step2

June 2, 2024

[ ]:
```
!pip install -r requirements.txt
```

[3]:
```
from pandas import set_option

def append_suffix(name: str, suffix: str) -> str:
    return f"{name.removesuffix('.parquet')}{suffix}.parquet"

set_option('display.max_columns', None)
```

0.1 Remove unnecessary data

- remove non-TCP flows (https://en.wikipedia.org/wiki/List_of_IP_protocol_numbers)
- remove rows containing infinity/NaN values in numeric columns
- remove columns that contain low amount of unique values
- remove columns with high correlation

[4]:
```
from pandas import read_parquet
from collections import Counter
from numpy import isfinite
import seaborn as sns
import matplotlib.pyplot as plt

file = 'data.parquet'

print(f"Reading file {file}")
df = read_parquet(file)
print(df.columns)

print(f"Starting dataframe shape: {df.shape}")
print(f"Counted rows by protocol: {Counter(df['Protocol'])}")
print("Removing non-TCP rows...")
df = df[df['Protocol'] == 6]

print("Remove 'DoS attacks-Slowloris' rows")

df = df[df['Label'] != 'DoS attacks-Slowloris']

print("Removing 'Protocol' and 'Timestamp' columns")
```

```
df.drop(columns=['Protocol', 'Timestamp'], inplace=True)

cols_to_remove = []

UNIQUE_VALUES_THRESHOLD = 10

for column in df:
    unique_vals = len(set(df[column]))
    print(f"column: '{column}', unique values: {unique_vals}")
    if unique_vals < UNIQUE_VALUES_THRESHOLD and column != 'Label':
        cols_to_remove.append(column)
        print(f"Removing column {column}")
        print(f"Column values: {Counter(df[column])}")

df.drop(columns=cols_to_remove, inplace=True)

print("Correlation matrix")
df_features = df.drop(columns=['Label'], inplace=False)

# removing rows with Inf/Nan values
finite_indexes = isfinite(df_features).all(1)
df_features = df_features[finite_indexes]
df = df[finite_indexes]

correlation_matrix = df_features.corr()

CORRELATION_THRESHOLD = 0.95

save_columns = []
drop_columns = []
for index, row in correlation_matrix.iterrows():
    for colname in correlation_matrix.columns:
        if colname == index:
            continue
        if colname in save_columns or colname in drop_columns:
            continue

        if abs(row[colname]) > CORRELATION_THRESHOLD and colname not in␣
 ↪save_columns:
            drop_columns.append(colname)
            save_columns.append(index)
            print(f"Removing column {colname} as its corellation to {index} is␣
 ↪{row[colname]}")

df.drop(columns=drop_columns, inplace=True)
df.reset_index(drop=True, inplace=True)
print(f"""
```

```
###################################
Final data state for file {file}
Shape (rows, columns): {df.shape}
Label counts: {Counter(df['Label'])}
###################################

""")
df.to_parquet(append_suffix(file, '_pruned'))
```

```
Reading file data.parquet
Index(['Dst Port', 'Protocol', 'Timestamp', 'Flow Duration', 'Tot Fwd Pkts',
       'Tot Bwd Pkts', 'TotLen Fwd Pkts', 'TotLen Bwd Pkts', 'Fwd Pkt Len Max',
       'Fwd Pkt Len Min', 'Fwd Pkt Len Mean', 'Fwd Pkt Len Std',
       'Bwd Pkt Len Max', 'Bwd Pkt Len Min', 'Bwd Pkt Len Mean',
       'Bwd Pkt Len Std', 'Flow Byts/s', 'Flow Pkts/s', 'Flow IAT Mean',
       'Flow IAT Std', 'Flow IAT Max', 'Flow IAT Min', 'Fwd IAT Tot',
       'Fwd IAT Mean', 'Fwd IAT Std', 'Fwd IAT Max', 'Fwd IAT Min',
       'Bwd IAT Tot', 'Bwd IAT Mean', 'Bwd IAT Std', 'Bwd IAT Max',
       'Bwd IAT Min', 'Fwd PSH Flags', 'Bwd PSH Flags', 'Fwd URG Flags',
       'Bwd URG Flags', 'Fwd Header Len', 'Bwd Header Len', 'Fwd Pkts/s',
