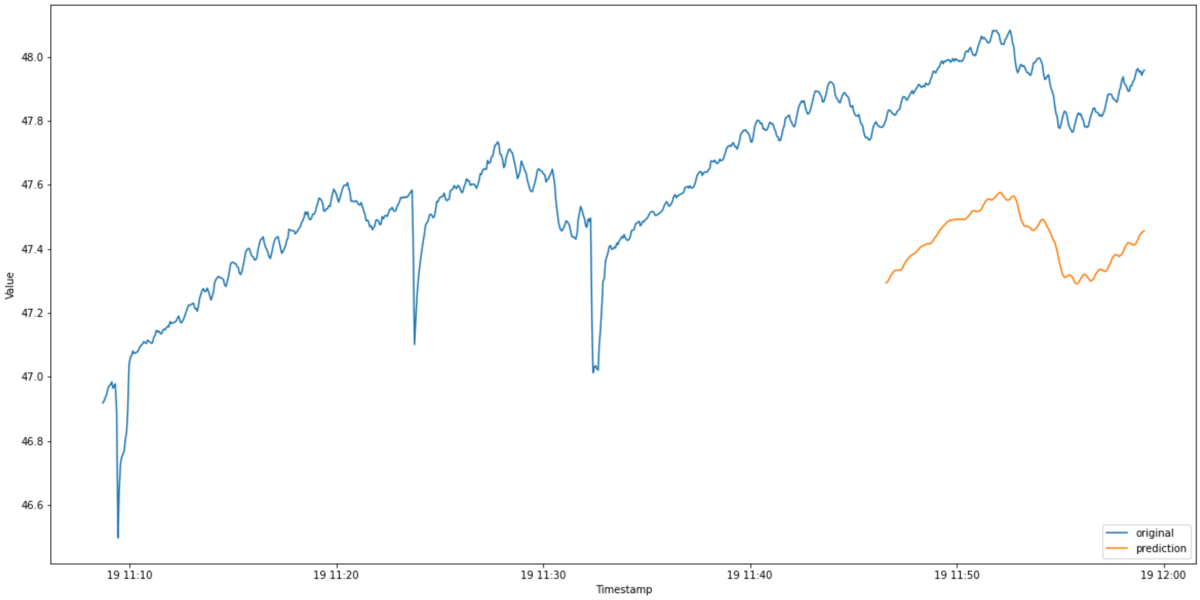
Time Series 0:

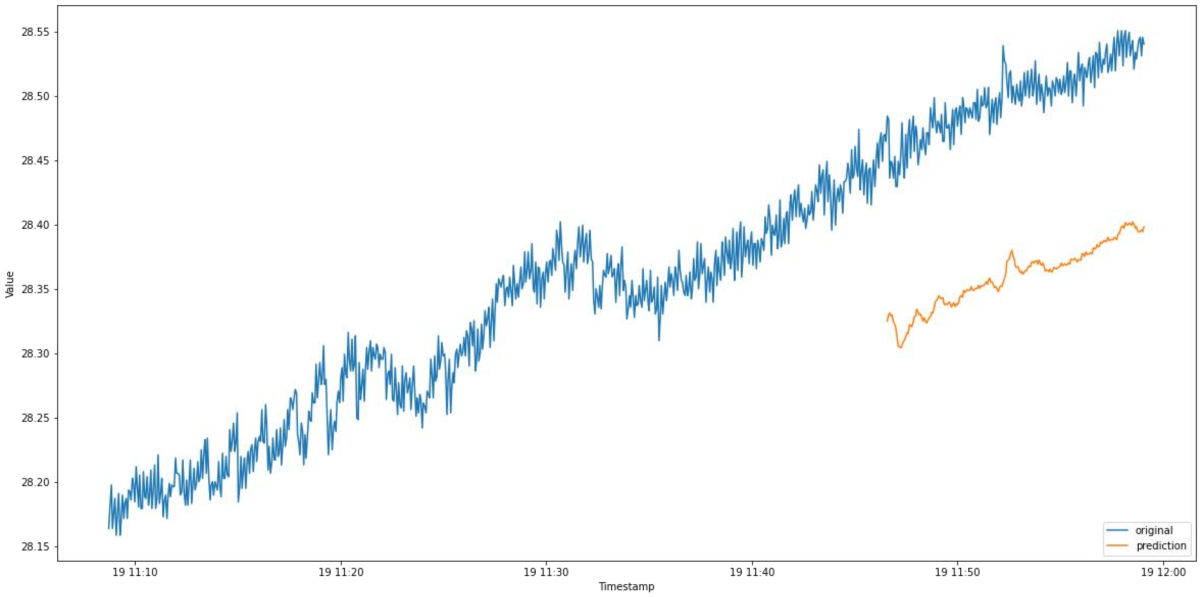
Plot:




```
Score:
        {'mae': 0.49806014,
         'mse': 0.24875183,
         'rmse': 0.49875027,
         'mape': 1.039231}
```

Time Series 1:

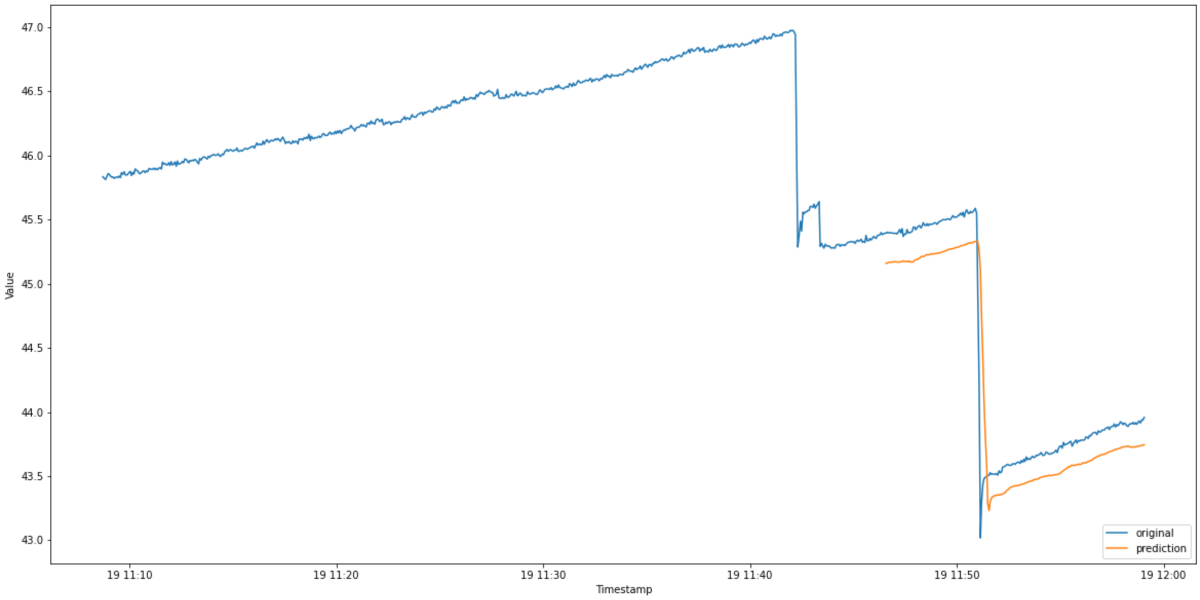
Plot:




```
      Score: {'mae': 0.13973509,
'mse': 0.019673182,
'rmse': 0.14026111,
'mape': 0.49029192}
```

Time Series 2:

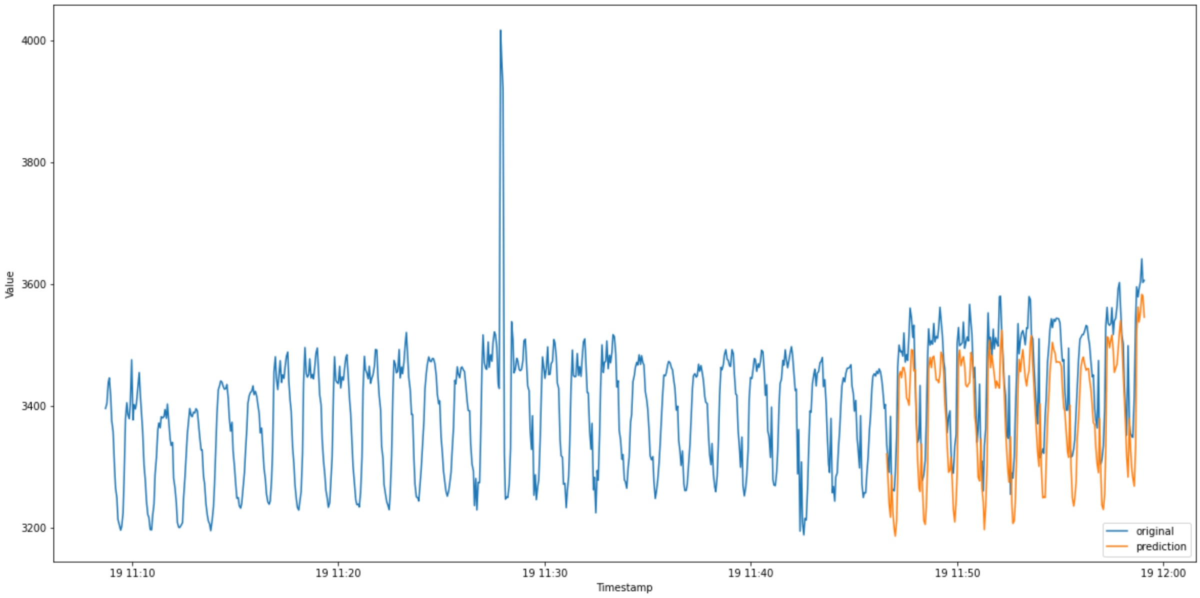
Plot:




```
       Score: {'mae': 0.23049092,
 'mse': 0.085161366,
 'rmse': 0.2918242,
'mape': 0.5200823}
```

Time Series 3:

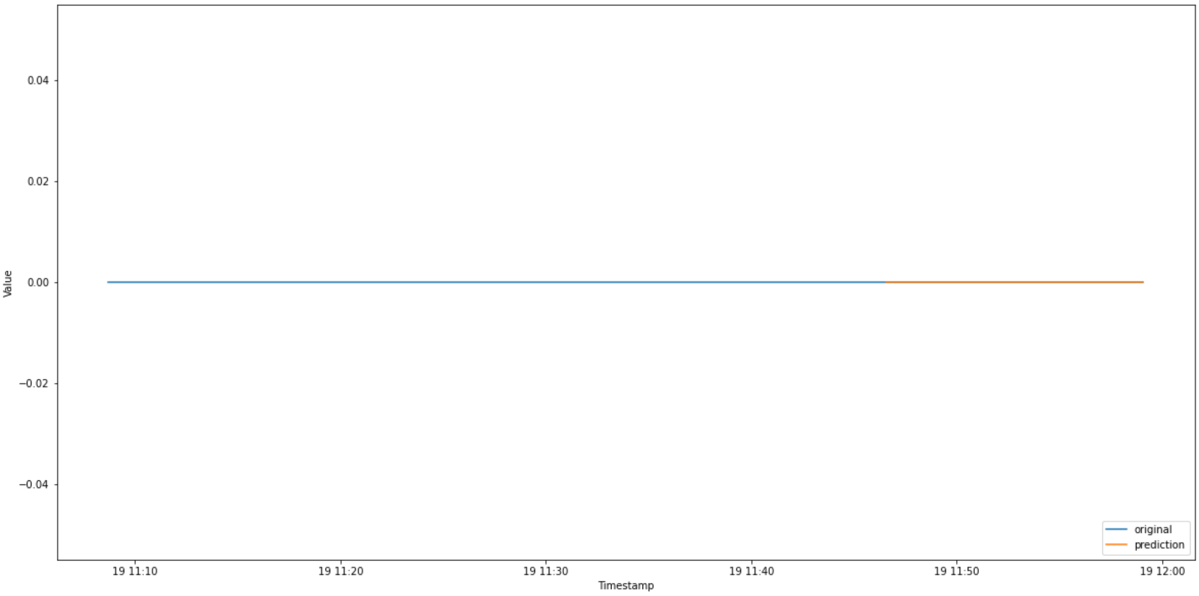
Plot:




```
       Score: {'mae': 67.306595,
 'mse': 6675.394,
 'rmse': 81.70309,
'mape': 1.9487281}
```

Time Series 4:

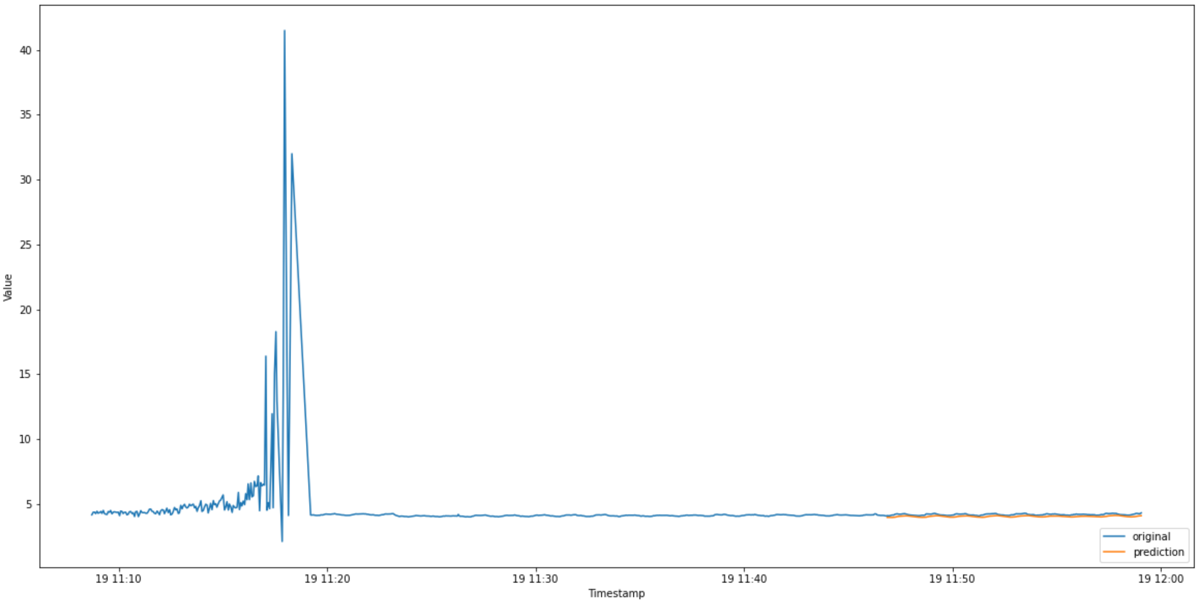
Plot:




```
      Score: {'mae': 0.0,
'mse': 0.0,
'rmse': 0.0,
 'mape': 0.0}
```

Time Series 5:

