',
'echo', 'http_443', 'uucp_path', 'shell', 'imap4', 'printer', 'login', 'disca
rd', 'X11', 'red_i', 'netbios_ssn', 'gopher', 'finger', 'pm_dump', 'daytime',
'auth', 'nntp', 'urh_i', 'IRC', 'private', 'exec', 'uucp', 'rje', 'tftp_u',
'pop_2', 'ftp_data', 'Z39_50', 'other']
[[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0]]
(109189, 66)
flag types are: ['OTH', 'SF', 'S2', 'RSTO', 'S3', 'S1', 'REJ', 'RSTOS0', 'S
0', 'SH', 'RSTR']
[[0 0 0 0 0 0 0 0 0 0 0]]
(109189, 11)
[-0.10952351]
[-0.00455701]
[-0.04801636]
```

```
In [7]: import datetime as dt
        from sklearn.metrics import accuracy_score, confusion_matrix, roc_auc_score, p
        recision_score, recall_score, f1_score
        from sklearn.model_selection import GridSearchCV
        from sklearn.externals import joblib
        def confusion_matrix_func(Y_test, y_test_pred):

            '''
            This function computes the confusion matrix using Predicted and Actual val
        ues and plots a confusion matrix heatmap
            '''
            C = confusion_matrix(Y_test, y_test_pred)
            cm_df = pd.DataFrame(C)
            labels = ['back', 'butter_overflow', 'loadmodule', 'guess_passwd', 'imap',
        'ipsweep', 'warezmaster', 'rootkit',
            'multihop', 'neptune', 'nmap', 'normal', 'phf', 'perl', 'pod', 'portsweep'
        , 'ftp_write', 'satan', 'smurf', 'teardrop', 'warezclient', 'land']
            plt.figure(figsize=(20,15))
            sns.set(font_scale=1.4)
            sns.heatmap(cm_df, annot=True, annot_kws={"size":12}, fmt='g',       xtick
        labels=labels, yticklabels=labels)
            plt.ylabel('Actual Class')
            plt.xlabel('Predicted Class')

            plt.show()
        def model(model_name, X_train, Y_train, X_test, Y_test):
          '''
          Fits the model on train data and predict the performance on train and test d
        ata.
          '''
          print('Fitting the model and prediction on train data:')
          start = dt.datetime.now()
          model_name.fit(X_train, Y_train)
          y_tr_pred = model_name.predict(X_train)
          print('Completed')
          print('Time taken:',dt.datetime.now()-start)
          print('='*50)

          results_tr = dict()
          y_tr_pred = model_name.predict(X_train)
          print(tpr_fpr_func(Y_train,y_tr_pred))
          results_tr['precision'] = precision_score(Y_train, y_tr_pred,         averag
        e='weighted')
          results_tr['recall'] = recall_score(Y_train, y_tr_pred, average='weighted')
          results_tr['f1_score'] = f1_score(Y_train, y_tr_pred, average='weighted')

          results_test = dict()
          print('Prediction on test data:')
          start = dt.datetime.now()
          y_test_pred = model_name.predict(X_test)
          print(tpr_fpr_func(Y_test,y_test_pred))
          print('Completed')
          print('Time taken:',dt.datetime.now()-start)
          print('='*50)