       'Bwd Pkts/s', 'Pkt Len Min', 'Pkt Len Max', 'Pkt Len Mean',
       'Pkt Len Std', 'Pkt Len Var', 'FIN Flag Cnt', 'SYN Flag Cnt',
       'RST Flag Cnt', 'PSH Flag Cnt', 'ACK Flag Cnt', 'URG Flag Cnt',
       'CWE Flag Count', 'ECE Flag Cnt', 'Down/Up Ratio', 'Pkt Size Avg',
       'Fwd Seg Size Avg', 'Bwd Seg Size Avg', 'Fwd Byts/b Avg',
       'Fwd Pkts/b Avg', 'Fwd Blk Rate Avg', 'Bwd Byts/b Avg',
       'Bwd Pkts/b Avg', 'Bwd Blk Rate Avg', 'Subflow Fwd Pkts',
       'Subflow Fwd Byts', 'Subflow Bwd Pkts', 'Subflow Bwd Byts',
       'Init Fwd Win Byts', 'Init Bwd Win Byts', 'Fwd Act Data Pkts',
       'Fwd Seg Size Min', 'Active Mean', 'Active Std', 'Active Max',
       'Active Min', 'Idle Mean', 'Idle Std', 'Idle Max', 'Idle Min', 'Label'],
      dtype='object')
Starting dataframe shape: (2097150, 80)
Counted rows by protocol: Counter({6: 1513795, 17: 552908, 0: 30447})
Removing non-TCP rows…
Remove 'DoS attacks-Slowloris' rows
Removing 'Protocol' and 'Timestamp' columns
column: 'Dst Port', unique values: 28441
column: 'Flow Duration', unique values: 715356
column: 'Tot Fwd Pkts', unique values: 1030
column: 'Tot Bwd Pkts', unique values: 1552
column: 'TotLen Fwd Pkts', unique values: 9091
column: 'TotLen Bwd Pkts', unique values: 30162
column: 'Fwd Pkt Len Max', unique values: 1442
column: 'Fwd Pkt Len Min', unique values: 87
column: 'Fwd Pkt Len Mean', unique values: 33418
column: 'Fwd Pkt Len Std', unique values: 65213
```

```
column: 'Bwd Pkt Len Max', unique values: 1190
column: 'Bwd Pkt Len Min', unique values: 81
column: 'Bwd Pkt Len Mean', unique values: 52921
column: 'Bwd Pkt Len Std', unique values: 67594
column: 'Flow Byts/s', unique values: 655633
column: 'Flow Pkts/s', unique values: 758242
column: 'Flow IAT Mean', unique values: 756127
column: 'Flow IAT Std', unique values: 754015
column: 'Flow IAT Max', unique values: 529697
column: 'Flow IAT Min', unique values: 104083
column: 'Fwd IAT Tot', unique values: 664631
column: 'Fwd IAT Mean', unique values: 699138
column: 'Fwd IAT Std', unique values: 664768
column: 'Fwd IAT Max', unique values: 545788
column: 'Fwd IAT Min', unique values: 136202
column: 'Bwd IAT Tot', unique values: 567307
column: 'Bwd IAT Mean', unique values: 589298
column: 'Bwd IAT Std', unique values: 639881
column: 'Bwd IAT Max', unique values: 407882
column: 'Bwd IAT Min', unique values: 170427
column: 'Fwd PSH Flags', unique values: 2
Removing column Fwd PSH Flags
Column values: Counter({0: 1424761, 1: 78044})
column: 'Bwd PSH Flags', unique values: 1
Removing column Bwd PSH Flags
Column values: Counter({0: 1502805})
column: 'Fwd URG Flags', unique values: 1
Removing column Fwd URG Flags
Column values: Counter({0: 1502805})
column: 'Bwd URG Flags', unique values: 1
Removing column Bwd URG Flags
Column values: Counter({0: 1502805})
column: 'Fwd Header Len', unique values: 1905
column: 'Bwd Header Len', unique values: 2955
column: 'Fwd Pkts/s', unique values: 749197
column: 'Bwd Pkts/s', unique values: 653274
column: 'Pkt Len Min', unique values: 21
column: 'Pkt Len Max', unique values: 1346
column: 'Pkt Len Mean', unique values: 79277
column: 'Pkt Len Std', unique values: 102091
column: 'Pkt Len Var', unique values: 102444
column: 'FIN Flag Cnt', unique values: 2
Removing column FIN Flag Cnt
Column values: Counter({0: 1493032, 1: 9773})
column: 'SYN Flag Cnt', unique values: 2
Removing column SYN Flag Cnt
Column values: Counter({0: 1424761, 1: 78044})
column: 'RST Flag Cnt', unique values: 2
```