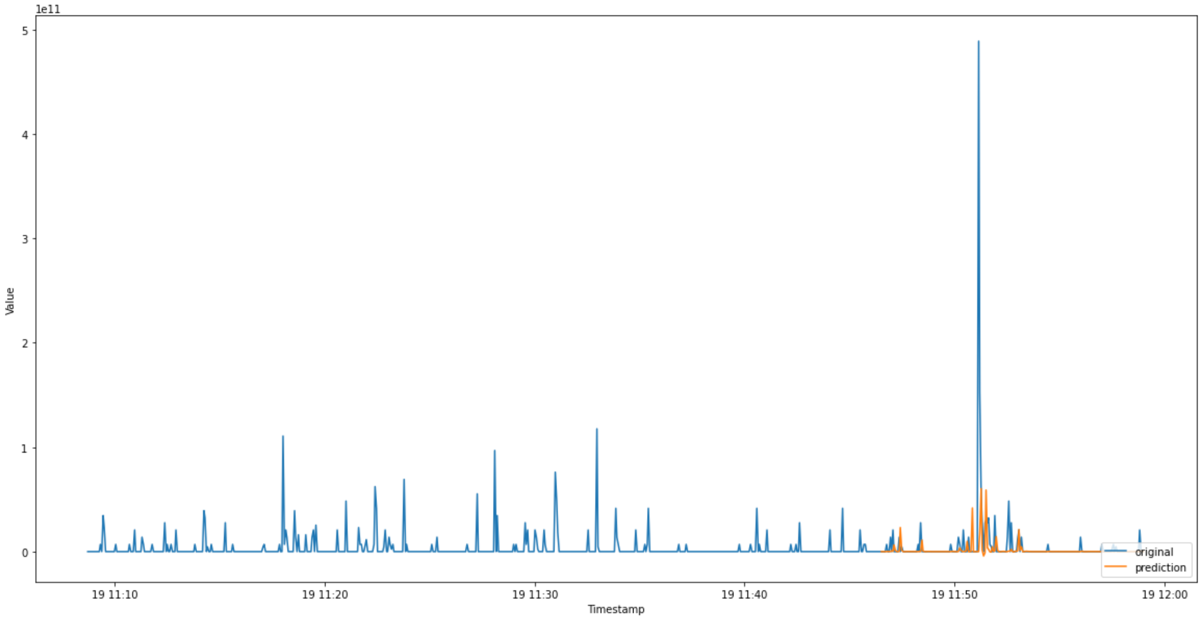
Plot:




```
      Score: {'mae': 0.14471313,
'mse': 0.022491913,
'rmse': 0.14997303,
'mape': 3.4594226}
```

Time Series 6:

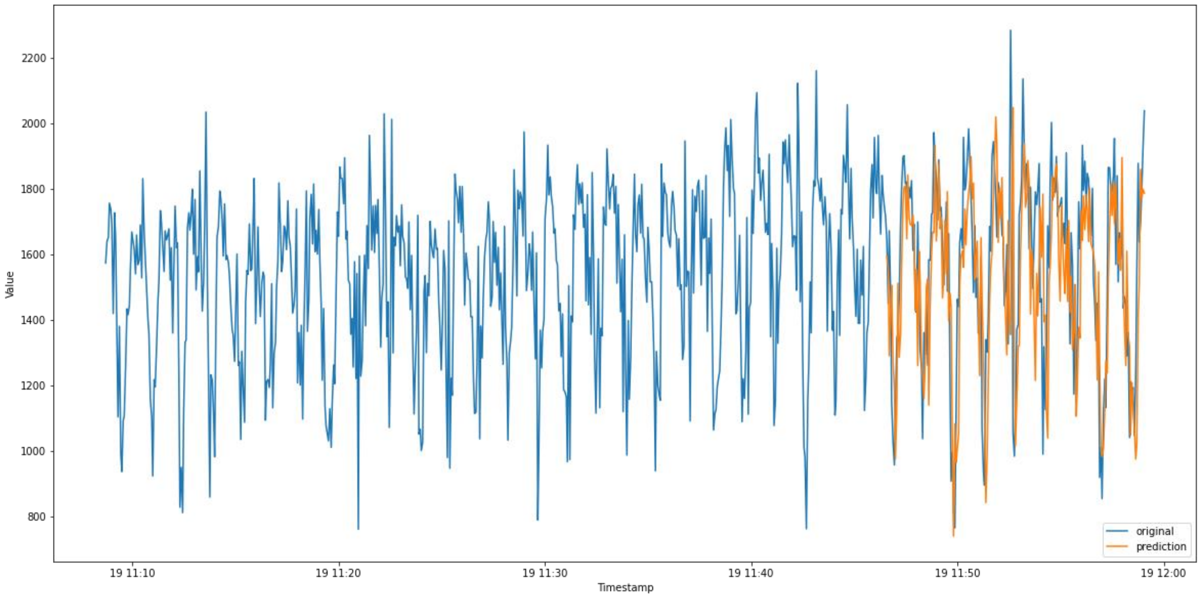
Plot:



Score: {'mae': 6093802000.0,
'mse': 1.3058174e+21,
'rmse': 36136096000.0,
'mape': 78142830.0}

Time Series 7:

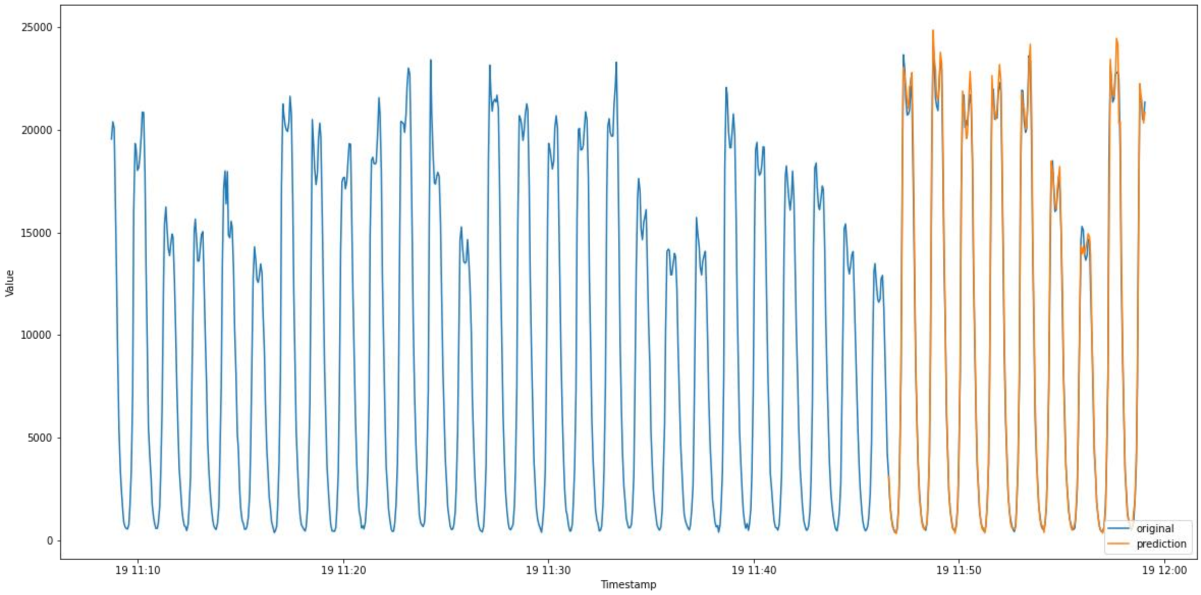
Plot:



Score: {'mae': 180.48619,
'mse': 56518.105,
'rmse': 237.73537,
'mape': 12.365687}

Time Series 8:

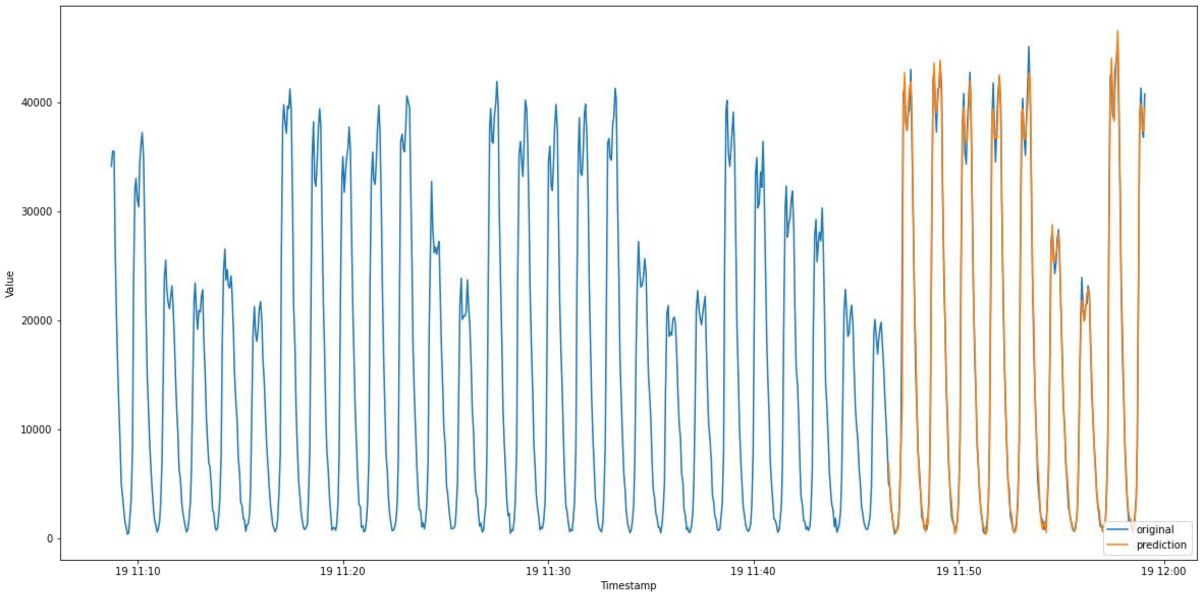
Plot:



Score: {'mae': 406.34543,
'mse': 380433.84,
'rmse': 616.7932,
'mape': 6.576588}

Time Series 9:

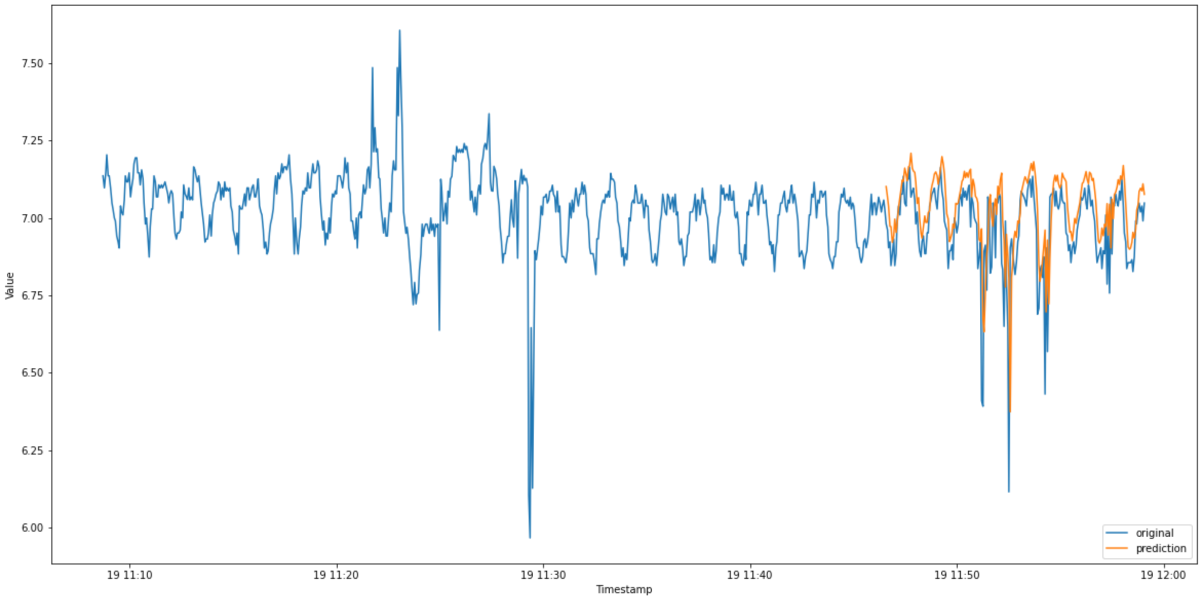
Plot:



Score: {'mae': 810.0666,
'mse': 1807310.9,
'rmse': 1344.3627,
'mape': 12.871523}

Time Series 10:

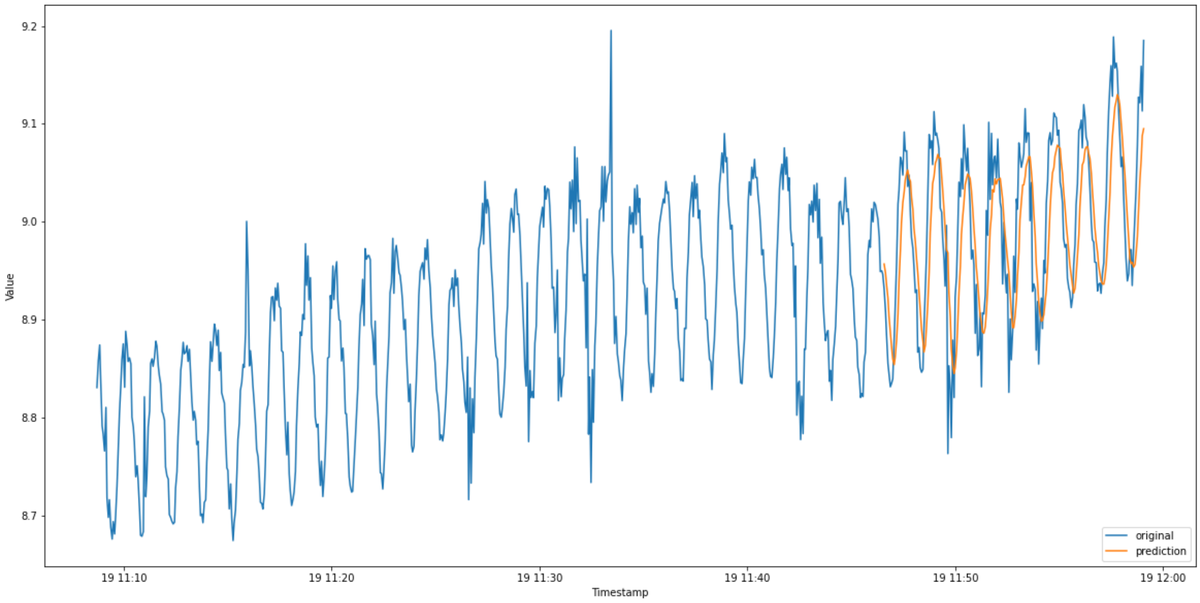
Plot:




```
Score: {'mae': 0.092252485,
 'mse': 0.017602913,
 'rmse': 0.13267598,
'mape': 1.3461926}
```

Time Series 11:

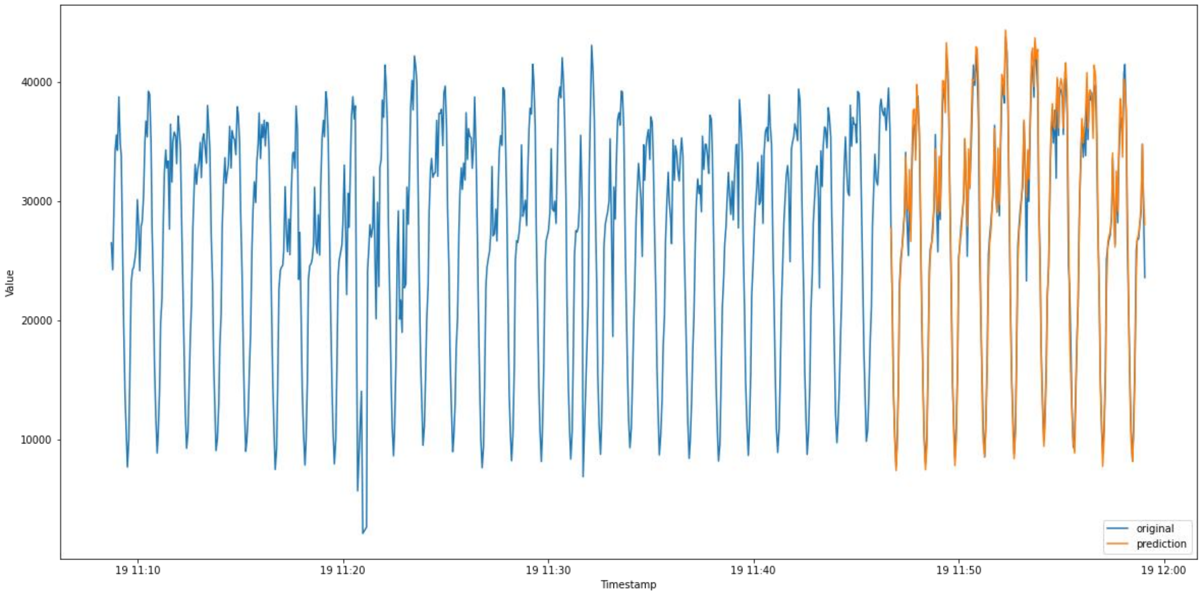
Plot:




```
Score: {'mae': 0.048872214,
 'mse': 0.0034413848,
 'rmse': 0.058663316,
 'mape': 0.5430168}
```

Time Series 12:

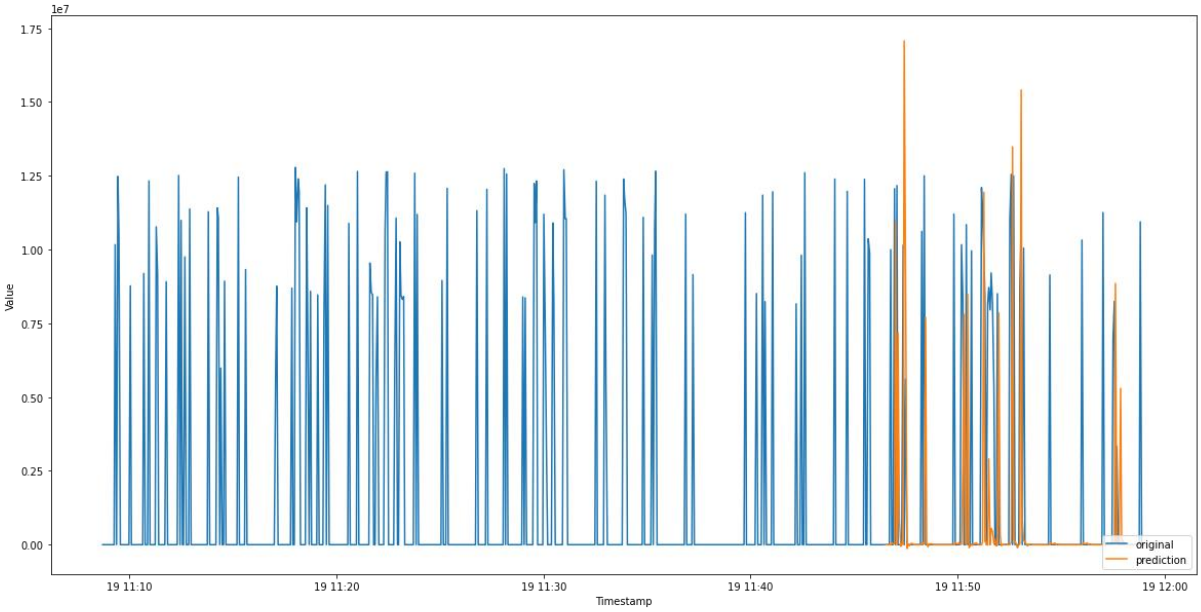
Plot:




```
Score: {'mae': 1017.9857,
'mse': 2067046.0,
 'rmse': 1437.7225,
'mape': 4.076213}
```

Time Series 13:

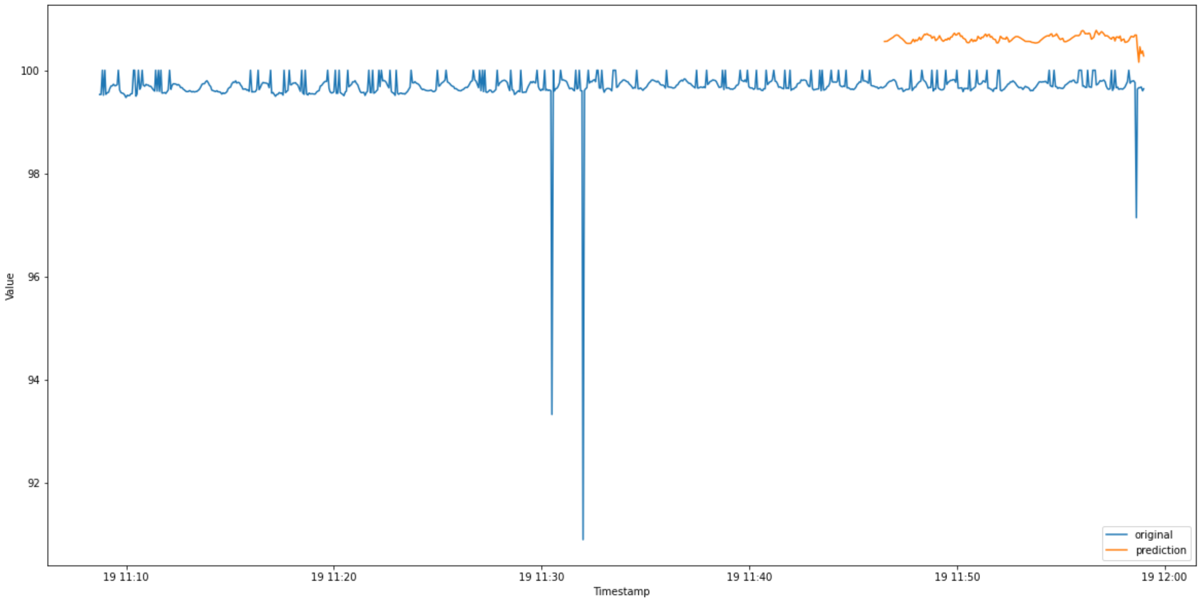
Plot:




```
Score: {'mae': 2036142.1,
'mse': 20134872000000.0,
'rmse': 4487189.5,
'mape': 13266906.0}
```

Time Series 14:

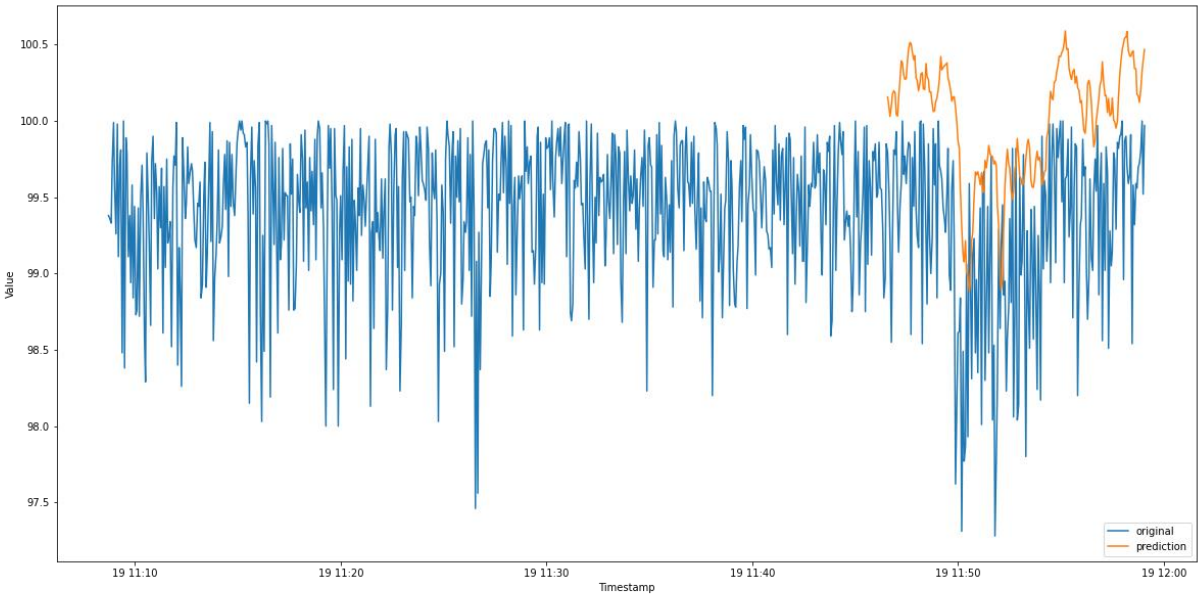
Plot:



Score: {'mae': 0.8977141,
'mse': 0.8548599,
'rmse': 0.92458636,
'mape': 0.9007063}

Time Series 15:

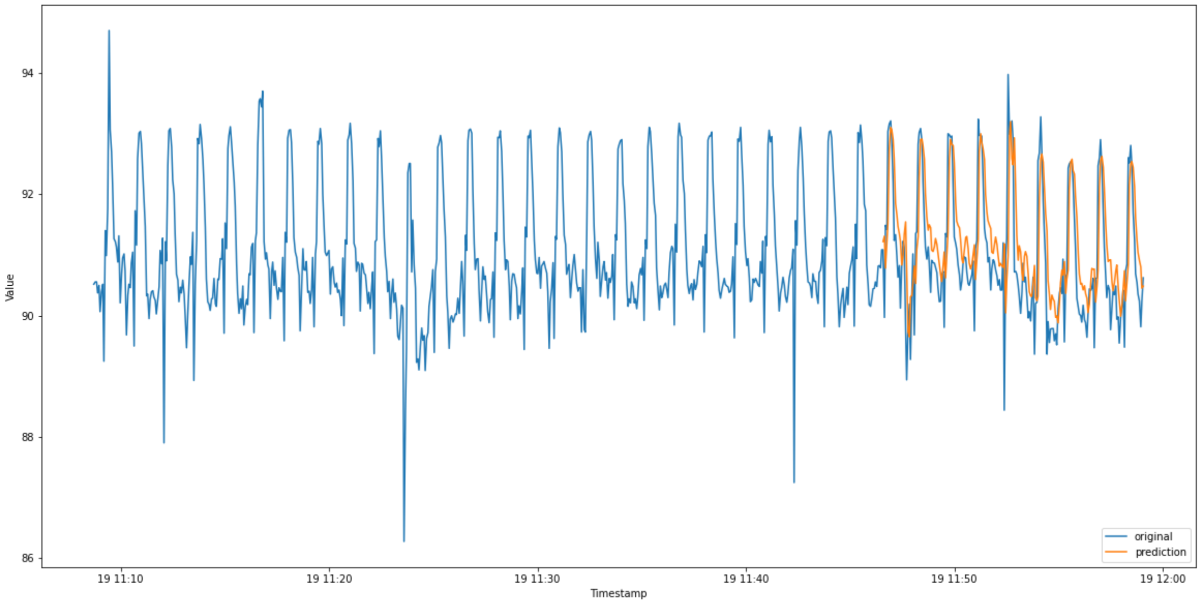
Plot:



Score: {'mae': 0.7860363,
'mse': 0.8758021,
'rmse': 0.935843,
'mape': 0.7945792}

Time Series 16:

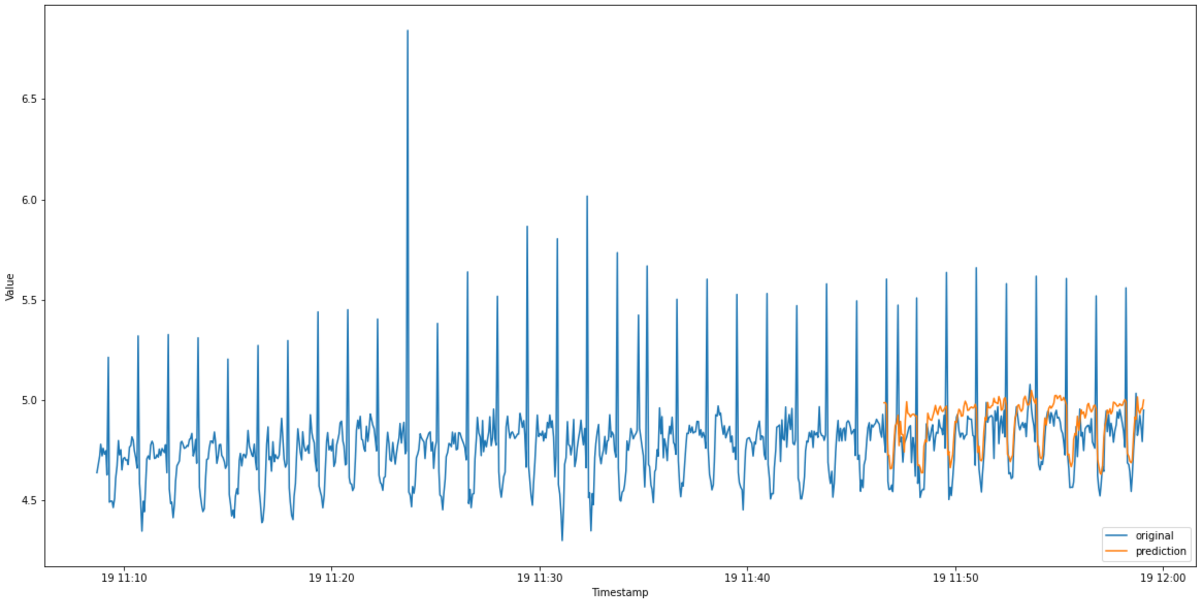
Plot:




```
Score: {'mae': 0.60037404,
'mse': 0.6213193,
'rmse': 0.7882381,
'mape': 0.6604153}
```

Time Series 17:

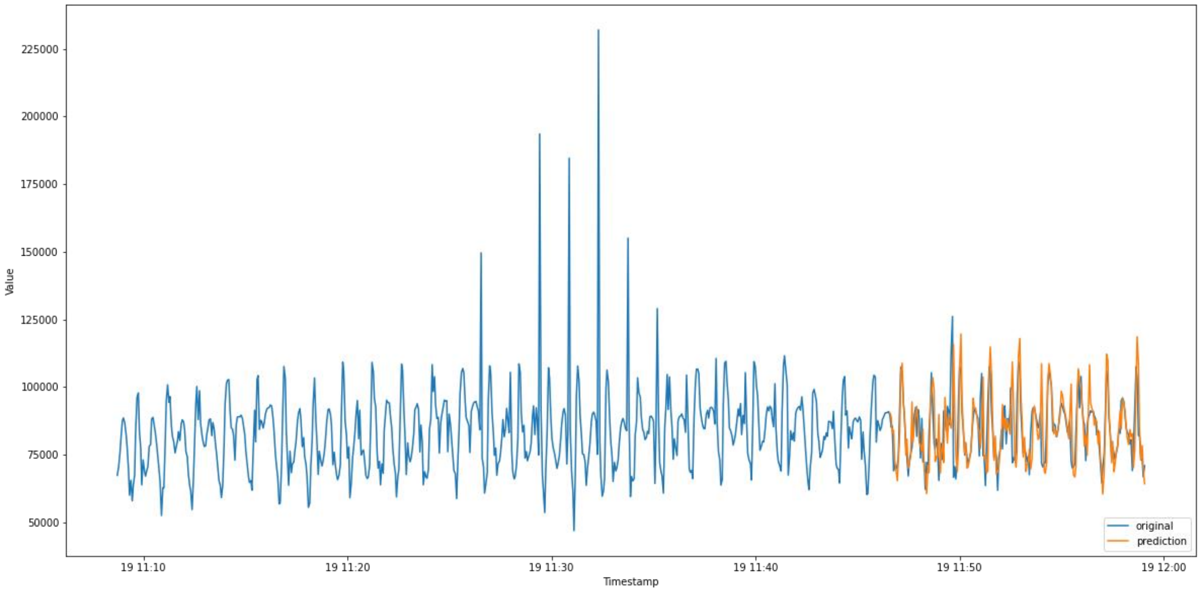
Plot:




```
Score: {'mae': 0.12561063,
'mse': 0.031377714,
'rmse': 0.17713755,
'mape': 2.5468004}
```

Time Series 18:

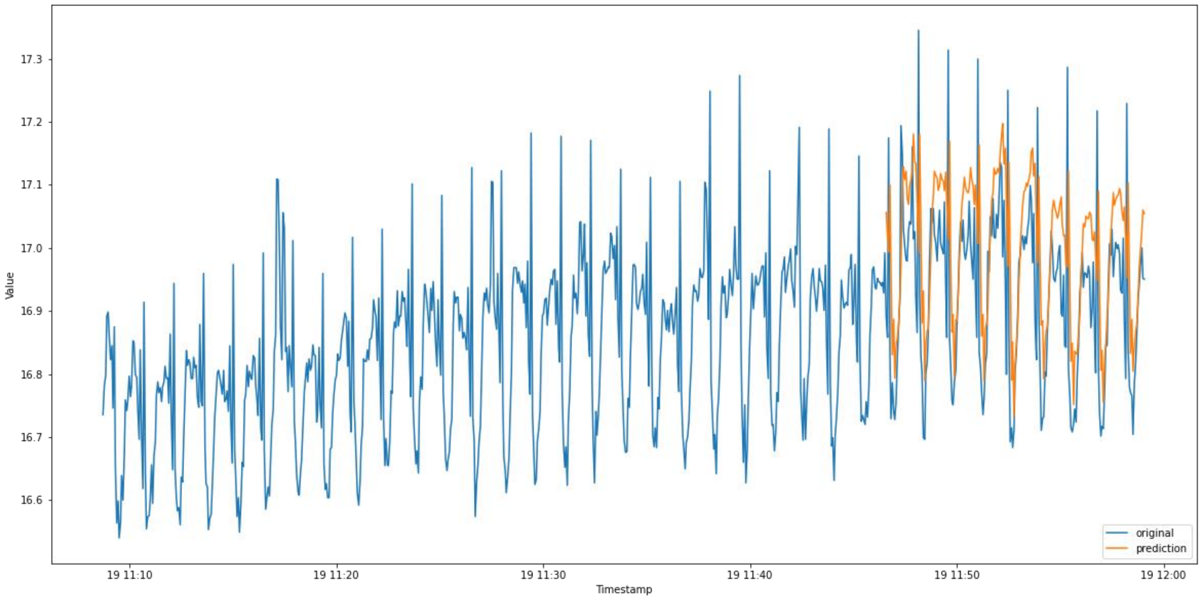
Plot:




```
Score: {'mae': 6625.828,
 'mse': 104180270.0,
'rmse': 10206.874,
'mape': 8.066024}
```

Time Series 19:

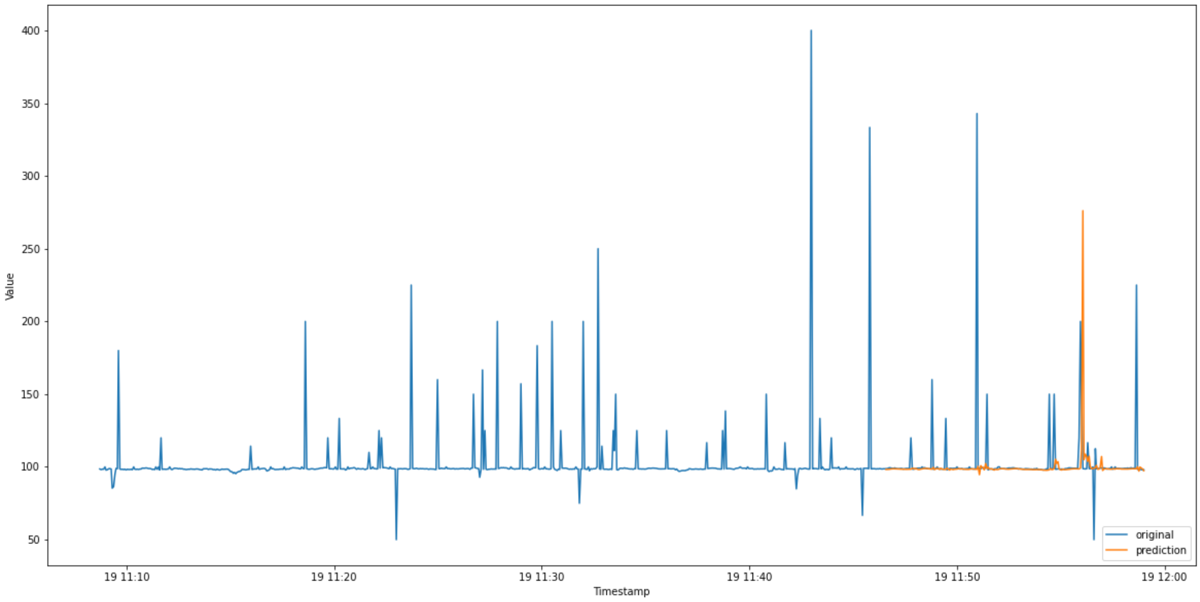
Plot:




```
Score: {'mae': 0.10584639,
 'mse': 0.017967751,
'rmse': 0.13404384,
'mape': 0.62530476}
```

Time Series 20:

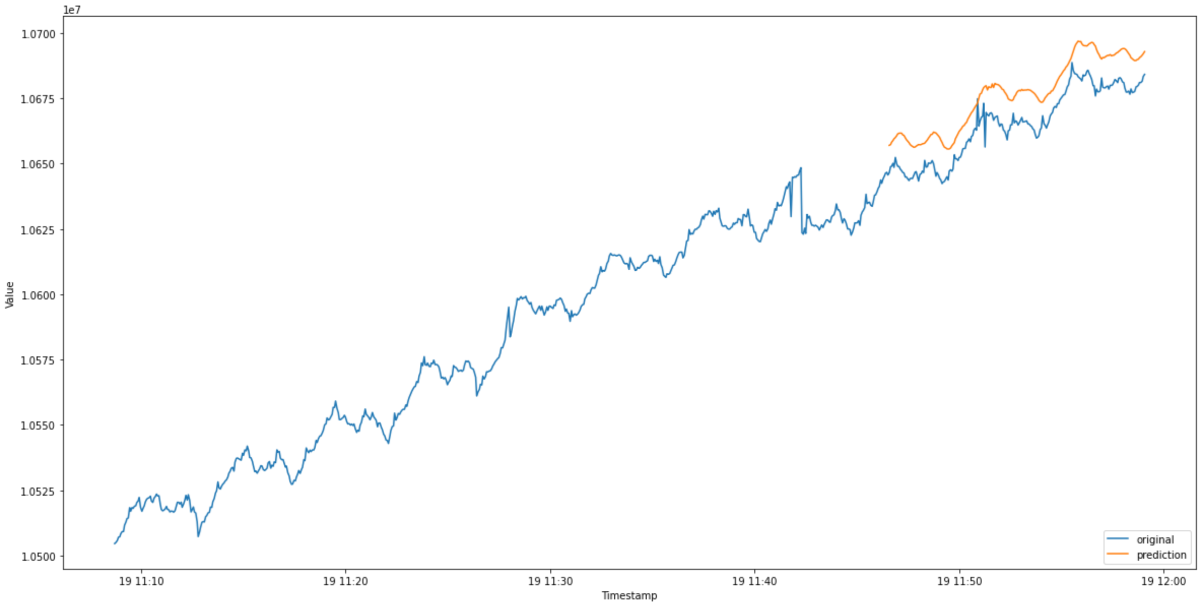
Plot:




```
Score: {'mae': 5.8487477,
'mse': 653.66724,
'rmse': 25.566917,
'mape': 4.1778164}
```

Time Series 21:

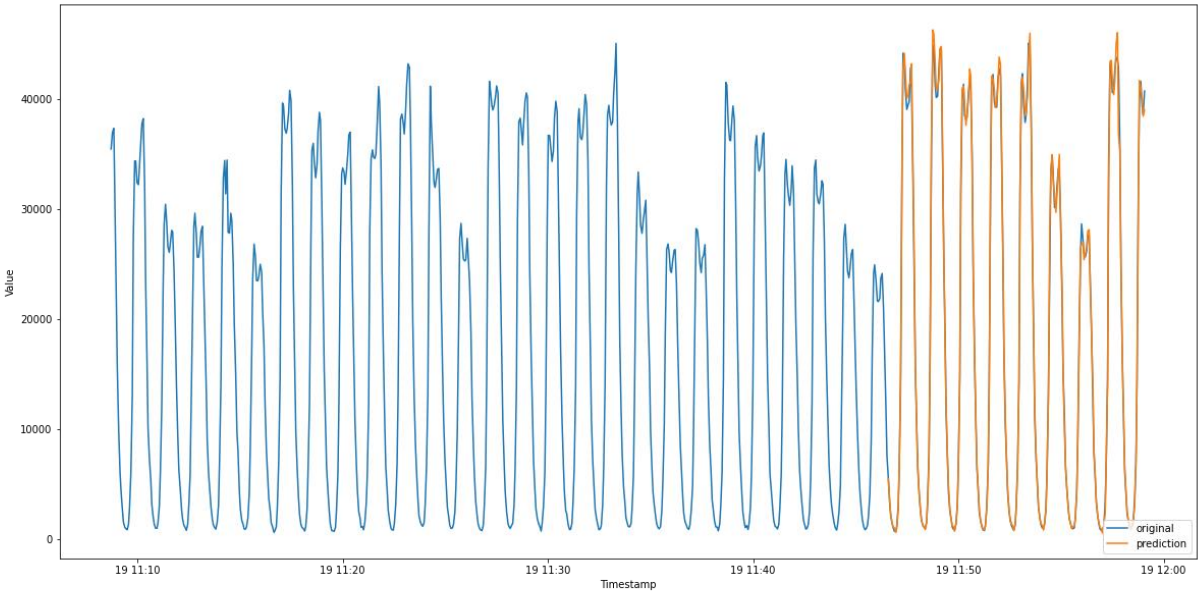
Plot:




```
Score: {'mae': 11317.01,
'mse': 135860640.0,
'rmse': 11655.928,
'mape': 0.10611574}
```

Time Series 22:

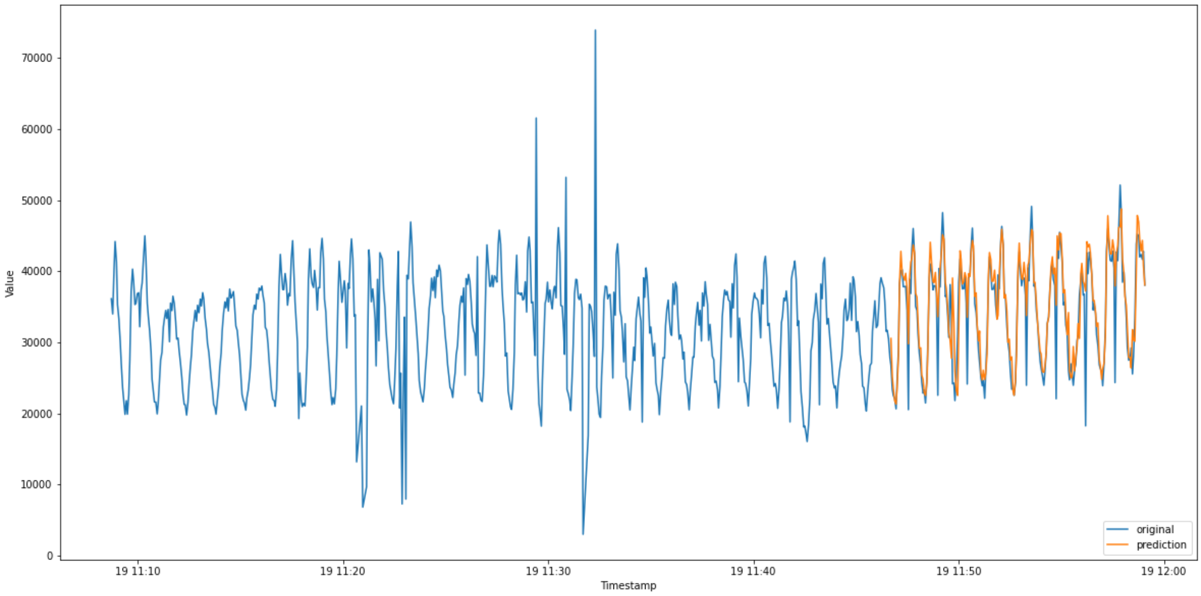
Plot:




```
Score: {'mae': 552.1356,
 'mse': 747765.06,
'rmse': 864.7341,
'mape': 5.521176}
```

Time Series 23:

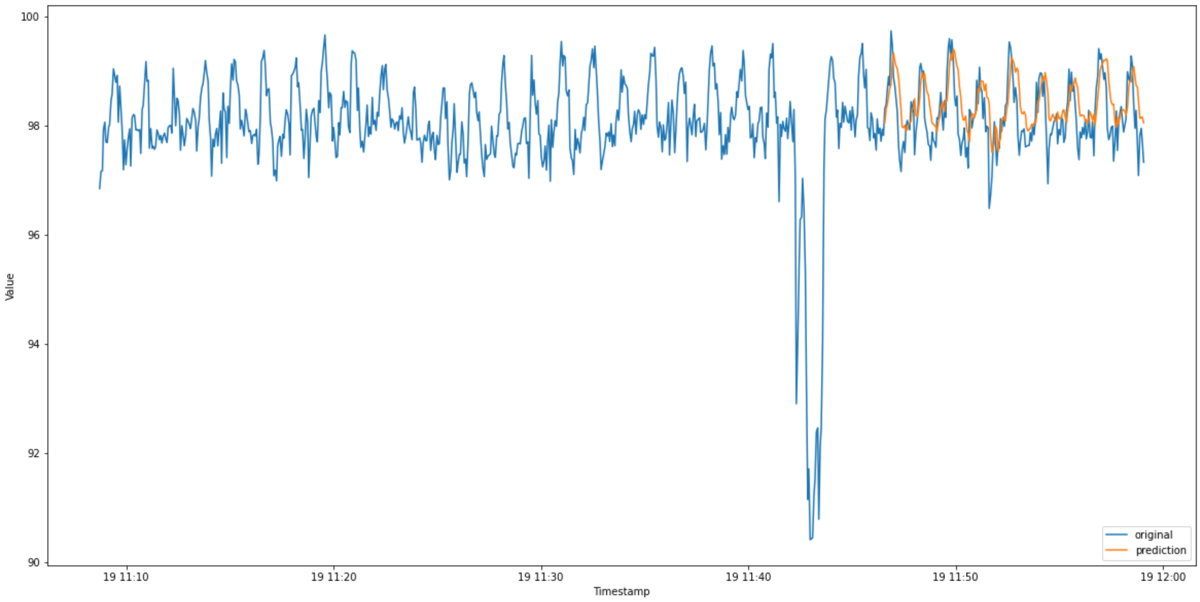
Plot:




```
Score: {'mae': 2076.4133,
 'mse': 11640534.0,
'rmse': 3411.8228,
'mape': 7.0215106}
```

Time Series 24:

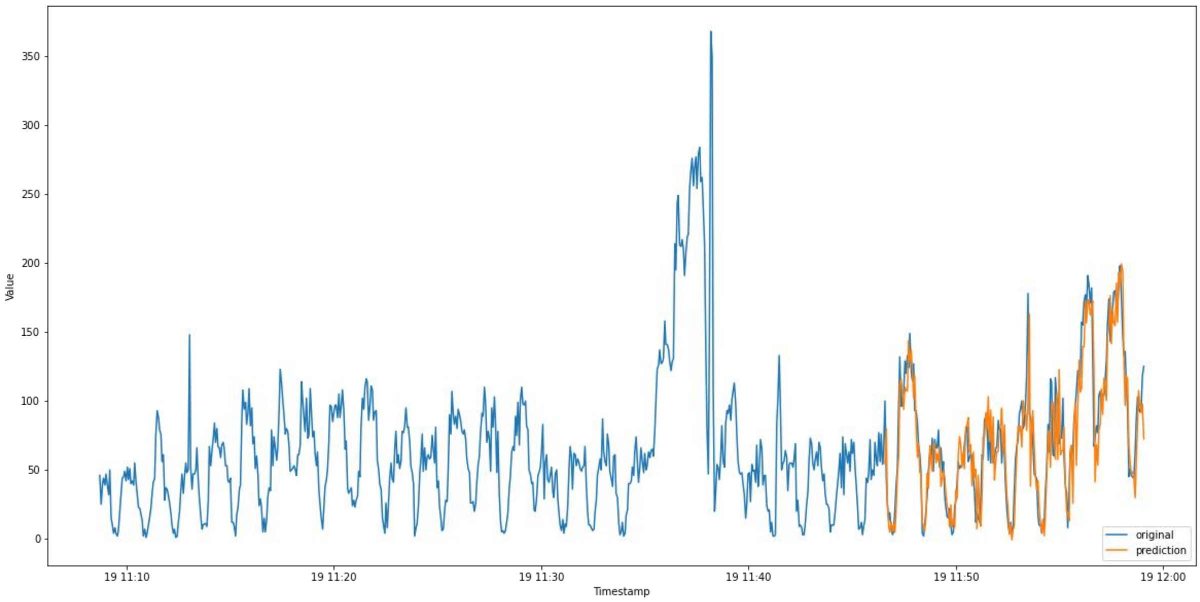
Plot:




```
        Score: {'mae': 0.42822286,
 'mse': 0.3010158,
'rmse': 0.5486491,
 'mape': 0.43694383}
```

Time Series 25:

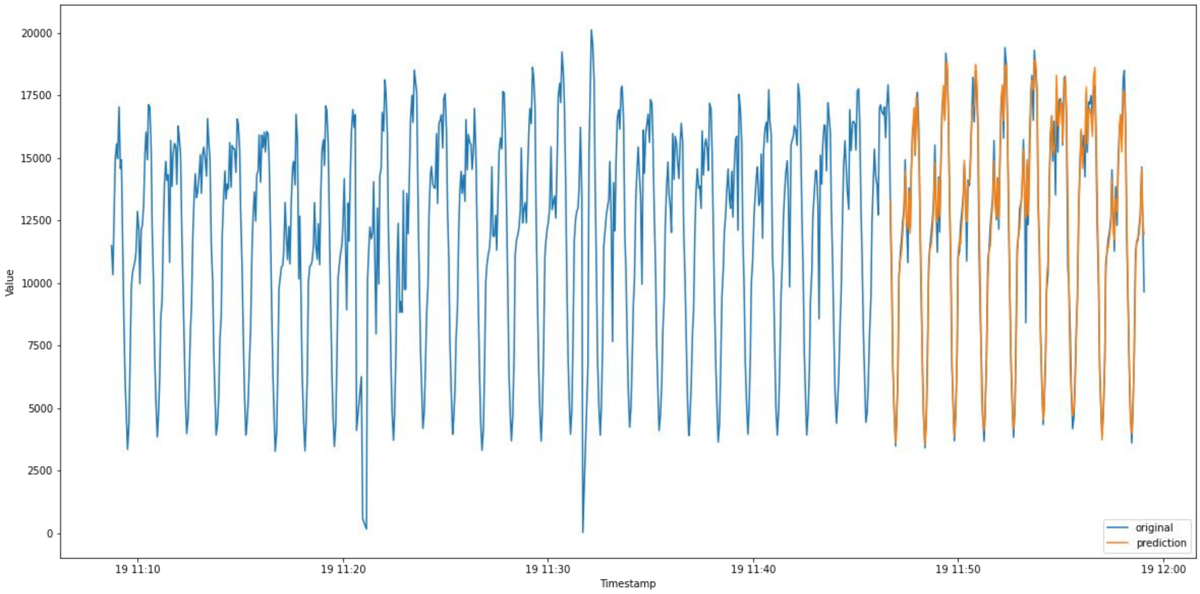
Plot:




```
        Score: {'mae': 16.893368,
'mse': 495.08542,
 'rmse': 22.250515,
'mape': 35.752575}
```

Time Series 26:

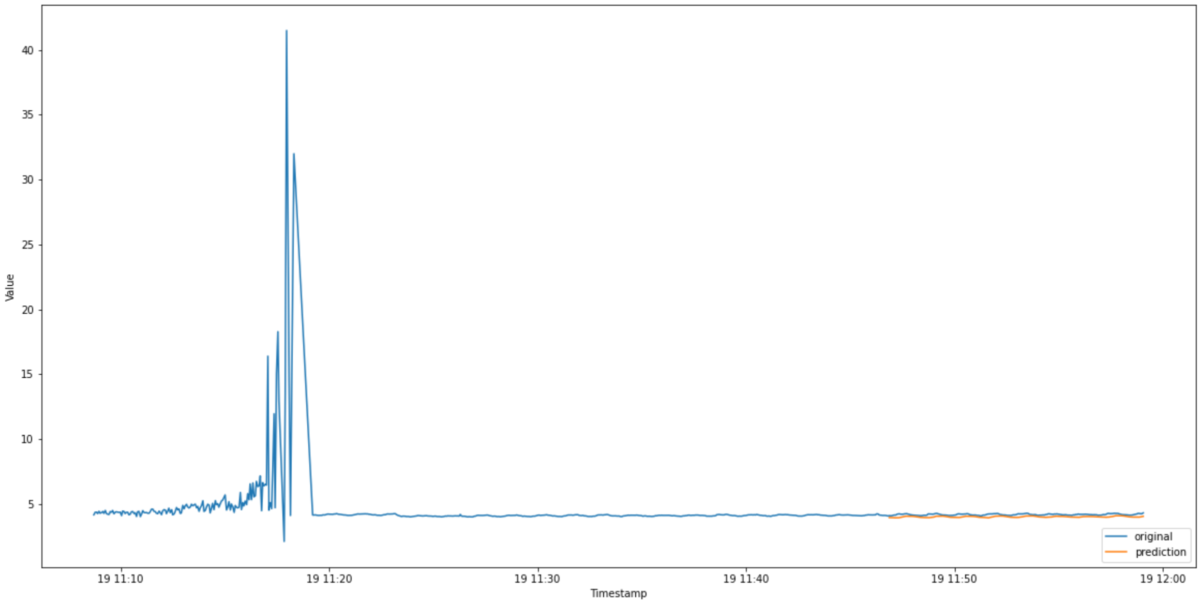
Plot:



Score: {'mae': 438.06924,
'mse': 427168.47,
'rmse': 653.58124,
'mape': 3.9723997}

Time Series 27:

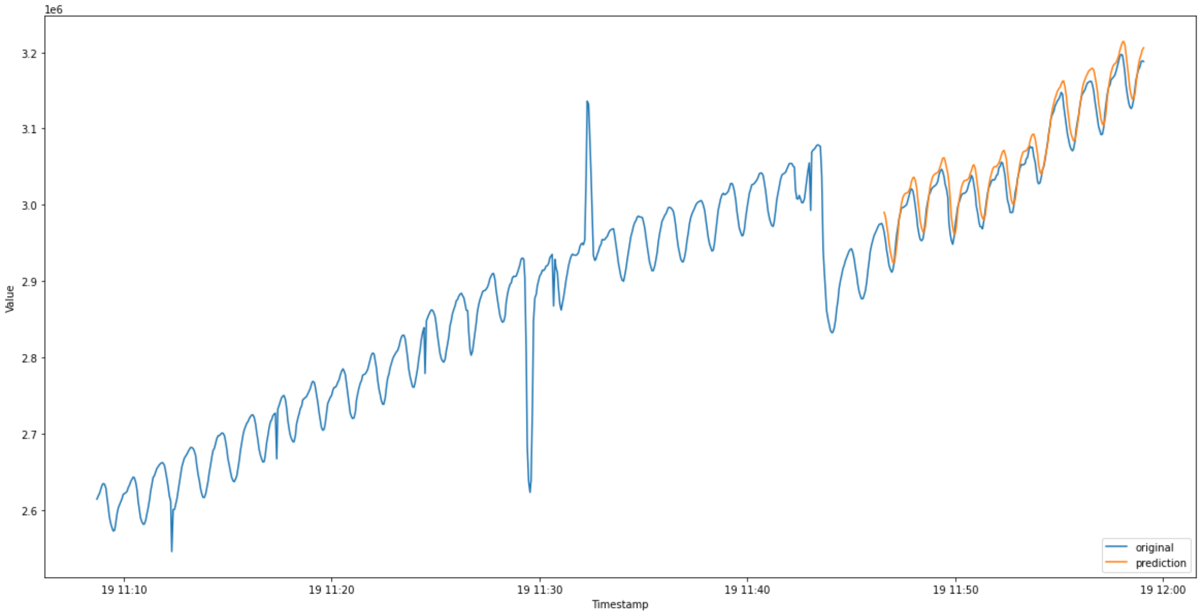
Plot:



Score: {'mae': 0.17367658,
'mse': 0.033233434,
'rmse': 0.18230039,
'mape': 4.1493883}

Time Series 28:

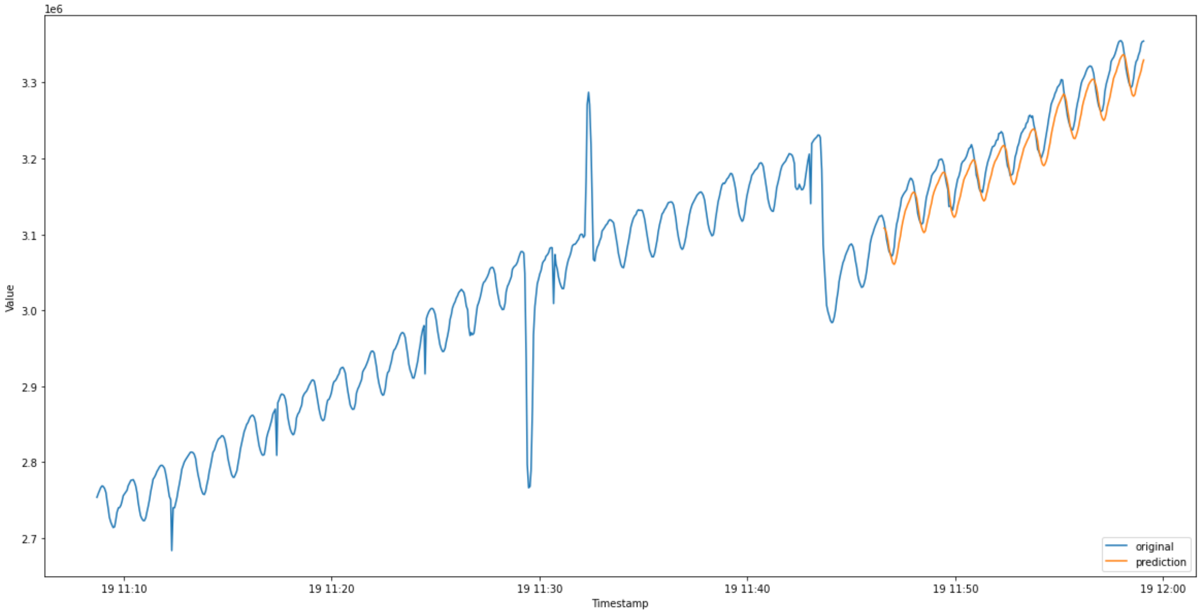
Plot:




```
Score: {'mae': 15514.5205,
'mse': 366389000.0,
'rmse': 19141.291,
'mape': 0.5088664}
```

Time Series 29:

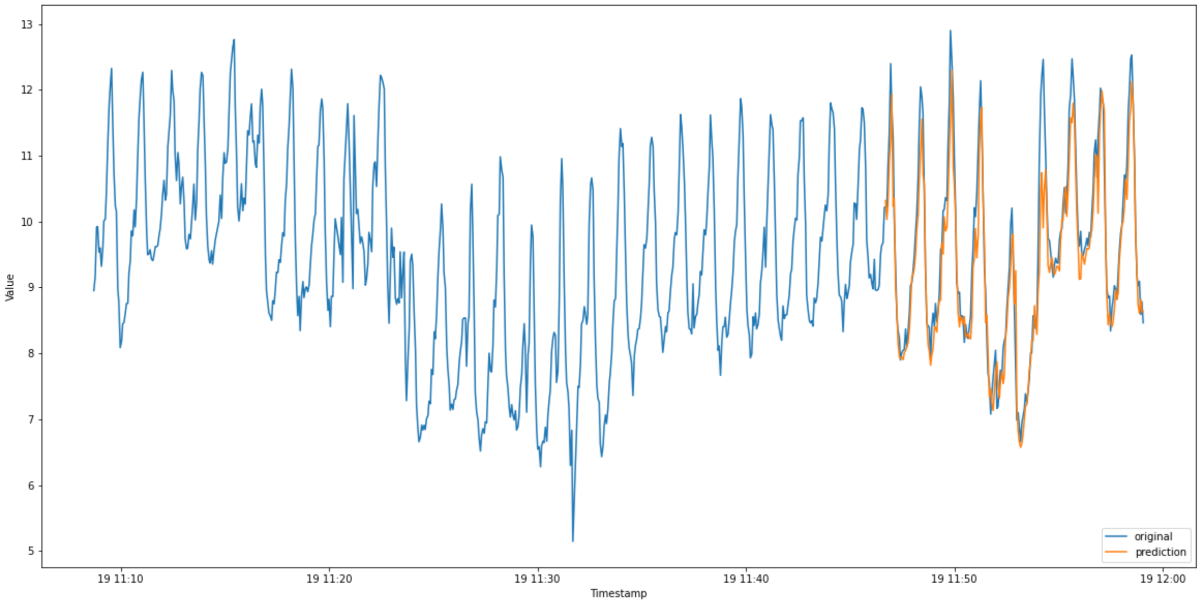
Plot:




```
Score: {'mae': 20974.666,
'mse': 590389440.0,
'rmse': 24297.932,
'mape': 0.64912534}
```

Time Series 30:

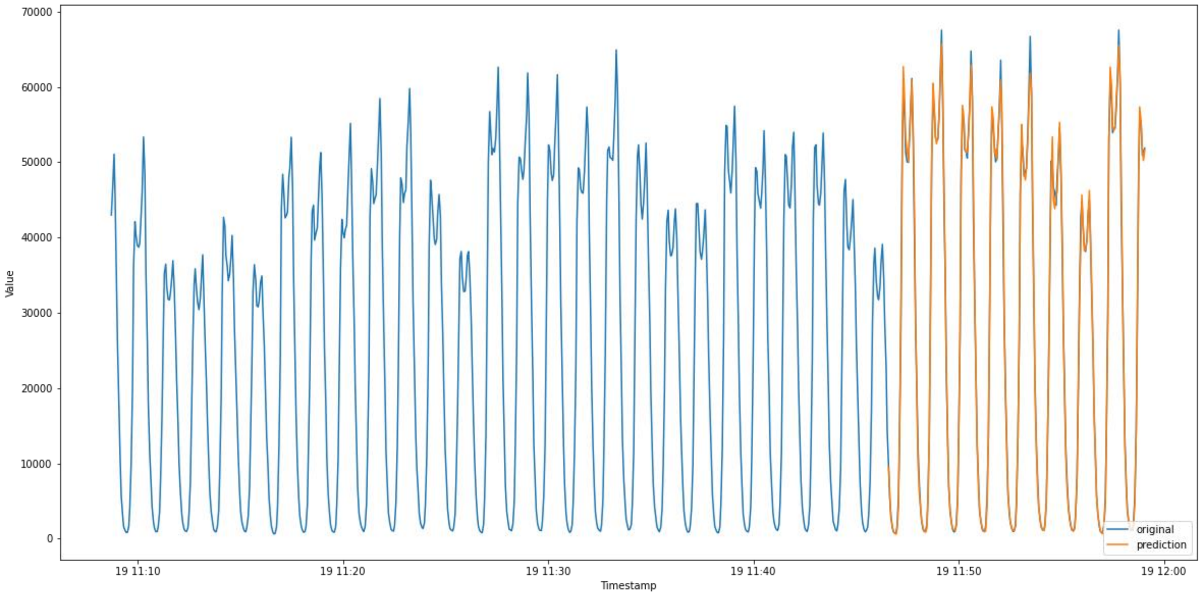
Plot:




```
Score: {'mae': 0.35505214,
 'mse': 0.25230864,
'rmse': 0.5023033,
'mape': 3.5615993}
```

Time Series 31:

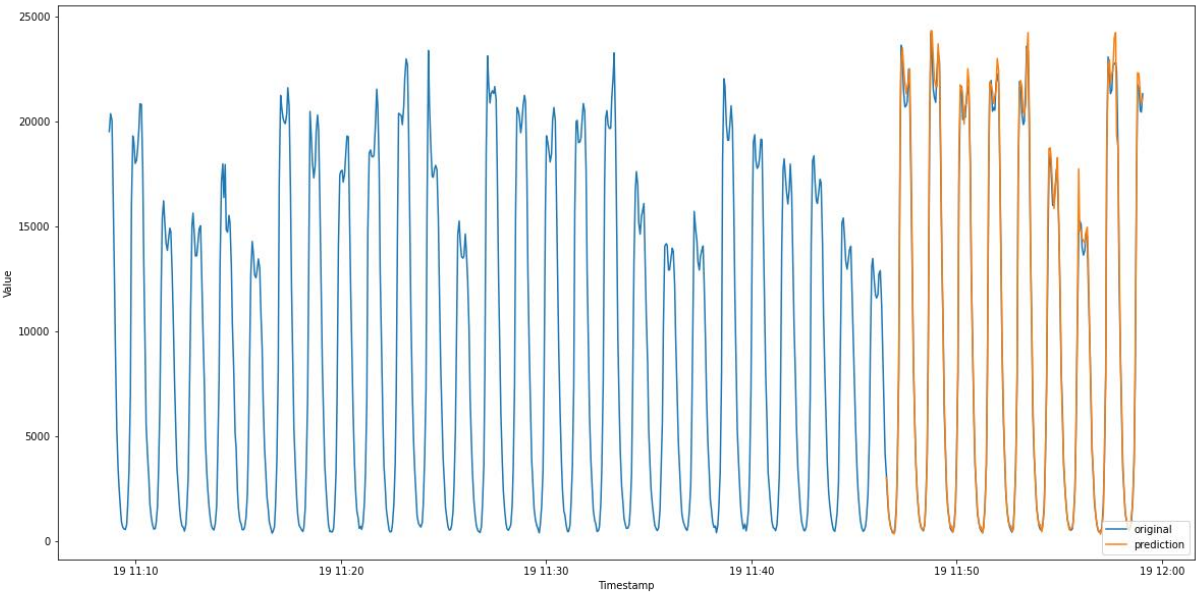
Plot:




```
Score: {'mae': 809.52704,
'mse': 1492936.4,
'rmse': 1221.8578,
'mape': 5.5825644}
```

Time Series 32:

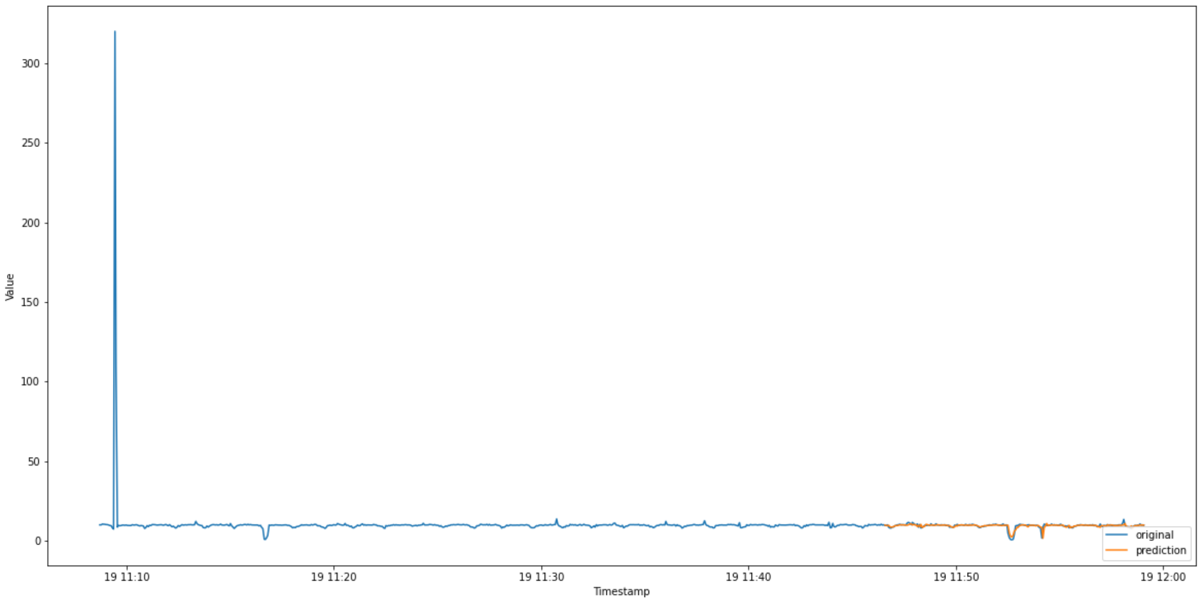
Plot:




```
Score: {'mae': 381.7696,
'mse': 350417.56,
'rmse': 591.96075,
'mape': 6.2006726}
```

Time Series 33:

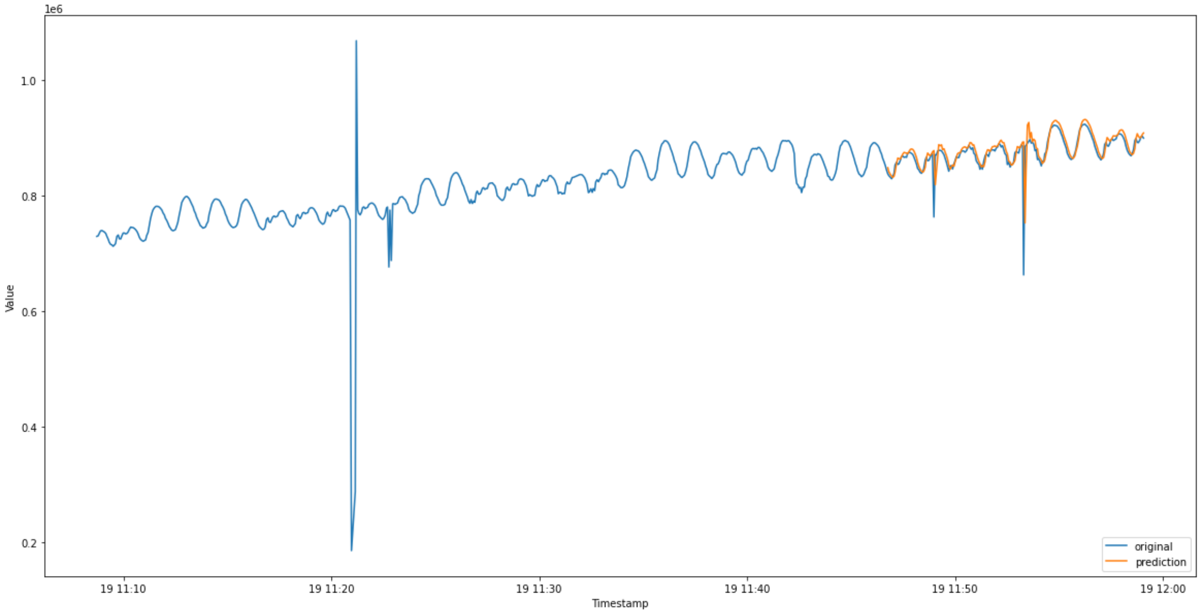
Plot:




```
Score: {'mae': 381.7696,
'mse': 350417.56,
'rmse': 591.96075,
'mape': 6.2006726}
```

Time Series 34:

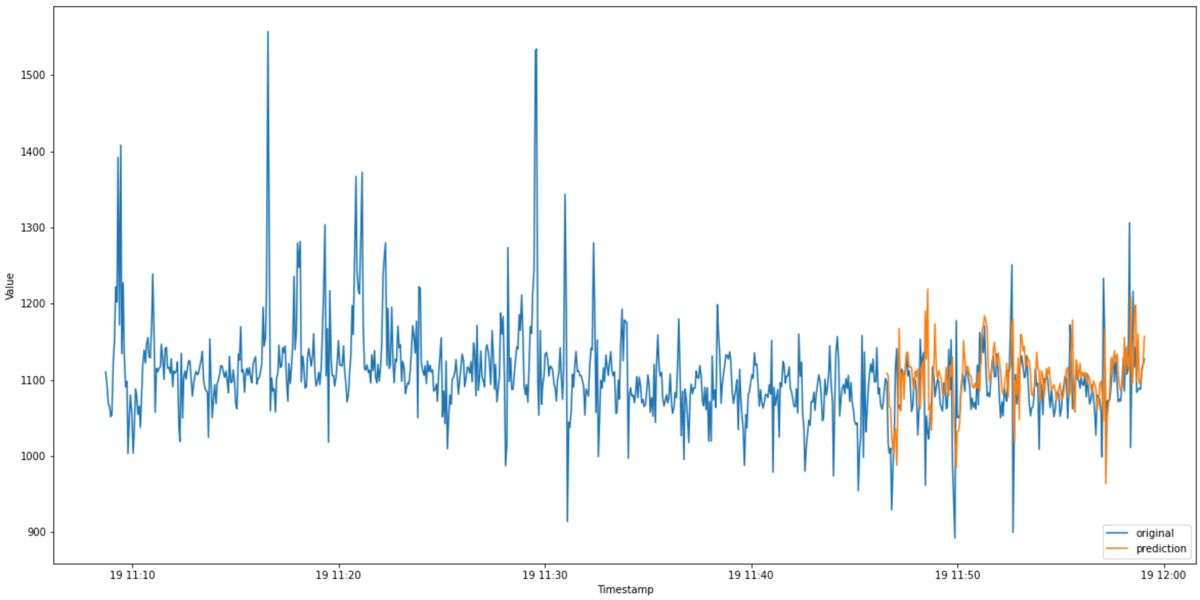
Plot:



Score: {'mae': 9758.505,
'mse': 504421760.0,
'rmse': 22459.336,
'mape': 1.1617001}

Time Series 35:

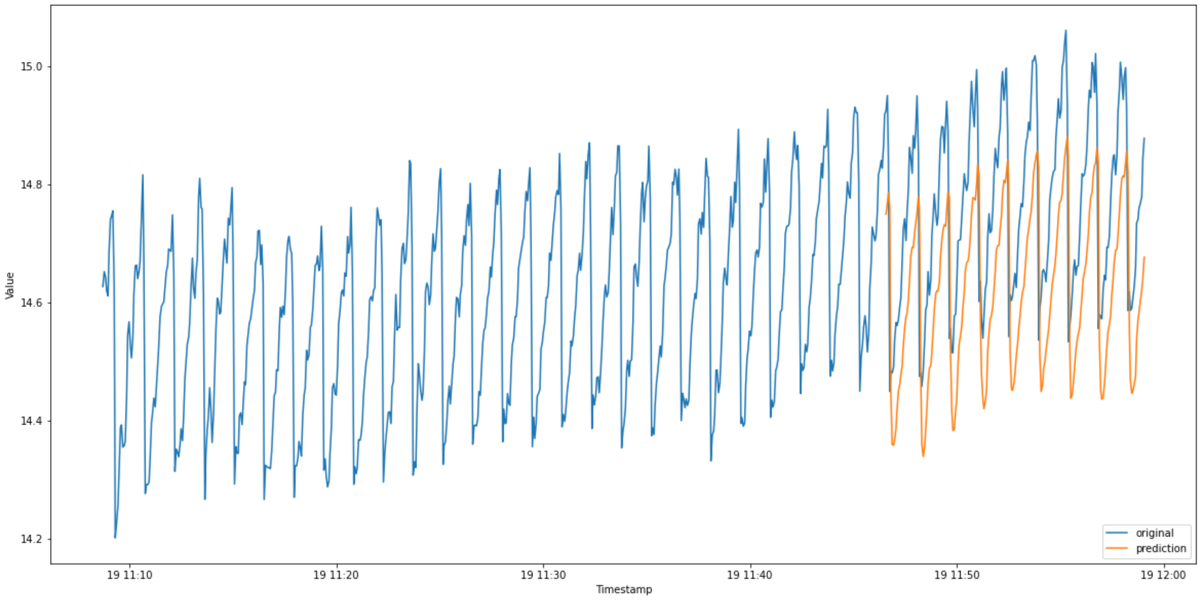
Plot:



Score: {'mae': 37.281876,
'mse': 3301.527,
'rmse': 57.458916,
'mape': 3.4817472}

Time Series 36:

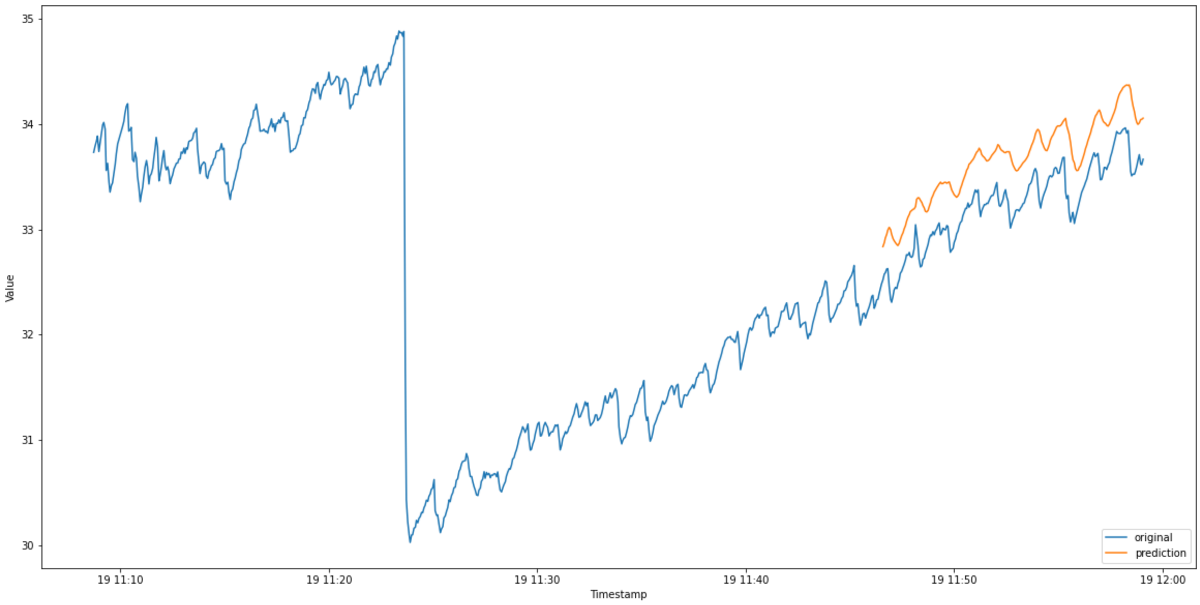
Plot:




```
        Score: {'mae': 0.1638387,
'mse': 0.029130483,
'rmse': 0.17067654,
 'mape': 1.1093642}
```

Time Series 37:

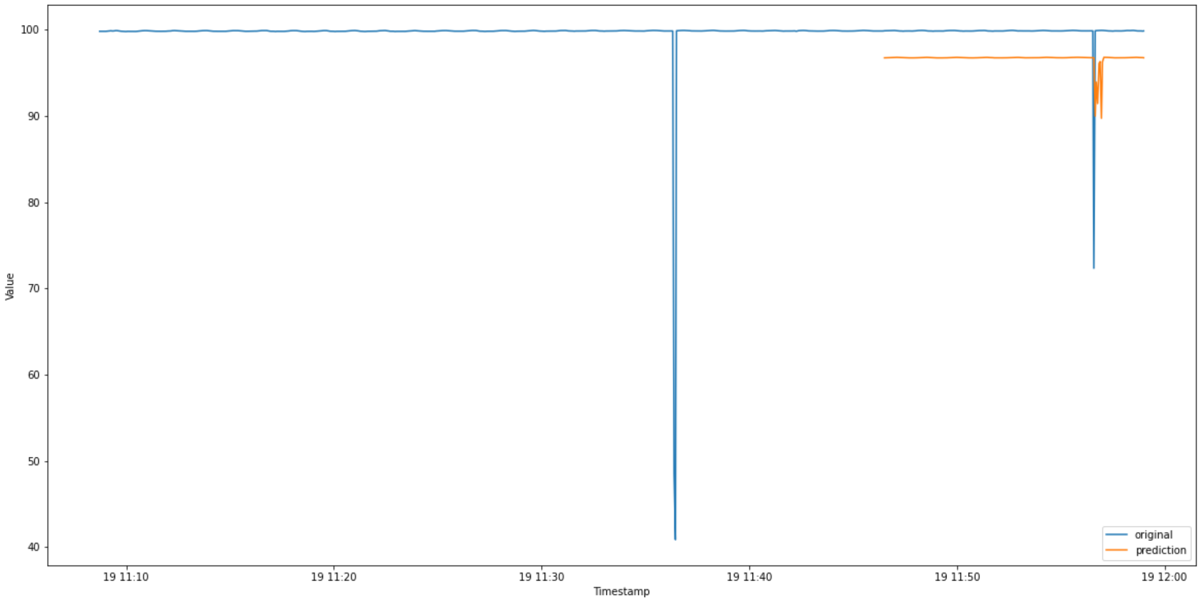
Plot:




```
        Score: {'mae': 0.42024562,
'mse': 0.18895529,
'rmse': 0.43468988,
'mape': 1.2652173}
```

Time Series 38:

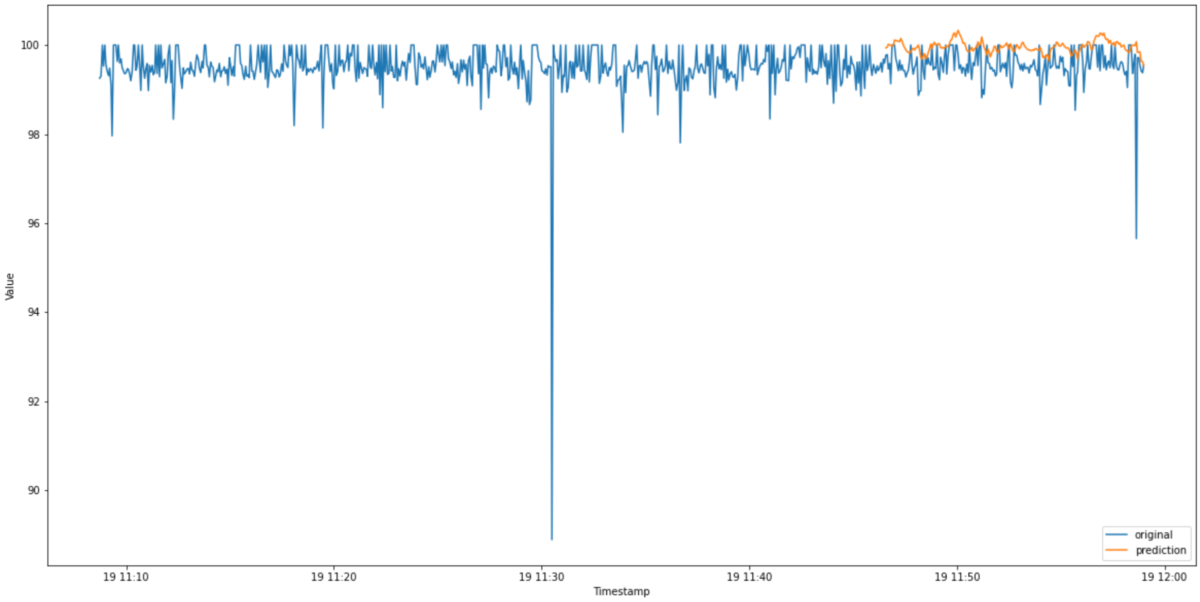
Plot:



Score: {'mae': 3.3323846,
'mse': 13.889019,
'rmse': 3.7267973,
'mape': 3.3804073}

Time Series 39:

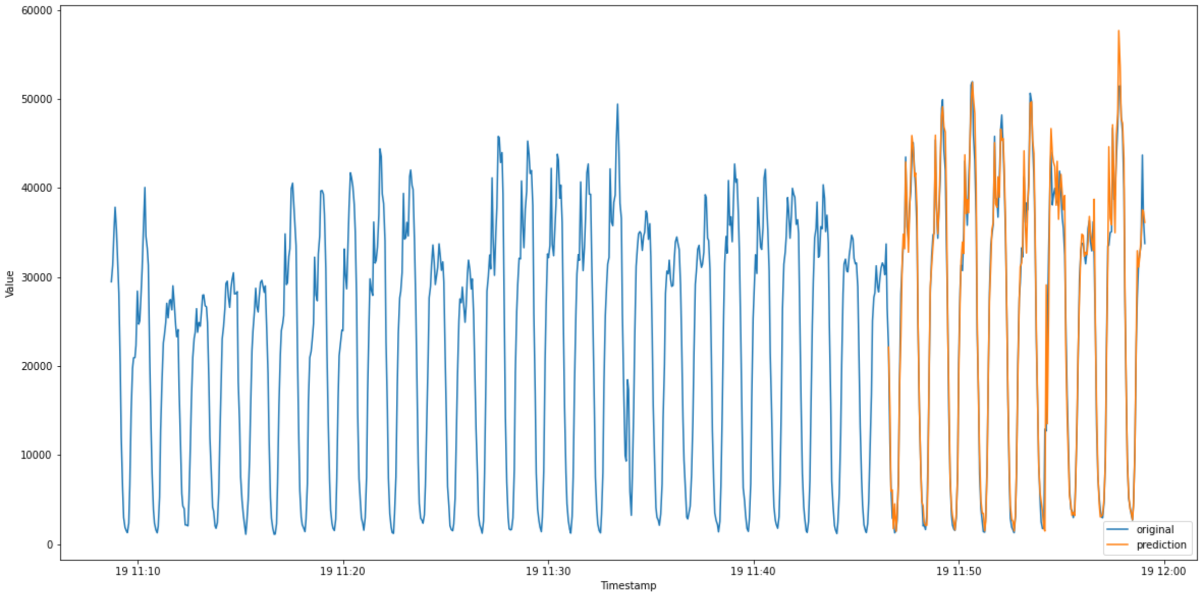
Plot:



Score: {'mae': 0.4351024,
'mse': 0.33712503,
'rmse': 0.5806247,
'mape': 0.43854645}

Time Series 40:

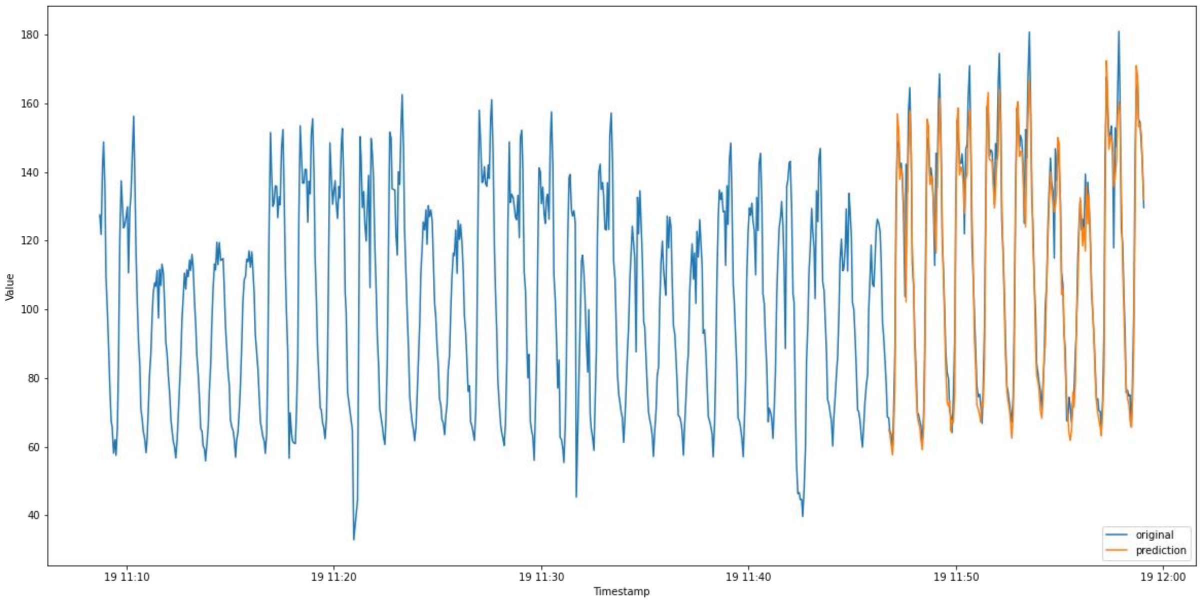
Plot:



Score: {'mae': 1698.7518,
'mse': 7285826.0,
'rmse': 2699.2268,
'mape': 13.346617}

Time Series 41:

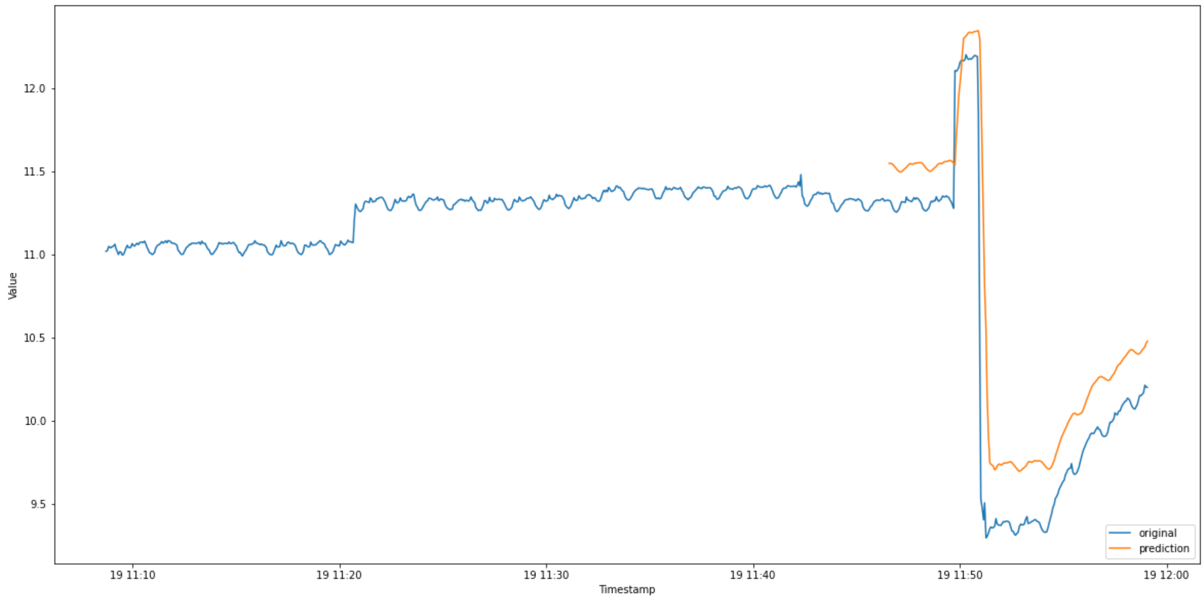
Plot:



Score: {'mae': 5.6619434,
'mse': 61.460682,
'rmse': 7.8396864,
'mape': 5.0056515}

Time Series 42:

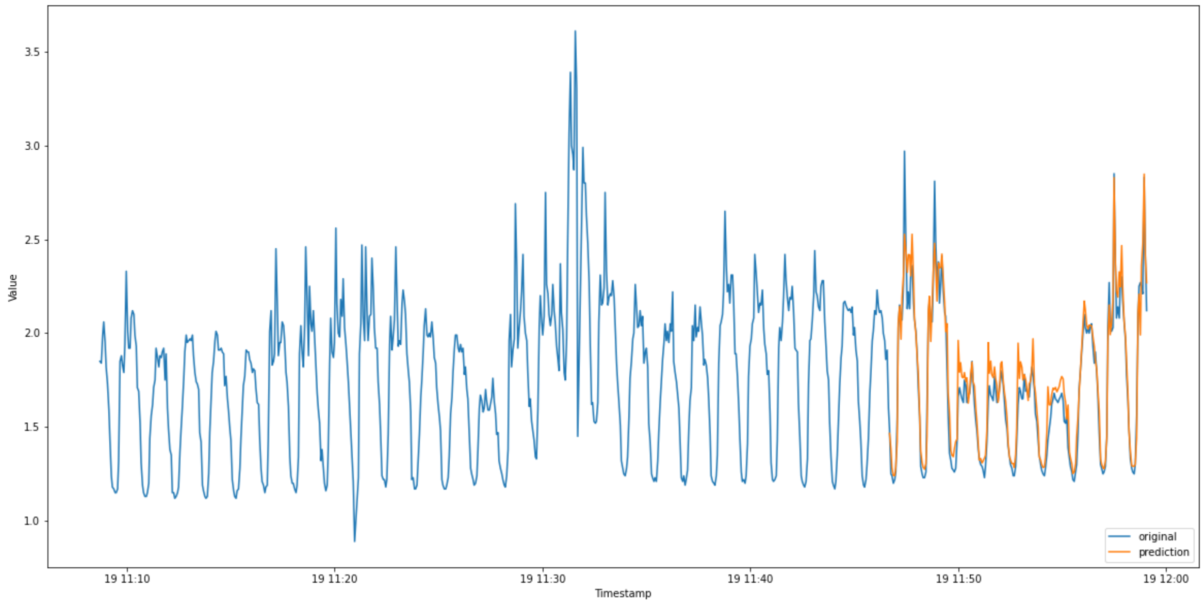
Plot:



Score: {'mae': 0.32905412,
'mse': 0.18557951,
'rmse': 0.4307894,
'mape': 3.2885172}

Time Series 43:

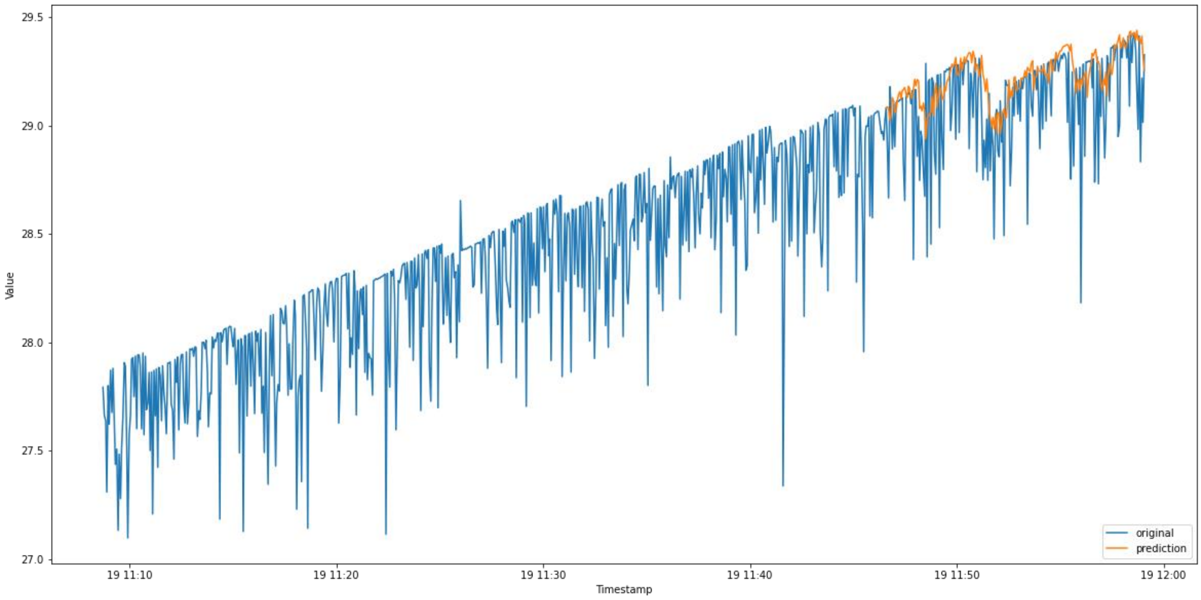
Plot:



Score: {'mae': 0.07652119,
'mse': 0.011263377,
'rmse': 0.10612906,
'mape': 4.476671}

Time Series 44:

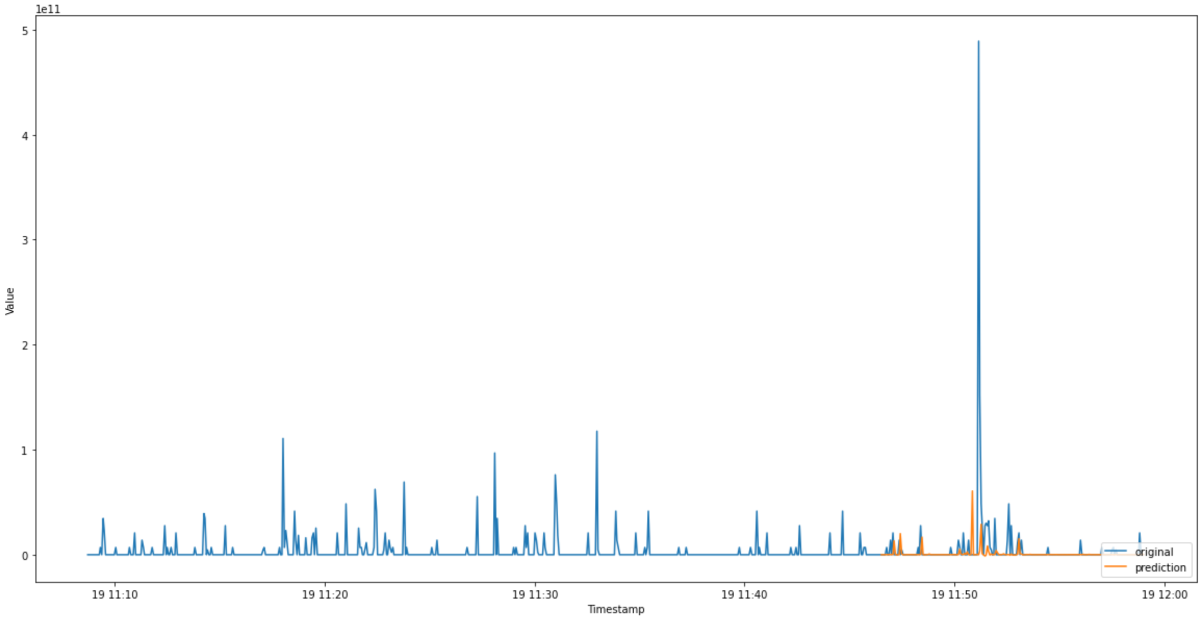
Plot:




```
Score: {'mae': 0.16193204,
'mse': 0.058550995,
'rmse': 0.24197313,
'mape': 0.56061107
```

Time Series 45:

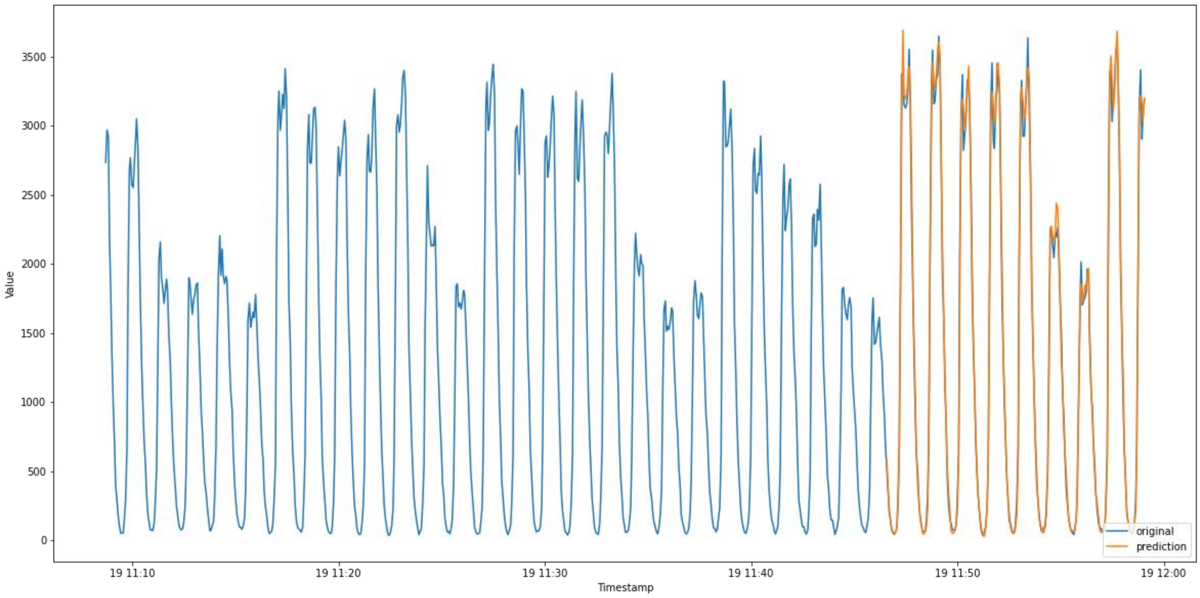
Plot:




```
Score: {'mae': 6272677400.0,
'mse': 1.335853e+21,
'rmse': 36549320000.0,
'mape': 6449819.0}
```

Time Series 46:

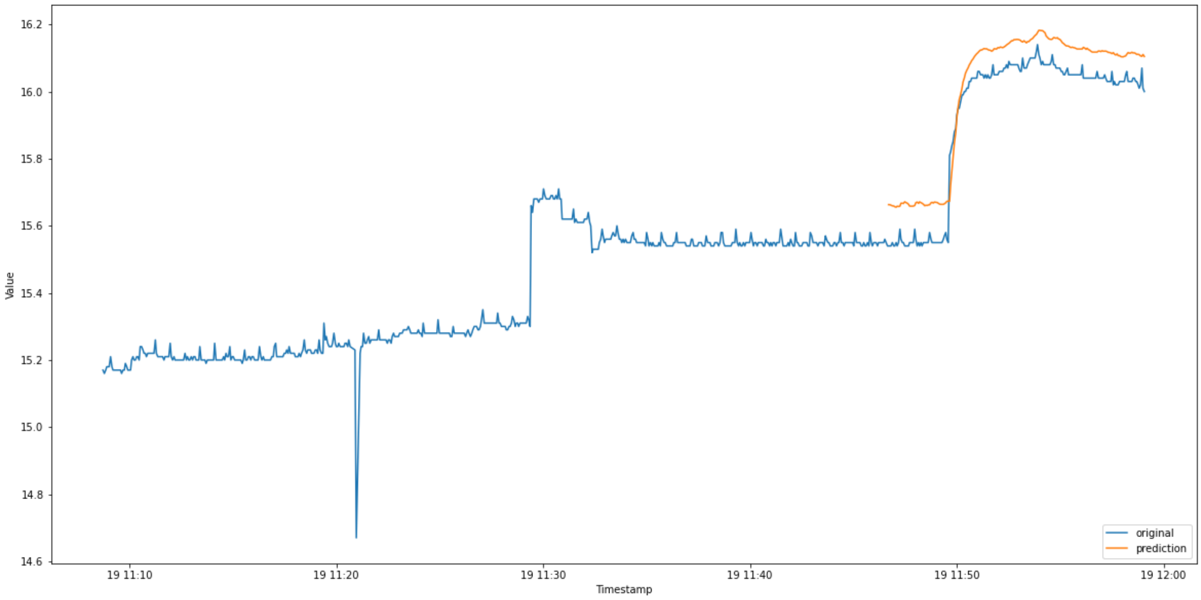
Plot:




```
        Score: {'mae': 72.37821,
'mse': 11199.924,
'rmse': 105.82969,
 'mape': 10.487291}
```

Time Series 47:

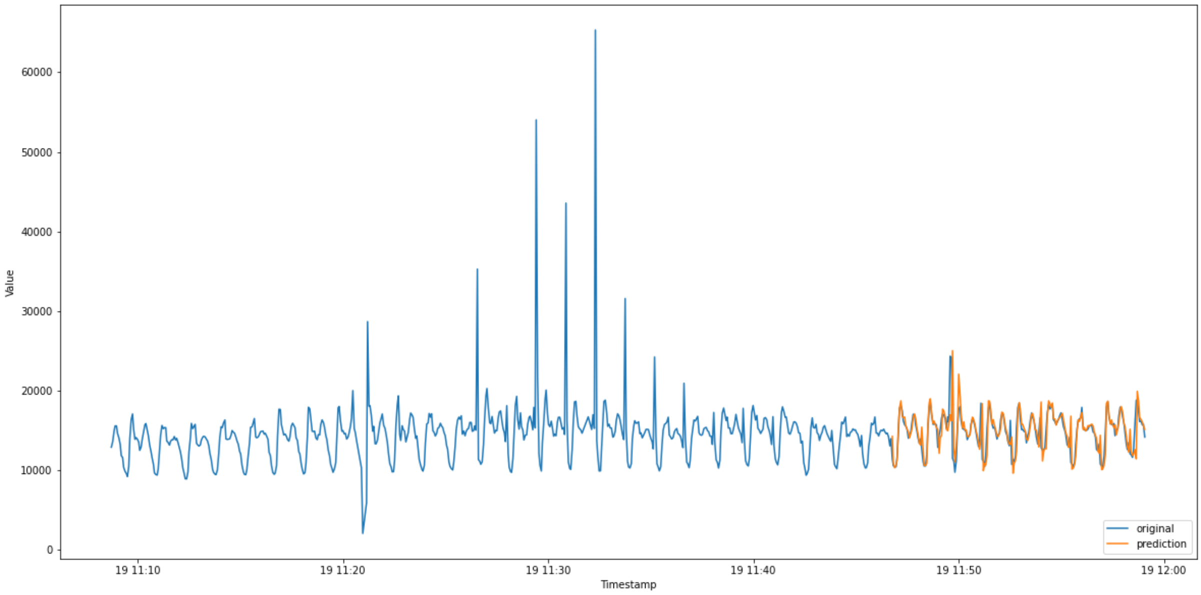
Plot:




```
        Score: {'mae': 0.08240504,
 'mse': 0.0073985024,
 'rmse': 0.08601455,
 'mape': 0.51890814}
```

Time Series 48:

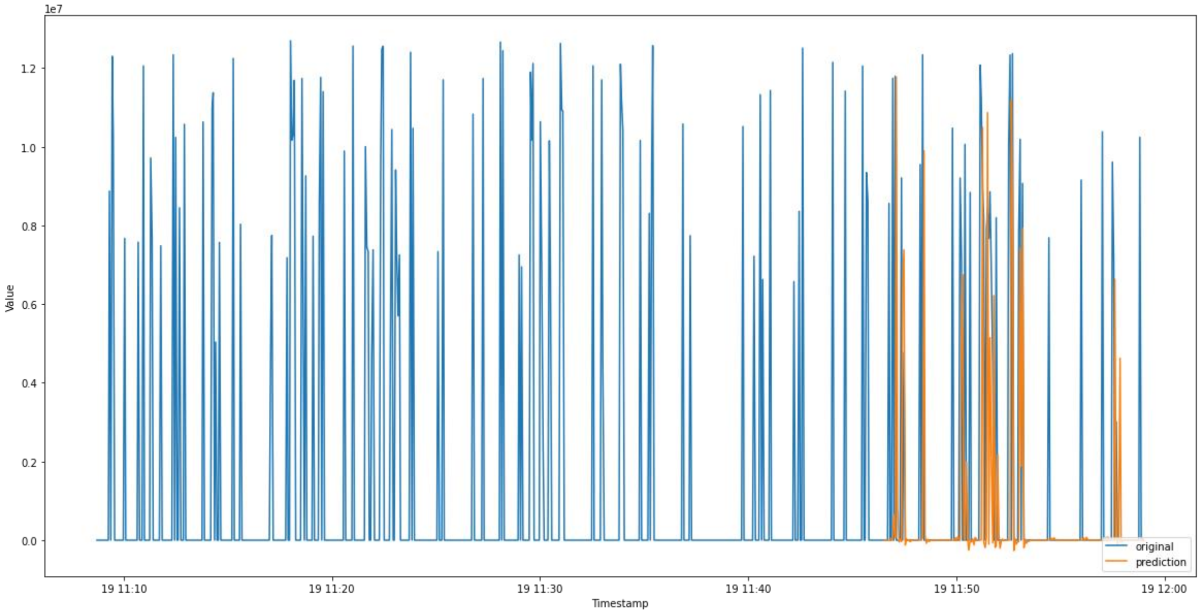
Plot:




```
       Score: {'mae': 944.19293,
'mse': 3436423.2,
'rmse': 1853.7593,
'mape': 6.618242}
```

Time Series 49:

Plot:




```
       Score: {'mae': 1674998.2,
 'mse': 14745200000000.0,
 'rmse': 3839948.0,
 'mape': 1675324.0}
```

REVERSIBLE INSTANCE NORMALIZATION:

As we all know time-series forecasting models suffers from unique characteristics in time-series data i.e. their statistical properties that's changes over the time known as distribution shift. So this might lead in distribution shift in the train and test dataset to overcome this problem we go for reversible instance normalization (RevIN) where first we normalize the input sequences and then denormalize the models output sequences. Instead of performing normalization over the whole data at once we perform normalization over the input instance with instance specific mean and standard deviation and denormalization over the output instance with instance specific mean and standard deviation (the length of input as well as output instances may differ) .This method helps to reduce the noise and error while predicting.

This works because the shift occurred in the input instance is normalized and the output is produced then while denormalization the output instance the reverse processes is used while using the same parameters that are used in input instance so no loss of distribution.

Over the time period this will also work because it mostly depends on the input instance not the drift.

HYPERPARAMETERS:

I have used mostly the paper presented hyperparameters as they were working well for the data only changes were made in batch size and number of epochs. The batch size was reduced to 125 from 1024 and while I was working on cpu number of epochs was reduce to 250 from 5000.

While checking the number of epochs for 100 epochs the error was quite high and for 500 epochs the system took more then 25 min on a single data set but accuracy was quite good as compared to 250 number of epochs.

Parameters used:

HORIZON = 1 # how far to predict forward

WINDOW_SIZE = 7 # how far to lookback

N_EPOCHS = 250 # "Iterations" (5000)

N_NEURONS = 512 # "Width"

N_LAYERS = 4 # "Block-layers"

N_STACKS = 30 #"Stacks"

BATCH_SIZE=125 #"Bath"(1024)

```
INPUT_SIZE = WINDOW_SIZE * HORIZON #  "Lookback period"

THETA_SIZE = INPUT_SIZE + HORIZON
```

NOTE: (I was able to implement NBeats but how to implement RevIN in input layer and output layer was I quite didn’t get)