          print('Performance metrics:')
```

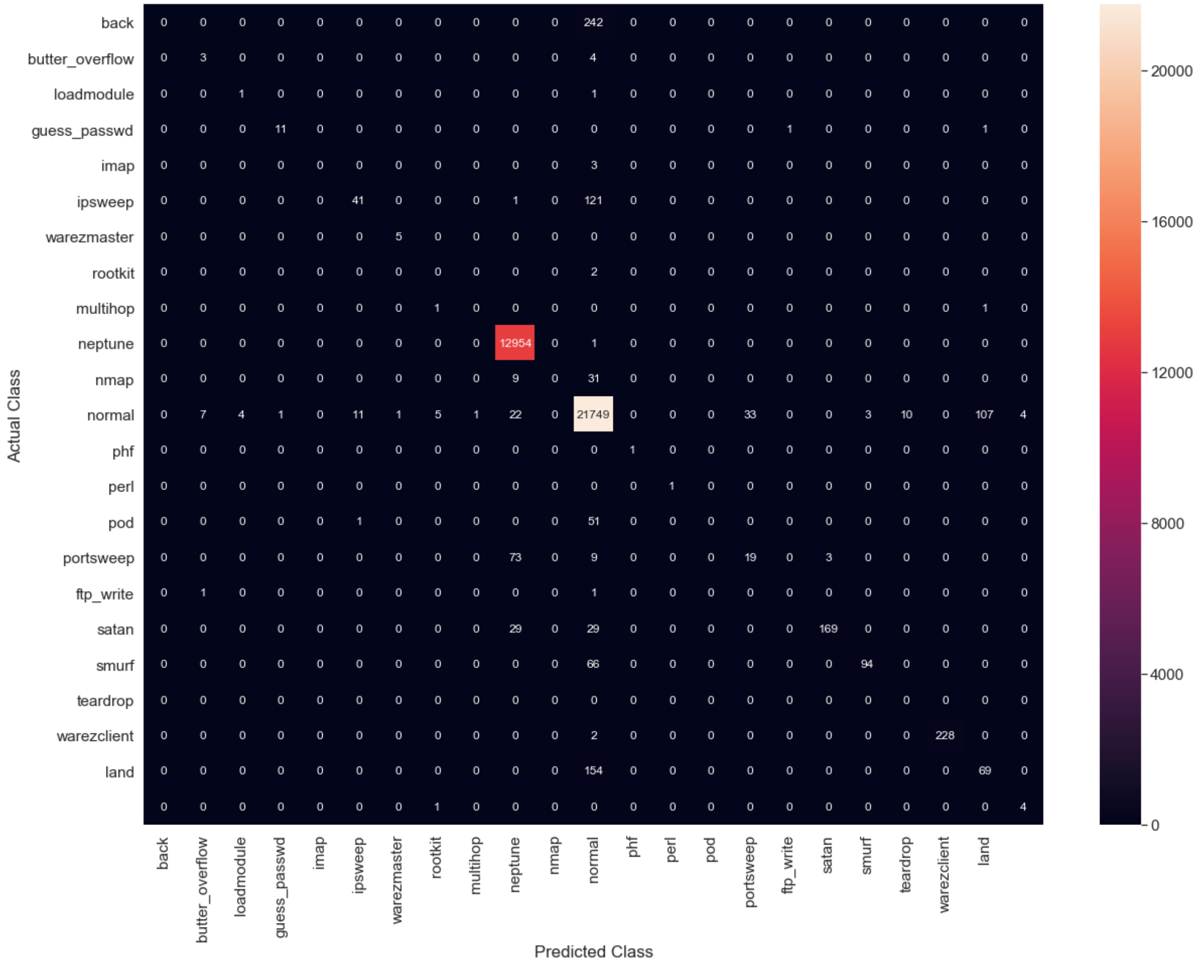
```
  print('='*50)
  print('Confusion Matrix is:')
  confusion_matrix_func(Y_test, y_test_pred)
  print('='*50)
  results_test['precision'] = precision_score(Y_test, y_test_pred, average='weighted')
  print('Precision score is:')
  print(precision_score(Y_test, y_test_pred, average='weighted'))
  print('='*50)
  results_test['recall'] = recall_score(Y_test, y_test_pred, average='weighted')
  print('Recall score is:')
  print(recall_score(Y_test, y_test_pred, average='weighted'))
  print('='*50)
  results_test['f1_score'] = f1_score(Y_test, y_test_pred, average='weighted')
  print('F1-score is:')
  print(f1_score(Y_test, y_test_pred, average='weighted'))
  # add the trained  model to the results
  results_test['model'] = model

  return results_tr, results_test
def print_grid_search_attributes(model):