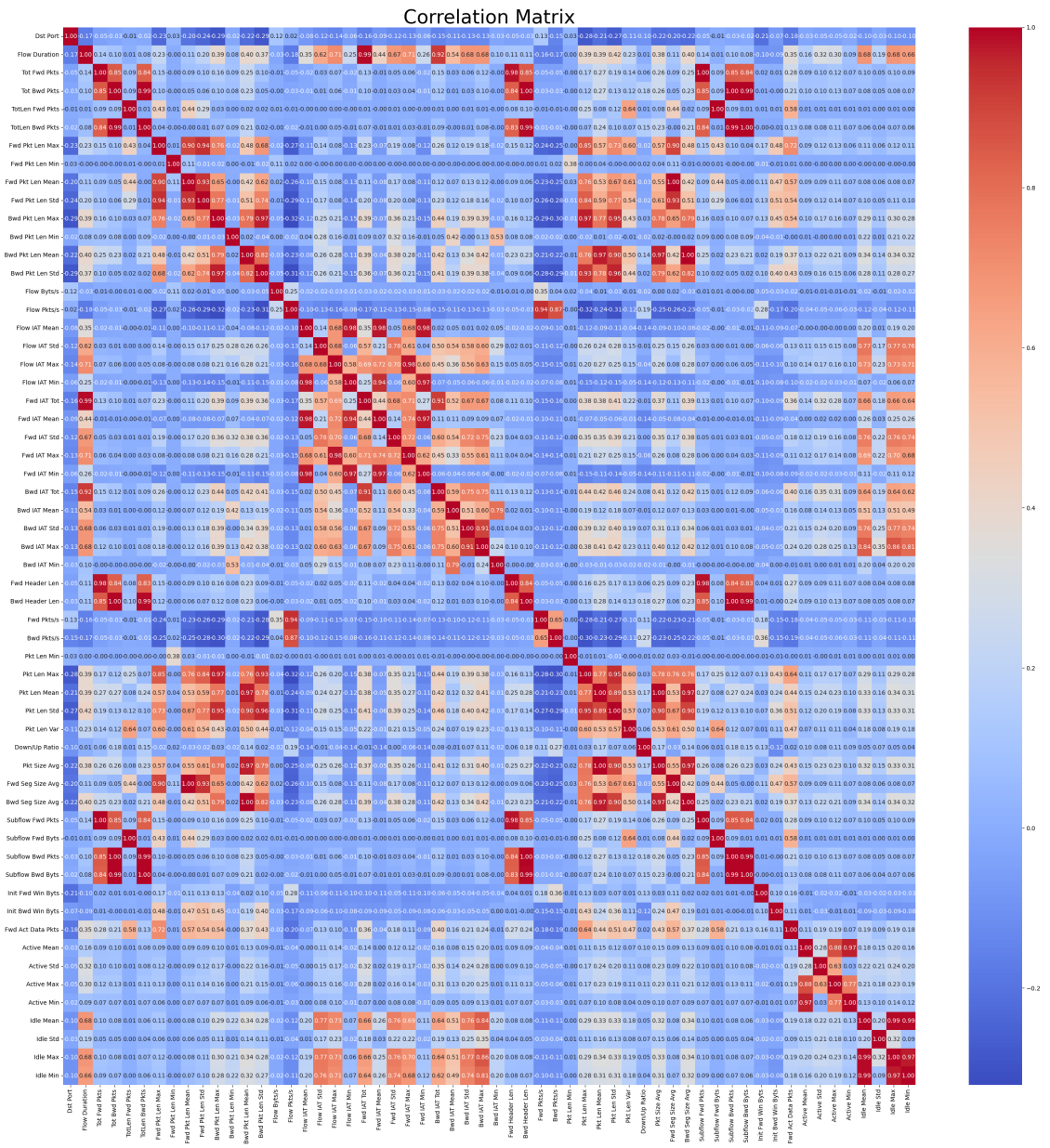
```
Removing column RST Flag Cnt
Column values: Counter({0: 1397328, 1: 105477})
column: 'PSH Flag Cnt', unique values: 2
Removing column PSH Flag Cnt
Column values: Counter({1: 924841, 0: 577964})
column: 'ACK Flag Cnt', unique values: 2
Removing column ACK Flag Cnt
Column values: Counter({0: 930654, 1: 572151})
column: 'URG Flag Cnt', unique values: 2
Removing column URG Flag Cnt
Column values: Counter({0: 1325783, 1: 177022})
column: 'CWE Flag Count', unique values: 1
Removing column CWE Flag Count
Column values: Counter({0: 1502805})
column: 'ECE Flag Cnt', unique values: 2
Removing column ECE Flag Cnt
Column values: Counter({0: 1397331, 1: 105474})
column: 'Down/Up Ratio', unique values: 57
column: 'Pkt Size Avg', unique values: 78861
column: 'Fwd Seg Size Avg', unique values: 33418
column: 'Bwd Seg Size Avg', unique values: 52917
column: 'Fwd Byts/b Avg', unique values: 1
Removing column Fwd Byts/b Avg
Column values: Counter({0: 1502805})
column: 'Fwd Pkts/b Avg', unique values: 1
Removing column Fwd Pkts/b Avg
Column values: Counter({0: 1502805})
column: 'Fwd Blk Rate Avg', unique values: 1
Removing column Fwd Blk Rate Avg
Column values: Counter({0: 1502805})
column: 'Bwd Byts/b Avg', unique values: 1
Removing column Bwd Byts/b Avg
Column values: Counter({0: 1502805})
column: 'Bwd Pkts/b Avg', unique values: 1
Removing column Bwd Pkts/b Avg
Column values: Counter({0: 1502805})
column: 'Bwd Blk Rate Avg', unique values: 1
Removing column Bwd Blk Rate Avg
Column values: Counter({0: 1502805})
column: 'Subflow Fwd Pkts', unique values: 1030
column: 'Subflow Fwd Byts', unique values: 9091
column: 'Subflow Bwd Pkts', unique values: 1552
column: 'Subflow Bwd Byts', unique values: 30162
column: 'Init Fwd Win Byts', unique values: 5785
column: 'Init Bwd Win Byts', unique values: 6438
column: 'Fwd Act Data Pkts', unique values: 153
column: 'Fwd Seg Size Min', unique values: 9
Removing column Fwd Seg Size Min
```

```
Column values: Counter({20: 1059406, 32: 233664, 40: 206443, 28: 2948, 24: 244,
36: 53, 44: 43, 48: 3, 56: 1})
column: 'Active Mean', unique values: 185787
column: 'Active Std', unique values: 142028
column: 'Active Max', unique values: 174064
column: 'Active Min', unique values: 87786
column: 'Idle Mean', unique values: 206175
column: 'Idle Std', unique values: 145124
column: 'Idle Max', unique values: 152664
column: 'Idle Min', unique values: 173765
column: 'Label', unique values: 4
Correlation matrix
Removing column Fwd IAT Tot as its corellation to Flow Duration is
0.9929247702927323
Removing column Fwd Header Len as its corellation to Tot Fwd Pkts is
0.9830168283144606
Removing column Subflow Fwd Pkts as its corellation to Tot Fwd Pkts is 1.0
Removing column TotLen Bwd Pkts as its corellation to Tot Bwd Pkts is
0.994875600454904
Removing column Bwd Header Len as its corellation to Tot Bwd Pkts is
0.9995280693546675
Removing column Subflow Bwd Pkts as its corellation to Tot Bwd Pkts is 1.0
Removing column Subflow Bwd Byts as its corellation to Tot Bwd Pkts is
0.994875600454904
Removing column Subflow Fwd Byts as its corellation to TotLen Fwd Pkts is 1.0
Removing column Fwd Seg Size Avg as its corellation to Fwd Pkt Len Mean is 1.0
Removing column Bwd Pkt Len Std as its corellation to Bwd Pkt Len Max is
0.9683467506798784
Removing column Pkt Len Max as its corellation to Bwd Pkt Len Max is
0.966573013722215
Removing column Pkt Len Mean as its corellation to Bwd Pkt Len Mean is
0.969111152588415
Removing column Pkt Size Avg as its corellation to Bwd Pkt Len Mean is
0.9684307097410466
Removing column Bwd Seg Size Avg as its corellation to Bwd Pkt Len Mean is 1.0
Removing column Pkt Len Std as its corellation to Bwd Pkt Len Std is
0.9624456884201957
Removing column Flow IAT Min as its corellation to Flow IAT Mean is
0.9756217279860007
Removing column Fwd IAT Mean as its corellation to Flow IAT Mean is
0.9786094878395148
Removing column Fwd IAT Min as its corellation to Flow IAT Mean is
0.9761392417187464
Removing column Fwd IAT Max as its corellation to Flow IAT Max is
0.9772225653548794
Removing column Active Min as its corellation to Active Mean is
0.9673499638033494
Removing column Idle Max as its corellation to Idle Mean is 0.9923188803154489
```

```
Removing column Idle Min as its corellation to Idle Mean is 0.9927083431753968


###################################
Final data state for file data.parquet
Shape (rows, columns): (1490956, 37)
Label counts: Counter({'Benign': 1068505, 'FTP-BruteForce': 193354, 'SSH-
Bruteforce': 187589, 'DoS attacks-GoldenEye': 41508})
###################################
```

0.1.1 Initial orrelation matrix

[5]:
```
# Set up the matplotlib figure
plt.figure(figsize=(32, 32))

# Draw the heatmap with seaborn
sns.heatmap(correlation_matrix, annot=True, fmt='.2f', cmap='coolwarm')

# Set the title of the heatmap
plt.title('Correlation Matrix', fontsize=32)

plt.savefig('correlation_matrix.png', dpi='figure', bbox_inches='tight',␣
 ↪format="png")
```

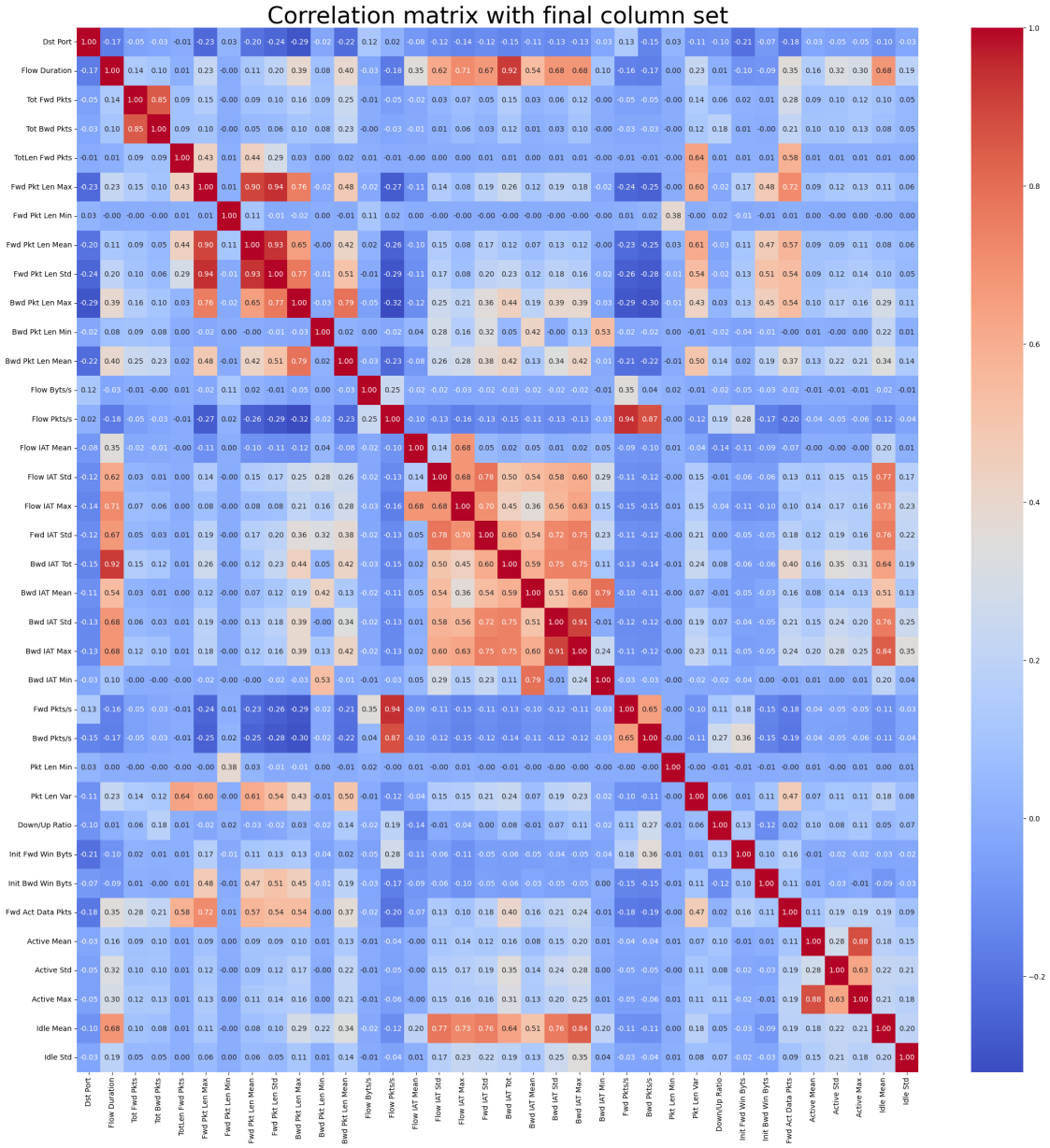


0.1.2 Final correlation matrix

[6]:
```
print(df.shape)
df_features = df.drop(columns=['Label'], inplace=False)

correlation_matrix = df_features.corr()

# Set up the matplotlib figure
plt.figure(figsize=(26, 26))
```

```
# Draw the heatmap with seaborn
sns.heatmap(correlation_matrix, annot=True, fmt='.2f', cmap='coolwarm')

# Set the title of the heatmap
plt.title('Correlation matrix with final column set', fontsize=32)

plt.savefig('final_correlation_matrix.png', dpi='figure', bbox_inches='tight',␣
 ↪format="png")
```

```
(1490956, 37)
```



0.2 Normalization

- use MinMaxScaler to normalize data
- save the fitted scaler to file, for usage on any new data to analyze

[7]:
```
from sklearn.preprocessing import MinMaxScaler
from pandas import DataFrame, concat
from numpy import isfinite
from pickle import dump

file = "data_pruned.parquet"

print(f"Reading file {file}")
df = read_parquet(file)
df_features = df.drop(columns=['Label'], inplace=False)

scaler = MinMaxScaler()

normalized_features = DataFrame(scaler.fit_transform(df_features),␣
 ↪columns=df_features.columns)

with open("min_max_scaler_fitted.pickle", 'wb') as file_output:
    dump(scaler, file_output)

# Combine the normalized numeric data with the non-numeric data
df = concat([normalized_features, df['Label']], axis=1, ignore_index=False)
print("Example row after normalization:")
# df = df.reset_index(drop=True)
print(df.head(1))
print(df.shape)
print("dtypes:")
print(df.dtypes)
df.to_parquet(append_suffix(file, '_normalized'))
```

```
Reading file data_pruned.parquet
Example row after normalization:
   Dst Port  Flow Duration  Tot Fwd Pkts  Tot Bwd Pkts  TotLen Fwd Pkts  \
0  0.000259        0.31139      0.001441      0.000626         0.000248

   Fwd Pkt Len Max  Fwd Pkt Len Min  Fwd Pkt Len Mean  Fwd Pkt Len Std  \
0         0.011049              0.0          0.009369          0.01385

   Bwd Pkt Len Max  Bwd Pkt Len Min  Bwd Pkt Len Mean   Flow Byts/s  \
0         0.360414              0.0          0.170922  1.063669e-07

    Flow Pkts/s  Flow IAT Mean  Flow IAT Std  Flow IAT Max  Fwd IAT Std  \
0  1.697423e-07       0.012582      0.046197      0.130552     0.060543
```

```
   Bwd IAT Tot  Bwd IAT Mean  Bwd IAT Std  Bwd IAT Max  Bwd IAT Min  \
0      0.31139      0.035225     0.065928     0.133203     0.000009

     Fwd Pkts/s    Bwd Pkts/s  Pkt Len Min  Pkt Len Var  Down/Up Ratio  \
0  9.073749e-08  1.605705e-07          0.0     0.000916            0.0

   Init Fwd Win Byts  Init Bwd Win Byts  Fwd Act Data Pkts  Active Mean  \
0           0.445563           0.003555            0.00545     0.009147

   Active Std  Active Max  Idle Mean  Idle Std   Label
0    0.010785    0.014297   0.095644  0.056435  Benign
(1490956, 37)
dtypes:
Dst Port                   float64
Flow Duration              float64
Tot Fwd Pkts               float64
Tot Bwd Pkts               float64
TotLen Fwd Pkts            float64
Fwd Pkt Len Max            float64
Fwd Pkt Len Min            float64
Fwd Pkt Len Mean           float64
Fwd Pkt Len Std            float64
Bwd Pkt Len Max            float64
Bwd Pkt Len Min            float64
Bwd Pkt Len Mean           float64
Flow Byts/s                float64
Flow Pkts/s                float64
Flow IAT Mean              float64
Flow IAT Std               float64
Flow IAT Max               float64
Fwd IAT Std                float64
Bwd IAT Tot                float64
Bwd IAT Mean               float64
Bwd IAT Std                float64
Bwd IAT Max                float64
Bwd IAT Min                float64
Fwd Pkts/s                 float64
Bwd Pkts/s                 float64
Pkt Len Min                float64
Pkt Len Var                float64
Down/Up Ratio              float64
Init Fwd Win Byts          float64
Init Bwd Win Byts          float64
Fwd Act Data Pkts          float64
Active Mean                float64
Active Std                 float64
Active Max                 float64
Idle Mean                  float64
```

```
Idle Std                    float64
Label                string[python]
dtype: object
```

[ ]: