CNN Model

I chose a CNN model for this weather dataset because CNNs are highly effective at detecting local patterns and relationships within structured data. Weather observations are often organized as sequences over time, and CNNs can efficiently capture short-term dependencies and trends between variables, such as temperature and humidity. Additionally, CNNs process large datasets quickly and are less prone to overfitting compared to more complex models. While RNNs are commonly used for time-series data, CNNs offer faster training times and can handle the multi-dimensional nature of this dataset with ease, making them a strong choice for the task

Scenario 1:

Hyperparameters

Epoch = 30

Batch Size = 16

Hidden Layers = 32

Activation function: Softmax

```
epochs = 30
batch_size = 16
n_hidden = 32

timesteps = len(X_train[0])
input_dim = len(X_train[0][0])
n_classes = len(y_train[0])

model = Sequential()
model.add(Conv1D(n_hidden, kernel_size=2, activation='relu', input_shape=(timesteps, input_dim)))
model.add(Dense(16, activation='relu'))
model.add(MaxPooling1D())
model.add(Flatten())
model.add(Dense(n_classes, activation='softmax')) # Options: sigmoid, tanh, softmax, relu
```

Partial Output

```
Epoch 1/30
1076/1076 - 12s - 11ms/step - accuracy: 0.1334 - loss: 5412.6689
Epoch 2/30
1076/1076 - 8s - 8ms/step - accuracy: 0.1339 - loss: 53711.4922
Epoch 3/30
1076/1076 - 12s - 11ms/step - accuracy: 0.1329 - loss: 184324.0781
Epoch 4/30
1076/1076 - 6s - 6ms/step - accuracy: 0.1311 - loss: 394287.6562
Epoch 5/30
1076/1076 - 7s - 6ms/step - accuracy: 0.1303 - loss: 699204.8750
Epoch 6/30
1076/1076 - 6s - 6ms/step - accuracy: 0.1340 - loss: 1131961.5000
Epoch 7/30
1076/1076 - 8s - 8ms/step - accuracy: 0.1318 - loss: 1659987.1250
Epoch 8/30
1076/1076 - 7s - 6ms/step - accuracy: 0.1300 - loss: 2337215.5000
Epoch 9/30
1076/1076 - 7s - 6ms/step - accuracy: 0.1287 - loss: 3101672.5000
Epoch 10/30
1076/1076 - 7s - 7ms/step - accuracy: 0.1301 - loss: 4050989.7500
Epoch 11/30
1076/1076 - 6s - 5ms/step - accuracy: 0.1322 - loss: 5097046.0000
Epoch 12/30
1076/1076 - 7s - 6ms/step - accuracy: 0.1305 - loss: 6289769.0000
Epoch 13/30
1076/1076 - 6s - 6ms/step - accuracy: 0.1347 - loss: 7685916.5000
Epoch 14/30
1076/1076 - 7s - 6ms/step - accuracy: 0.1324 - loss: 9358731.0000
Epoch 15/30
1076/1076 - 7s - 6ms/step - accuracy: 0.1300 - loss: 11218251.0000
Epoch 16/30
1076/1076 - 6s - 6ms/step - accuracy: 0.1330 - loss: 13169979.0000
Epoch 17/30
1076/1076 - 9s - 8ms/step - accuracy: 0.1312 - loss: 15345613.0000
Epoch 18/30
1076/1076 - 6s - 6ms/step - accuracy: 0.1323 - loss: 17739460.0000
```

Confusion Matrix

```
180/180 ━━━━━━━━━━━━━━━━━━━━ 2s 8ms/step
```

Pred / True	DEBILT	DUSSELDORF	HEATHROW	KASSEL	LJUBLJANA	MAASTRICHT
BASEL	255	11	3	59	1769	307
BELGRADE	15	0	0	0	698	29
BUDAPEST	3	0	0	0	94	8
DEBILT	3	0	0	0	33	3
DUSSELDORF	1	0	0	0	6	5
HEATHROW	5	0	0	0	8	4
KASSEL	1	0	0	0	6	1
LJUBLJANA	1	0	0	0	23	2
MAASTRICHT	0	0	0	0	5	0
MADRID	30	0	0	1	72	23
MUNCHENB	0	0	0	0	6	0
OSLO	0	0	0	0	1	0
STOCKHOLM	0	0	0	0	2	0
VALENTIA	1	0	0	0	0	0

Pred / True	MADRID	OSLO	SONNBLICK	STOCKHOLM	VALENTIA
BASEL	1003	49	7	183	36
BELGRADE	283	4	0	63	0
BUDAPEST	91	0	0	18	0
DEBILT	28	0	0	15	0
DUSSELDORF	11	0	0	6	0
HEATHROW	53	1	0	11	0
KASSEL	2	1	0	0	0
LJUBLJANA	33	0	0	2	0
MAASTRICHT	4	0	0	0	0
MADRID	292	4	0	36	0
MUNCHENB	1	1	0	0	0
OSLO	1	0	0	3	0
STOCKHOLM	2	0	0	0	0
VALENTIA	0	0	0	0	0

Scenario 2:

Hyperparameters

Epochs = 30

Batch size = 16

Hidden Layers = 4

Activation function: softmax

```python
epochs = 30
batch_size = 16
n_hidden = 4

timesteps = len(X_train[0])
input_dim = len(X_train[0][0])
n_classes = len(y_train[0])

model = Sequential()
model.add(Conv1D(n_hidden, kernel_size=2, activation='relu', input_shape=(timesteps, input_dim)))
model.add(Dense(16, activation='relu'))
model.add(MaxPooling1D())
model.add(Flatten())
model.add(Dense(n_classes, activation='softmax')) # Options: sigmoid, tanh, softmax, relu
```

Partial Output

```
Epoch 1/30
1076/1076 - 12s - 11ms/step - accuracy: 0.1199 - loss: 310.2769
Epoch 2/30
1076/1076 - 6s - 6ms/step - accuracy: 0.1312 - loss: 2824.5261
Epoch 3/30
1076/1076 - 8s - 7ms/step - accuracy: 0.1361 - loss: 9457.5664
Epoch 4/30
1076/1076 - 7s - 6ms/step - accuracy: 0.1382 - loss: 20126.2129
Epoch 5/30
1076/1076 - 8s - 7ms/step - accuracy: 0.1405 - loss: 37439.6953
Epoch 6/30
1076/1076 - 6s - 6ms/step - accuracy: 0.1354 - loss: 58767.5352
Epoch 7/30
1076/1076 - 6s - 6ms/step - accuracy: 0.1329 - loss: 85150.3203
Epoch 8/30
1076/1076 - 6s - 5ms/step - accuracy: 0.1390 - loss: 121737.5781
Epoch 9/30
1076/1076 - 6s - 5ms/step - accuracy: 0.1383 - loss: 164103.5625
Epoch 10/30
1076/1076 - 6s - 6ms/step - accuracy: 0.1393 - loss: 212404.8750
Epoch 11/30
1076/1076 - 6s - 6ms/step - accuracy: 0.1329 - loss: 271060.2500
Epoch 12/30
1076/1076 - 6s - 5ms/step - accuracy: 0.1368 - loss: 339110.1562
Epoch 13/30
1076/1076 - 6s - 6ms/step - accuracy: 0.1332 - loss: 412769.4375
Epoch 14/30
1076/1076 - 6s - 6ms/step - accuracy: 0.1328 - loss: 499810.2188
Epoch 15/30
1076/1076 - 6s - 6ms/step - accuracy: 0.1363 - loss: 597532.1875
Epoch 16/30
1076/1076 - 7s - 6ms/step - accuracy: 0.1346 - loss: 706225.9375
Epoch 17/30
1076/1076 - 6s - 6ms/step - accuracy: 0.1323 - loss: 822432.1250
Epoch 18/30
1076/1076 - 6s - 5ms/step - accuracy: 0.1292 - loss: 955152.5000
```

Confusion Matrix

```
Pred        LJUBLJANA  MAASTRICHT  MADRID  MUNCHENB  OSLO  SONNBLICK  \
True
BASEL              15          50    1794       168   215         22
BELGRADE            5           0     767        35    59          0
BUDAPEST            2           0     149         4    25          0
DEBILT              3           0      46         2    16          0
DUSSELDORF          0           0      16         0     6          0
HEATHROW            0           0      46         0    12          0
KASSEL              1           0       4         1     1          0
LJUBLJANA           0           0      45         1     5          0
MAASTRICHT          0           0       5         0     0          0
MADRID              0           0     303         4    41          0
MUNCHENB            0           0       5         1     1          0
OSLO                0           0       1         0     4          0
STOCKHOLM           0           0       2         0     1          0
VALENTIA            0           0       0         0     0          0

Pred        STOCKHOLM  VALENTIA
True
BASEL              54        32
BELGRADE           11         0
BUDAPEST            1         0
DEBILT              0         0
DUSSELDORF          0         0
HEATHROW            1         0
KASSEL              0         0
LJUBLJANA           0         0
MAASTRICHT          0         0
MADRID              7         0
MUNCHENB            0         0
OSLO                0         0
STOCKHOLM           0         0
VALENTIA            0         0
```

Scenario 3

Hyperparameters

Epochs = 30

Batch size = 16

Hidden Layers = 128

Activation function: tanh

```
epochs = 30
batch_size = 16
n_hidden = 128

timesteps = len(X_train[0])
input_dim = len(X_train[0][0])
n_classes = len(y_train[0])

model = Sequential()
model.add(Conv1D(n_hidden, kernel_size=2, activation='relu', input_shape=(timesteps, input_dim)))
model.add(Dense(16, activation='relu'))
model.add(MaxPooling1D())
model.add(Flatten())
model.add(Dense(n_classes, activation='tanh')) # Options: sigmoid, tanh, softmax, relu
```

Partial Output

```
Epoch 1/30
1076/1076 - 12s - 11ms/step - accuracy: 0.0613 - loss: 24.0554
Epoch 2/30
1076/1076 - 10s - 9ms/step - accuracy: 0.0049 - loss: 24.4627
Epoch 3/30
1076/1076 - 7s - 7ms/step - accuracy: 0.0016 - loss: 24.4309
Epoch 4/30
1076/1076 - 7s - 7ms/step - accuracy: 0.0016 - loss: 24.4309
Epoch 5/30
1076/1076 - 7s - 6ms/step - accuracy: 0.0016 - loss: 24.4309
Epoch 6/30
1076/1076 - 7s - 6ms/step - accuracy: 0.0016 - loss: 24.4309
Epoch 7/30
1076/1076 - 7s - 6ms/step - accuracy: 0.0016 - loss: 24.4309
Epoch 8/30
1076/1076 - 7s - 7ms/step - accuracy: 0.0016 - loss: 24.4309
Epoch 9/30
1076/1076 - 7s - 6ms/step - accuracy: 0.0016 - loss: 24.4309
Epoch 10/30
1076/1076 - 7s - 6ms/step - accuracy: 0.0016 - loss: 24.4309
Epoch 11/30
1076/1076 - 7s - 7ms/step - accuracy: 0.0016 - loss: 24.4309
Epoch 12/30
1076/1076 - 7s - 6ms/step - accuracy: 0.0016 - loss: 24.4309
Epoch 13/30
1076/1076 - 9s - 8ms/step - accuracy: 0.0016 - loss: 24.4309
Epoch 14/30
1076/1076 - 6s - 6ms/step - accuracy: 0.0016 - loss: 24.4309
Epoch 15/30
1076/1076 - 7s - 7ms/step - accuracy: 0.0016 - loss: 24.4309
Epoch 16/30
1076/1076 - 7s - 6ms/step - accuracy: 0.0016 - loss: 24.4309
```

Confusion Matrix

```
180/180 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step
Pred        BELGRADE  DUSSELDORF  HEATHROW  LJUBLJANA  MAASTRICHT  MADRID  \
True
BASEL              4           1       111         80         860     109
BELGRADE           9           0        31         68         636       0
BUDAPEST           1           0         4          4          95       0
DEBILT             0           0         0          2          47       0
DUSSELDORF         0           0         0          1          11       0
HEATHROW           0           0         0          0          20       0
KASSEL             0           0         0          0           6       0
LJUBLJANA          0           0         0          2           5       0
MAASTRICHT         0           0         0          0           1       0
MADRID             2           0         2          1          58       0
MUNCHENB           0           0         0          0           4       0
OSLO               0           0         0          1           2       0
STOCKHOLM          0           0         0          0           3       0
VALENTIA           0           0         0          0           0       0

Pred        SONNBLICK  STOCKHOLM  VALENTIA
True
BASEL             611       1476       430
BELGRADE            1        326        21
BUDAPEST            0        105         5
DEBILT              1         30         2
DUSSELDORF          1         15         1
HEATHROW            2         60         0
KASSEL              0          5         0
LJUBLJANA           0         51         3
MAASTRICHT          0          8         0
MADRID             65        308        22
MUNCHENB            0          4         0
OSLO                0          1         1
STOCKHOLM           0          1         0
```

Scenario 4

Hyperparameters

Epochs = 30

Batch size = 16

Hidden Layers = 64

Activation function: tanh

```
epochs = 30
batch_size = 16
n_hidden = 64

timesteps = len(X_train[0])
input_dim = len(X_train[0][0])
n_classes = len(y_train[0])

model = Sequential()
model.add(Conv1D(n_hidden, kernel_size=2, activation='relu', input_shape=(timesteps, input_dim)))
model.add(Dense(16, activation='relu'))
model.add(MaxPooling1D())
model.add(Flatten())
model.add(Dense(n_classes, activation='tanh')) # Options: sigmoid, tanh, softmax, relu
```

Partial Output

```
Epoch 1/30
1076/1076 - 14s - 13ms/step - accuracy: 0.0282 - loss: 21.1009
Epoch 2/30
1076/1076 - 9s - 8ms/step - accuracy: 0.1758 - loss: 24.9006
Epoch 3/30
1076/1076 - 8s - 8ms/step - accuracy: 0.1946 - loss: 24.4666
Epoch 4/30
1076/1076 - 10s - 9ms/step - accuracy: 0.1946 - loss: 25.6212
Epoch 5/30
1076/1076 - 11s - 10ms/step - accuracy: 0.1946 - loss: 25.5893
Epoch 6/30
1076/1076 - 11s - 10ms/step - accuracy: 0.1946 - loss: 25.5500
Epoch 7/30
1076/1076 - 12s - 11ms/step - accuracy: 0.1946 - loss: 25.1220
Epoch 8/30
1076/1076 - 9s - 9ms/step - accuracy: 0.1946 - loss: 23.5778
Epoch 9/30
1076/1076 - 11s - 10ms/step - accuracy: 0.1946 - loss: 23.5741
Epoch 10/30
1076/1076 - 11s - 10ms/step - accuracy: 0.1946 - loss: 23.5740
Epoch 11/30
1076/1076 - 11s - 10ms/step - accuracy: 0.1946 - loss: 23.5741
Epoch 12/30
1076/1076 - 10s - 10ms/step - accuracy: 0.1946 - loss: 23.5741
Epoch 13/30
1076/1076 - 12s - 11ms/step - accuracy: 0.1946 - loss: 23.5722
Epoch 14/30
1076/1076 - 11s - 10ms/step - accuracy: 0.1946 - loss: 23.5750
Epoch 15/30
1076/1076 - 9s - 9ms/step - accuracy: 0.1946 - loss: 23.5740
Epoch 16/30
1076/1076 - 12s - 11ms/step - accuracy: 0.1946 - loss: 23.5834
Epoch 17/30
```

Confusion Matrix

```
180/180 ━━━━━━━━━━━━━━━━━━━━ 2s 8ms/step
Pred         BASEL  BELGRADE  BUDAPEST  DEBILT  MADRID  MUNCHENB  VALENTIA
True
BASEL         1446      2165        41       6       6        14         4
BELGRADE       886       206         0       0       0         0         0
BUDAPEST       191        23         0       0       0         0         0
DEBILT          80         2         0       0       0         0         0
DUSSELDORF      26         3         0       0       0         0         0
HEATHROW        66        16         0       0       0         0         0
KASSEL          10         1         0       0       0         0         0
LJUBLJANA       33        28         0       0       0         0         0
MAASTRICHT       5         4         0       0       0         0         0
MADRID         162       296         0       0       0         0         0
MUNCHENB         4         4         0       0       0         0         0
OSLO             5         0         0       0       0         0         0
STOCKHOLM        4         0         0       0       0         0         0
VALENTIA         1         0         0       0       0         0         0
```

Scenario 5

Hyperparameters

Epochs = 30

Batch size = 16

Hidden Layers = 64

Activation function: sigmoid

```python
epochs = 30
batch_size = 16
n_hidden = 64

timesteps = len(X_train[0])
input_dim = len(X_train[0][0])
n_classes = len(y_train[0])

model = Sequential()
model.add(Conv1D(n_hidden, kernel_size=2, activation='relu', input_shape=(timesteps, input_dim)))
model.add(Dense(16, activation='relu'))
model.add(MaxPooling1D())
model.add(Flatten())
model.add(Dense(n_classes, activation='sigmoid')) # Options: sigmoid, tanh, softmax, relu
```

Partial Output

```
Epoch 1/30
1076/1076 - 14s - 13ms/step - accuracy: 0.6181 - loss: 7473.1250
Epoch 2/30
1076/1076 - 8s - 7ms/step - accuracy: 0.6436 - loss: 79615.6562
Epoch 3/30
1076/1076 - 9s - 9ms/step - accuracy: 0.6438 - loss: 276341.9688
Epoch 4/30
1076/1076 - 9s - 8ms/step - accuracy: 0.6439 - loss: 600801.6250
Epoch 5/30
1076/1076 - 11s - 10ms/step - accuracy: 0.6439 - loss: 1080887.8750
Epoch 6/30
1076/1076 - 11s - 10ms/step - accuracy: 0.6439 - loss: 1708916.5000
Epoch 7/30
1076/1076 - 10s - 9ms/step - accuracy: 0.6440 - loss: 2512762.2500
Epoch 8/30
1076/1076 - 10s - 9ms/step - accuracy: 0.6440 - loss: 3492421.0000
Epoch 9/30
1076/1076 - 10s - 9ms/step - accuracy: 0.6440 - loss: 4698262.5000
Epoch 10/30
1076/1076 - 9s - 9ms/step - accuracy: 0.6440 - loss: 6076169.0000
Epoch 11/30
1076/1076 - 10s - 9ms/step - accuracy: 0.6440 - loss: 7757760.0000
Epoch 12/30
1076/1076 - 12s - 11ms/step - accuracy: 0.6439 - loss: 9654584.0000
Epoch 13/30
1076/1076 - 8s - 8ms/step - accuracy: 0.6440 - loss: 11808758.0000
Epoch 14/30
1076/1076 - 11s - 10ms/step - accuracy: 0.6440 - loss: 14281733.0000
Epoch 15/30
1076/1076 - 11s - 10ms/step - accuracy: 0.6440 - loss: 16997162.0000
Epoch 16/30
1076/1076 - 9s - 9ms/step - accuracy: 0.6440 - loss: 20127426.0000
Epoch 17/30
```

Confusion Matrix

```
180/180 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step
Pred          BASEL
True
BASEL          3682
BELGRADE       1092
BUDAPEST        214
DEBILT           82
DUSSELDORF       29
HEATHROW         82
KASSEL           11
LJUBLJANA        61
MAASTRICHT        9
MADRID          458
MUNCHENB          8
OSLO              5
STOCKHOLM         4
VALENTIA          1
```

Scenario 6

Hyperparameters

Epochs = 15

Batch size = 4

Hidden Layers = 4

Activation function: relu

```
epochs = 15
batch_size = 4
n_hidden = 4

timesteps = len(X_train[0])
input_dim = len(X_train[0][0])
n_classes = len(y_train[0])

model = Sequential()
model.add(Conv1D(n_hidden, kernel_size=2, activation='relu', input_shape=(timesteps, input_dim)))
model.add(Dense(16, activation='relu'))
model.add(MaxPooling1D())
model.add(Flatten())
model.add(Dense(n_classes, activation='relu')) # Options: sigmoid, tanh, softmax, relu
```

Partial Output

```
Epoch 1/15
4303/4303 - 34s - 8ms/step - accuracy: 0.4771 - loss: nan
Epoch 2/15
4303/4303 - 29s - 7ms/step - accuracy: 0.6440 - loss: nan
Epoch 3/15
4303/4303 - 29s - 7ms/step - accuracy: 0.6440 - loss: nan
Epoch 4/15
4303/4303 - 27s - 6ms/step - accuracy: 0.6440 - loss: nan
Epoch 5/15
4303/4303 - 27s - 6ms/step - accuracy: 0.6440 - loss: nan
Epoch 6/15
4303/4303 - 26s - 6ms/step - accuracy: 0.6440 - loss: nan
Epoch 7/15
4303/4303 - 31s - 7ms/step - accuracy: 0.6440 - loss: nan
Epoch 8/15
4303/4303 - 32s - 7ms/step - accuracy: 0.6440 - loss: nan
Epoch 9/15
4303/4303 - 35s - 8ms/step - accuracy: 0.6440 - loss: nan
Epoch 10/15
4303/4303 - 39s - 9ms/step - accuracy: 0.6440 - loss: nan
Epoch 11/15
4303/4303 - 33s - 8ms/step - accuracy: 0.6440 - loss: nan
Epoch 12/15
4303/4303 - 31s - 7ms/step - accuracy: 0.6440 - loss: nan
Epoch 13/15
4303/4303 - 32s - 7ms/step - accuracy: 0.6440 - loss: nan
Epoch 14/15
4303/4303 - 35s - 8ms/step - accuracy: 0.6440 - loss: nan
Epoch 15/15
4303/4303 - 33s - 8ms/step - accuracy: 0.6440 - loss: nan
```

Confusion Matrix

```
180/180 ━━━━━━━━━━━━━━━━━━━━ 2s 8ms/step
Pred        BASEL
True
BASEL        3682
BELGRADE     1092
BUDAPEST      214
DEBILT         82
DUSSELDORF     29
HEATHROW       82
KASSEL         11
LJUBLJANA      61
MAASTRICHT      9
MADRID        458
MUNCHENB        8
OSLO            5
STOCKHOLM       4
VALENTIA        1
```

Final Comments

In this project, I explored six different CNN model configurations by varying key hyperparameters such as the number of hidden units, batch size, epochs, and activation functions (softmax, tanh, relu, and sigmoid). Each scenario gave different insights into how these parameters affect model performance for multi-class weather station classification.

Parameter effects and interpretation:

- **Hidden Layer Size (n_hidden):**
  Increasing the number of hidden units (e.g., 64 or 128) allowed the network to learn more complex patterns, but also sometimes led to unstable results or overfitting. Smaller networks (n_hidden = 4 or 16) tended to converge faster but did not always capture the full complexity of the data.

- **Activation Function:**
  The choice of activation function in the output layer had a strong effect. Softmax and sigmoid usually worked better for multi-class classification, enabling the model to predict probabilities for each class. In some cases, relu or tanh resulted in non-converging models or poor class separation.

- **Batch Size and Epochs:**
  Larger batch sizes (16) and moderate epochs (15–30) often provided a balance between training speed and performance, but some models plateaued quickly or even produced NaN loss values, indicating instability or exploding gradients.

- **Accuracy and Confusion Matrices:**
  In the scenarios with higher accuracy (e.g., around 64 percent), the confusion matrix showed that the model correctly classified a substantial number of samples, though some stations dominated the predictions and others were rarely recognized. In some configurations, the model's predictions collapsed to one or a few classes, especially when loss became NaN.

Model interpretation:

Across scenarios, the best results were obtained with a moderately high number of hidden units and by using the sigmoid activation in the final layer. However, even the best models struggled to recognize all 15 stations equally, with a tendency to favor the most represented classes in the dataset. In several runs, a NaN loss or nearly constant accuracy suggested the need for better regularization, learning rate tuning, or possibly further data preprocessing.