  '''
  This function prints all the grid search attributes
  '''


  print('---------------------------')
  print('|      Best Estimator     |')
  print('---------------------------')
  print('\n\t{}\n'.format(model.best_estimator_))
  # parameters that gave best results while performing grid search
  print('---------------------------')
  print('|     Best parameters     |')
  print('---------------------------')
  print('\tParameters of best estimator : \n\n\t{}\n'.format(model.best_params_))
  #  number of cross validation splits
  print('----------------------------------')
  print('|   No of CrossValidation sets    |')
  print('----------------------------------')
  print('\n\tTotal number of cross validation sets: {}\n'.format(model.n_splits_))
  # Average cross validated score of the best estimator, from the Grid Search
  print('---------------------------')
  print('|        Best Score       |')
  print('---------------------------')
  print('\n\tAverage Cross Validate scores of best estimator : \n\n\t{}\n'.format(model.best_score_))
```

```
C:\Users\91969\Anaconda3\lib\site-packages\sklearn\externals\joblib\__init__.
py:15: DeprecationWarning: sklearn.externals.joblib is deprecated in 0.21 and
will be removed in 0.23. Please import this functionality directly from jobli
b, which can be installed with: pip install joblib. If this warning is raised
when loading pickled models, you may need to re-serialize those models with s
cikit-learn 0.21+.
  warnings.warn(msg, category=DeprecationWarning)
```

In [8]:

```
def tpr_fpr_func(Y_tr, Y_pred):
  '''
  This function computes the TPR and FPR scores using the actual and predicetd values.
  '''
  results = dict()
  Y_tr = Y_tr.to_list()
  tp = 0; fp = 0; positives = 0; negatives = 0; length = len(Y_tr)
  for i in range(len(Y_tr)):
    if Y_tr[i]=='normal.':
        positives += 1
    else:
        negatives += 1

  for i in range(len(Y_pred)):
    if Y_tr[i]=='normal.' and Y_pred[i]=='normal.':
        tp += 1
    elif Y_tr[i]!='normal.' and Y_pred[i]=='normal.':
        fp += 1

  tpr = tp/positives
  fpr = fp/negatives

  results['tp'] = tp; results['tpr'] = tpr; results['fp'] = fp; results['fpr'] = fpr

  return results
hyperparameter = {'var_smoothing':[10**x for x in range(-9,3)]}
from sklearn.naive_bayes import GaussianNB
nb = GaussianNB()
nb_grid = GridSearchCV(nb, param_grid=hyperparameter, cv=5, verbose=1, n_jobs=-1)
nb_grid_results = model(nb_grid, X_train_1.toarray(), Y_train, X_test_1.toarray(), Y_test)
```

```
Fitting the model and prediction on train data:
Fitting 5 folds for each of 12 candidates, totalling 60 fits

C:\Users\91969\Anaconda3\lib\site-packages\sklearn\model_selection\_split.py:
657: Warning: The least populated class in y has only 2 members, which is too
few. The minimum number of members in any class cannot be less than n_splits=
5.
  % (min_groups, self.n_splits)), Warning)
[Parallel(n_jobs=-1)]: Using backend LokyBackend with 12 concurrent workers.
[Parallel(n_jobs=-1)]: Done  26 tasks      | elapsed:   36.9s
[Parallel(n_jobs=-1)]: Done  60 out of  60 | elapsed:   57.6s finished

Completed
Time taken: 0:01:07.882527
==================================================
{'tp': 65291, 'tpr': 0.9911497707745089, 'fp': 2055, 'fpr': 0.047443149024587
32}

C:\Users\91969\Anaconda3\lib\site-packages\sklearn\metrics\classification.py:
1437: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 i
n labels with no predicted samples.
  'precision', 'predicted', average, warn_for)
C:\Users\91969\Anaconda3\lib\site-packages\sklearn\metrics\classification.py:
1437: UndefinedMetricWarning: F-score is ill-defined and being set to 0.0 in
labels with no predicted samples.
  'precision', 'predicted', average, warn_for)

Prediction on test data:
{'tp': 21749, 'tpr': 0.99048182894617, 'fp': 717, 'fpr': 0.04965717847496364}
Completed
Time taken: 0:00:02.164833
==================================================
Performance metrics:
==================================================
Confusion Matrix is:
```



	back	butter_overflow	loadmodule	guess_passwd	imap	ipsweep	warezmaster	rootkit	multihop	neptune	nmap	normal	phf	perl	pod	portsweep	ftp_write	satan	smurf	teardrop	warezclient	land	
back	0	0	0	0	0	0	0	0	0	0	0	242	0	0	0	0	0	0	0	0	0	0	0
butter_overflow	0	3	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0
loadmodule	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
guess_passwd	0	0	0	11	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0
imap	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0
ipsweep	0	0	0	0	0	41	0	0	0	1	0	121	0	0	0	0	0	0	0	0	0	0	0
warezmaster	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rootkit	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0
multihop	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
neptune	0	0	0	0	0	0	0	0	0	12954	0	1	0	0	0	0	0	0	0	0	0	0	0
nmap	0	0	0	0	0	0	0	0	0	9	0	31	0	0	0	0	0	0	0	0	0	0	0
normal	0	7	4	1	0	11	1	5	1	22	0	21749	0	0	0	33	0	0	3	10	0	107	4
phf	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
perl	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
pod	0	0	0	0	0	1	0	0	0	0	0	51	0	0	0	0	0	0	0	0	0	0	0
portsweep	0	0	0	0	0	0	0	0	0	73	0	9	0	0	0	19	0	3	0	0	0	0	0
ftp_write	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
satan	0	0	0	0	0	0	0	0	0	29	0	29	0	0	0	0	0	169	0	0	0	0	0
smurf	0	0	0	0	0	0	0	0	0	0	0	66	0	0	0	0	0	0	94	0	0	0	0
teardrop	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
warezclient	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	228	0	0
land	0	0	0	0	0	0	0	0	0	0	0	154	0	0	0	0	0	0	0	0	0	69	0
	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4

```
==================================================

C:\Users\91969\Anaconda3\lib\site-packages\sklearn\metrics\classification.py:
1437: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 i
n labels with no predicted samples.
  'precision', 'predicted', average, warn_for)

Precision score is:
0.9605488721458466
==================================================

C:\Users\91969\Anaconda3\lib\site-packages\sklearn\metrics\classification.py:
1439: UndefinedMetricWarning: Recall is ill-defined and being set to 0.0 in l
abels with no true samples.
  'recall', 'true', average, warn_for)

Recall score is:
0.9712064181113828
==================================================

C:\Users\91969\Anaconda3\lib\site-packages\sklearn\metrics\classification.py:
1437: UndefinedMetricWarning: F-score is ill-defined and being set to 0.0 in
labels with no predicted samples.
  'precision', 'predicted', average, warn_for)
C:\Users\91969\Anaconda3\lib\site-packages\sklearn\metrics\classification.py:
1439: UndefinedMetricWarning: F-score is ill-defined and being set to 0.0 in
labels with no true samples.
  'recall', 'true', average, warn_for)
```

```
F1-score is:
0.9647607402412135
```

In [9]:

```
from sklearn.linear_model import LogisticRegression
```

In [10]:

```
print_grid_search_attributes(nb_grid)
```

```
--------------------------
|      Best Estimator     |
--------------------------

        GaussianNB(priors=None, var_smoothing=10)

--------------------------
|     Best parameters     |
--------------------------
        Parameters of best estimator :

        {'var_smoothing': 10}

---------------------------------
|   No of CrossValidation sets   |
---------------------------------

        Total number of cross validation sets: 5

--------------------------
|        Best Score       |
--------------------------

        Average Cross Validate scores of best estimator :

        0.9713707424740587
```

In [11]:

```
Y_train_pred,Y_test_pred=nb_grid_results
print(nb_grid_results[:1])
!pip install xgboost
```

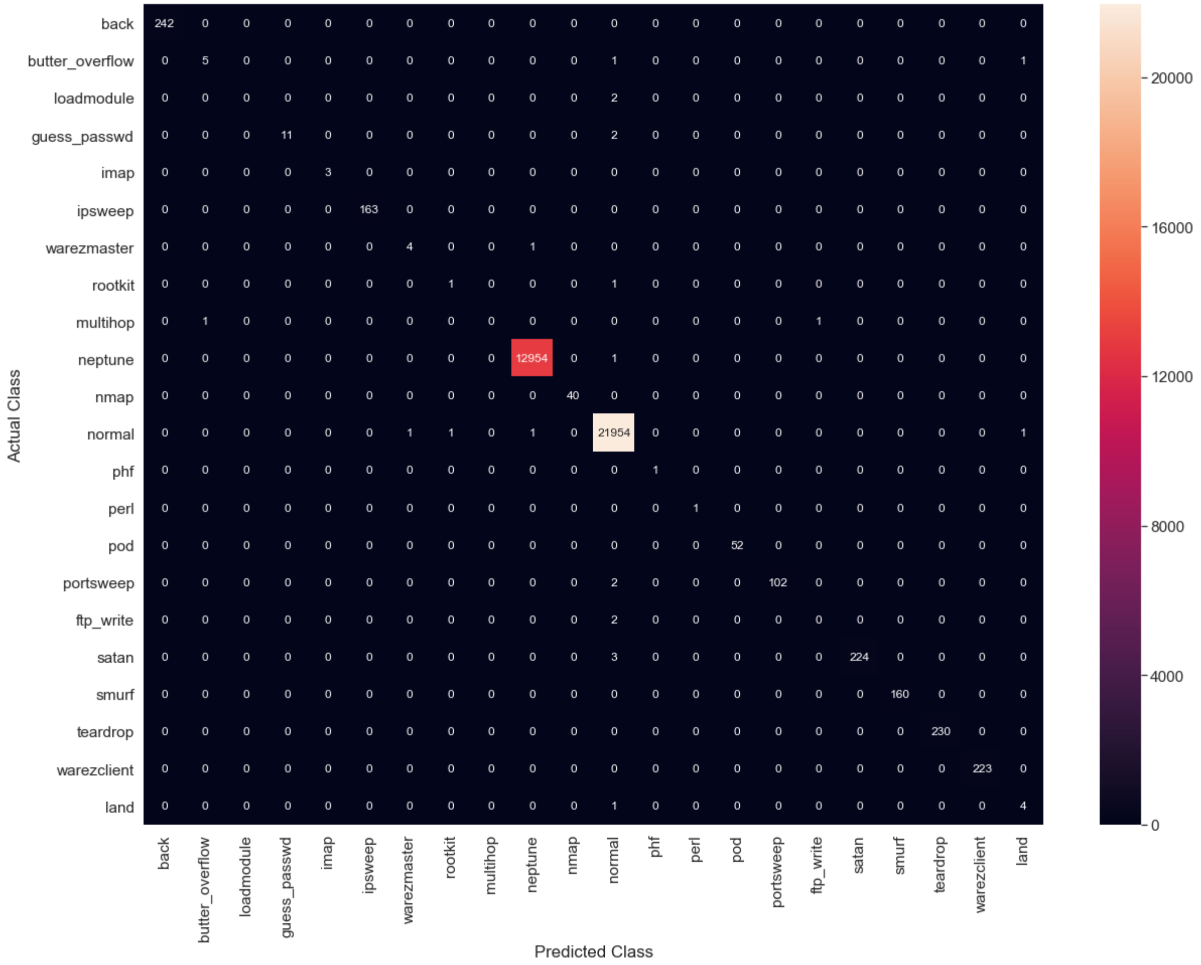
```
({'precision': 0.9617901935991932, 'recall': 0.9724056452573061, 'f1_score':
0.9660249765203339},)
Requirement already satisfied: xgboost in c:\users\91969\anaconda3\lib\site-p
ackages (1.4.2)
Requirement already satisfied: numpy in c:\users\91969\anaconda3\lib\site-pac
kages (from xgboost) (1.16.4)
Requirement already satisfied: scipy in c:\users\91969\anaconda3\lib\site-pac
kages (from xgboost) (1.4.1)
```

In [ ]:

In [12]:

```
hyperparameter = {'max_depth':[2, 3, 5, 7, 10], 'n_estimators': [10, 50, 100, 200, 500]}
from xgboost import XGBClassifier
xgb = XGBClassifier(objective='multi:softprob')
xgb_grid = GridSearchCV(xgb, param_grid=hyperparameter, cv=3, verbose=1, n_jobs=-1)
xgb_grid_results = model(xgb_grid, X_train_1.toarray(), Y_train, X_test_1.toarray(), Y_test)
```

```
Fitting the model and prediction on train data:
Fitting 3 folds for each of 25 candidates, totalling 75 fits

C:\Users\91969\Anaconda3\lib\site-packages\sklearn\model_selection\_split.py:
657: Warning: The least populated class in y has only 2 members, which is too
few. The minimum number of members in any class cannot be less than n_splits=
3.
  % (min_groups, self.n_splits)), Warning)
[Parallel(n_jobs=-1)]: Using backend LokyBackend with 12 concurrent workers.
[Parallel(n_jobs=-1)]: Done  26 tasks      | elapsed: 142.5min
[Parallel(n_jobs=-1)]: Done  75 out of  75 | elapsed: 204.7min finished
C:\Users\91969\Anaconda3\lib\site-packages\xgboost\sklearn.py:1146: UserWarni
ng: The use of label encoder in XGBClassifier is deprecated and will be remov
ed in a future release. To remove this warning, do the following: 1) Pass opt
ion use_label_encoder=False when constructing XGBClassifier object; and 2) En
code your labels (y) as integers starting with 0, i.e. 0, 1, 2, ..., [num_cla
ss - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)

[16:07:36] WARNING: C:/Users/Administrator/workspace/xgboost-win64_release_1.
4.0/src/learner.cc:1095: Starting in XGBoost 1.3.0, the default evaluation me
tric used with the objective 'multi:softprob' was changed from 'merror' to 'm
logloss'. Explicitly set eval_metric if you'd like to restore the old behavio
r.
Completed
Time taken: 3:27:02.333159
==================================================
{'tp': 65873, 'tpr': 0.9999848195039014, 'fp': 0, 'fpr': 0.0}
Prediction on test data:
{'tp': 21954, 'tpr': 0.9998178340468167, 'fp': 15, 'fpr': 0.00103885310617078
74}
Completed
Time taken: 0:00:00.160556
==================================================
Performance metrics:
==================================================
Confusion Matrix is:
```



	back	butter_overflow	loadmodule	guess_passwd	imap	ipsweep	warezmaster	rootkit	multihop	neptune	nmap	normal	phf	perl	pod	portsweep	ftp_write	satan	smurf	teardrop	warezclient	land
back	242	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
butter_overflow	0	5	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1
loadmodule	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0
guess_passwd	0	0	0	11	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0
imap	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ipsweep	0	0	0	0	0	163	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
warezmaster	0	0	0	0	0	0	4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
rootkit	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0
multihop	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
neptune	0	0	0	0	0	0	0	0	0	12954	0	1	0	0	0	0	0	0	0	0	0	0
nmap	0	0	0	0	0	0	0	0	0	0	40	0	0	0	0	0	0	0	0	0	0	0
normal	0	0	0	0	0	0	1	1	0	1	0	21954	0	0	0	0	0	0	0	0	0	1
phf	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
perl	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
pod	0	0	0	0	0	0	0	0	0	0	0	0	0	0	52	0	0	0	0	0	0	0
portsweep	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	102	0	0	0	0	0	0
ftp_write	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0
satan	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	224	0	0	0	0
smurf	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	160	0	0	0
teardrop	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	230	0	0
warezclient	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	223	0
land	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	4

```
==================================================
Precision score is:

C:\Users\91969\Anaconda3\lib\site-packages\sklearn\metrics\classification.py:
1437: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 i
n labels with no predicted samples.
  'precision', 'predicted', average, warn_for)

0.9992354955523365
==================================================
Recall score is:
0.9993680797867956
==================================================
F1-score is:

C:\Users\91969\Anaconda3\lib\site-packages\sklearn\metrics\classification.py:
1437: UndefinedMetricWarning: F-score is ill-defined and being set to 0.0 in
labels with no predicted samples.
  'precision', 'predicted', average, warn_for)

0.9992972057026701
```

In [1]:
```
print('himanshu')
```

```
himanshu
```

In [ ]: