

```
In [1]: # import all Libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.metrics import Accuracy, Precision, Recall, AUC
```

```
In [2]: # read in dataset
data = pd.read_csv('Occupancy.csv')
# get hour number from datetime column
data['Datetime'] = pd.to_datetime(data['date'])
data['Hour'] = data['Datetime'].dt.hour
# getting the weekday from the 'date' column
data['DOTW'] = pd.to_datetime(data['date']).dt.day_name()
# making the column 'DOTW' to be categorical and will now be changed to numerical values representing the days of the week
day_mapping = {
    'Monday': 1,
    'Tuesday': 2,
    'Wednesday': 3,
    'Thursday': 4,
    'Friday': 5,
    'Saturday': 6,
    'Sunday': 7
}
# now mapping the new values to the 'DOTW' column
data['DOTW_encoded'] = data['DOTW'].map(day_mapping)
# create a binary variable for weekday or not
data['Weekday'] = np.where(data['DOTW_encoded'] < 6, 1, 0)
# create a binary variable for working hours or not
data['Working_Hours'] = np.where(np.logical_and(data['Hour'] >= 6, data['Hour'] <= 18), 1, 0)
# create a binary variable for working hours on a weekday
data['Workdayhrs'] = np.where(np.logical_and(data['Weekday'] == 1, data['Working_Hours'] == 1), 1, 0)
# reduce Light variable to binary
data['Light_on_off'] = np.where(data['Light'] > 0, 1, 0)
# keep only desired columns
data1 = data[["Occupancy", "Light_on_off", "Weekday", "Working_Hours", "Workdayhrs", "Temperature"]].copy()
```

```
In [3]: ###SPLIT RANDOMLY
# split data into predictors (X) and target (y)
#X = data[['Temperature', 'Humidity', 'CO2', 'Light', 'HumidityRatio']]
#y = data['Occupancy']

# split into training and test sets
#X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
In [4]: ###SPLIT SEQUENTIALLY SINCE EACH ROW IS A SUBSEQUENT TIMEPOINT
# calculate index for splitting
split_index = int(len(data) * 0.8)

# split into training and test sets
```

```

train_data = data1.iloc[:split_index]
test_data = data1.iloc[split_index:]

# split data into predictors (X) and target (y)

#x predictor: Working_Hours y target: Light_on_off
X_train1 = train_data[['Working_Hours']]
y_train1 = train_data['Light_on_off']
X_test1 = test_data[['Working_Hours']]
y_test1 = test_data['Light_on_off']

#x predictor: Occupancy y target: Light_on_off
X_train2 = train_data[['Occupancy']]
y_train2 = train_data['Light_on_off']
X_test2 = test_data[['Occupancy']]
y_test2 = test_data['Light_on_off']

#x predictor: Temperature y target: Light_on_off
X_train3 = train_data[['Temperature']]
y_train3 = train_data['Light_on_off']
X_test3 = test_data[['Temperature']]
y_test3 = test_data['Light_on_off']

```

In [5]: # create function to rerun the neural network with different predictors

```

def fcdn(X_train,X_test,y_train,y_test,varname):
    # normalize data
    scaler = StandardScaler()
    X_train_scaled = scaler.fit_transform(X_train)
    X_test_scaled = scaler.transform(X_test)
    #####
    # create model for fully connected (dense) neural network
    model = Sequential([
        Dense(64, activation='relu', input_shape=(X_train_scaled.shape[1],)),
        Dense(32, activation='relu'),
        Dense(1, activation='sigmoid')
    ])
    #####
    # compile model
    model.compile(optimizer='adam',
                  loss='binary_crossentropy',
                  metrics=['accuracy', Precision(name='precision'), Recall(name='recall'), AUC(name='auc')])
    #####
    # train model
    history = model.fit(X_train_scaled, y_train, epochs=100, batch_size=32, validation_split=0.2, verbose=1)
    #####
    # evaluate model
    evaluation_metrics = model.evaluate(X_test_scaled, y_test)
    #####
    # Print evaluation metrics
    result_list = [round(evaluation_metrics[0],8),round(evaluation_metrics[1],8),round(evaluation_metrics[2],8),round(evaluation_metrics[3],8),round(evaluation_m
    #####
    # Predict probabilities
    predicted_probabilities = model.predict(X_test_scaled)
    predicted_labels = predicted_probabilities.flatten()
    # Ensure y_test is in the correct shape

```

```
actual_labels = y_test.values.flatten()
# Generate a time axis based on the length of the test data
time_axis = np.arange(len(y_test))
plt.figure(figsize=(15, 3))
# Plot actual and predicted values
plt.plot(time_axis, actual_labels, label='Actual', linewidth=2, color = "aquamarine")
plt.plot(time_axis, predicted_labels, label='Predicted', linewidth=2, linestyle='--', color = "gold")
plt.title(f"Comparison of Actual and Predicted Light_on_off Over Time by {varname}")
plt.xlabel('Time (minutes)')
plt.ylabel('Light_on_off')
plt.legend()
plt.show()
# save predictions to CSV
df = pd.DataFrame({'predicted_data': predicted_labels, 'actual_data': y_test})
df = df.reset_index(drop=True)
# return result list and dataframe to save to CSV
return result_list,df
```

```
In [6]: #x predictor: Working_Hours y target: Light_on_off
result1,df1 = fcdnn(X_train1,X_test1,y_train1,y_test1,"Working_Hours")
df1.to_csv("FCDNN_results_lightonoffVworkinghours.csv")
```







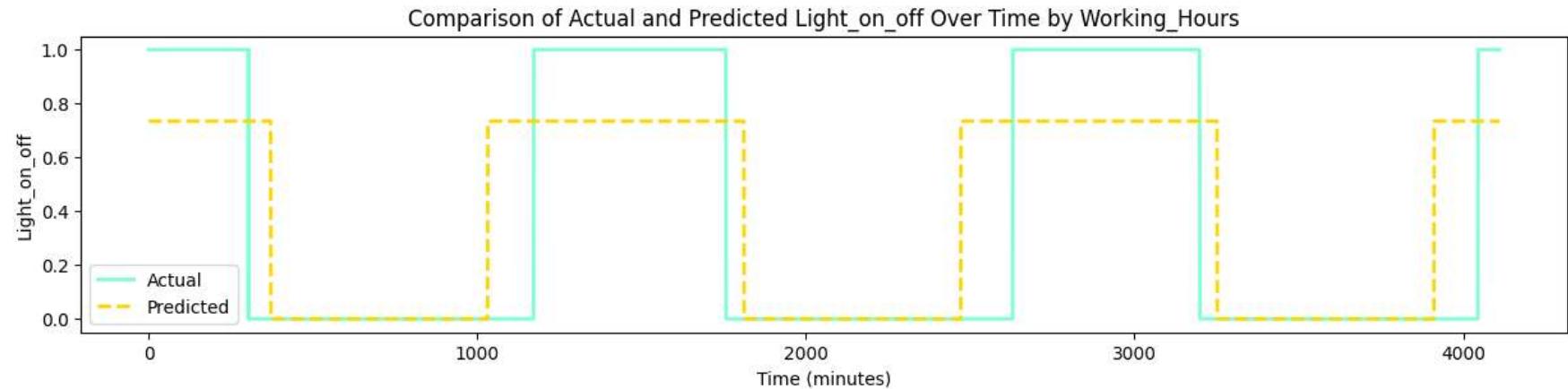




```

412/412 [=====] - 1s 2ms/step - loss: 0.2877 - accuracy: 0.8687 - precision: 0.7366 - recall: 1.0000 - auc: 0.8958 - val_loss: 0.3508 - val_accuracy: 0.8371 - val_precision: 0.7276 - val_recall: 1.0000 - val_auc: 0.8558
Epoch 95/100
412/412 [=====] - 1s 2ms/step - loss: 0.2877 - accuracy: 0.8687 - precision: 0.7366 - recall: 1.0000 - auc: 0.8960 - val_loss: 0.3503 - val_accuracy: 0.8371 - val_precision: 0.7276 - val_recall: 1.0000 - val_auc: 0.8558
Epoch 96/100
412/412 [=====] - 1s 2ms/step - loss: 0.2876 - accuracy: 0.8687 - precision: 0.7366 - recall: 1.0000 - auc: 0.8978 - val_loss: 0.3514 - val_accuracy: 0.8371 - val_precision: 0.7276 - val_recall: 1.0000 - val_auc: 0.8558
Epoch 97/100
412/412 [=====] - 1s 2ms/step - loss: 0.2876 - accuracy: 0.8687 - precision: 0.7366 - recall: 1.0000 - auc: 0.8969 - val_loss: 0.3504 - val_accuracy: 0.8371 - val_precision: 0.7276 - val_recall: 1.0000 - val_auc: 0.8558
Epoch 98/100
412/412 [=====] - 1s 2ms/step - loss: 0.2878 - accuracy: 0.8687 - precision: 0.7366 - recall: 1.0000 - auc: 0.8938 - val_loss: 0.3508 - val_accuracy: 0.8371 - val_precision: 0.7276 - val_recall: 1.0000 - val_auc: 0.8558
Epoch 99/100
412/412 [=====] - 1s 2ms/step - loss: 0.2877 - accuracy: 0.8687 - precision: 0.7366 - recall: 1.0000 - auc: 0.8954 - val_loss: 0.3503 - val_accuracy: 0.8371 - val_precision: 0.7276 - val_recall: 1.0000 - val_auc: 0.8558
Epoch 100/100
412/412 [=====] - 1s 2ms/step - loss: 0.2876 - accuracy: 0.8687 - precision: 0.7366 - recall: 1.0000 - auc: 0.8974 - val_loss: 0.3504 - val_accuracy: 0.8371 - val_precision: 0.7276 - val_recall: 1.0000 - val_auc: 0.8558
129/129 [=====] - 0s 1ms/step - loss: 0.3104 - accuracy: 0.8521 - precision: 0.7148 - recall: 1.0000 - auc: 0.8825
129/129 [=====] - 0s 890us/step

```



```
In [7]: #x predictor: Occupancy y target: Light_on_off
result2,df2 = fcdnn(X_train2,X_test2,y_train2,y_test2,"Occupancy")
df2.to_csv("FCDNN_results_lightonoffVSoccupancy.csv")
```







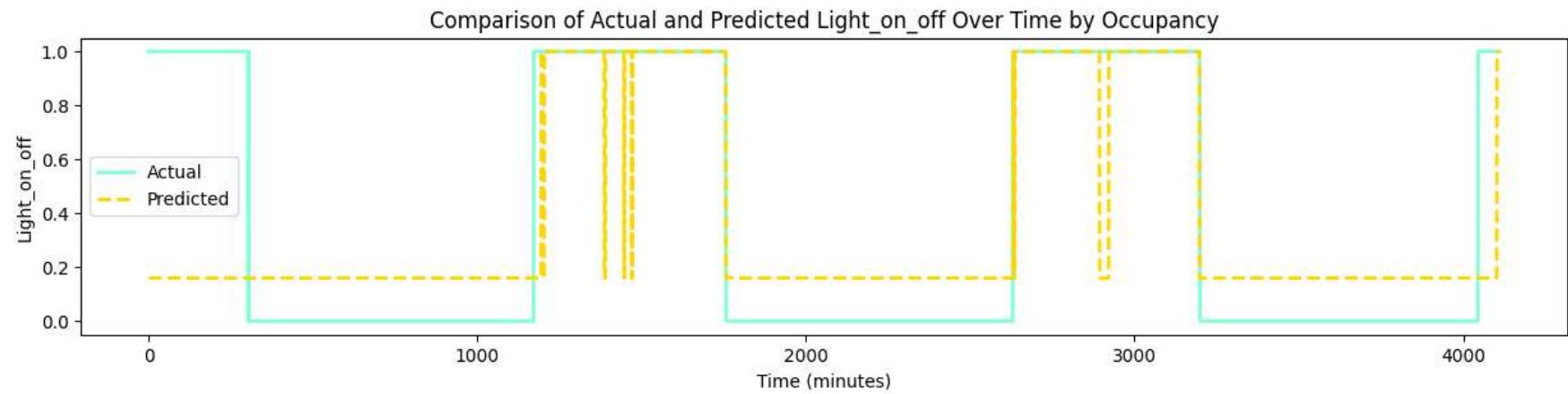




```

412/412 [=====] - 1s 2ms/step - loss: 0.3435 - accuracy: 0.8729 - precision: 1.0000 - recall: 0.6538 - auc: 0.8267 - val_loss: 0.6006 - val_accuracy: 0.7143 - val_precision: 0.9980 - val_recall: 0.3443 - val_auc: 0.6719
Epoch 95/100
412/412 [=====] - 1s 2ms/step - loss: 0.3435 - accuracy: 0.8729 - precision: 1.0000 - recall: 0.6538 - auc: 0.8279 - val_loss: 0.6238 - val_accuracy: 0.7143 - val_precision: 0.9980 - val_recall: 0.3443 - val_auc: 0.6719
Epoch 96/100
412/412 [=====] - 1s 2ms/step - loss: 0.3435 - accuracy: 0.8729 - precision: 1.0000 - recall: 0.6538 - auc: 0.8269 - val_loss: 0.5999 - val_accuracy: 0.7143 - val_precision: 0.9980 - val_recall: 0.3443 - val_auc: 0.6719
Epoch 97/100
412/412 [=====] - 1s 2ms/step - loss: 0.3435 - accuracy: 0.8729 - precision: 1.0000 - recall: 0.6538 - auc: 0.8265 - val_loss: 0.6055 - val_accuracy: 0.7143 - val_precision: 0.9980 - val_recall: 0.3443 - val_auc: 0.6719
Epoch 98/100
412/412 [=====] - 1s 2ms/step - loss: 0.3434 - accuracy: 0.8729 - precision: 1.0000 - recall: 0.6538 - auc: 0.8253 - val_loss: 0.6317 - val_accuracy: 0.7143 - val_precision: 0.9980 - val_recall: 0.3443 - val_auc: 0.6719
Epoch 99/100
412/412 [=====] - 1s 2ms/step - loss: 0.3435 - accuracy: 0.8729 - precision: 1.0000 - recall: 0.6538 - auc: 0.8275 - val_loss: 0.6170 - val_accuracy: 0.7143 - val_precision: 0.9980 - val_recall: 0.3443 - val_auc: 0.6719
Epoch 100/100
412/412 [=====] - 1s 2ms/step - loss: 0.3435 - accuracy: 0.8729 - precision: 1.0000 - recall: 0.6538 - auc: 0.8273 - val_loss: 0.6271 - val_accuracy: 0.7143 - val_precision: 0.9980 - val_recall: 0.3443 - val_auc: 0.6719
129/129 [=====] - 0s 1ms/step - loss: 0.3000 - accuracy: 0.8962 - precision: 1.0000 - recall: 0.7198 - auc: 0.8599
129/129 [=====] - 0s 853us/step

```



```
In [8]: #x predictor: Temperature y target: Light_on_off
result3,df3 = fcdnn(X_train3,X_test3,y_train3,y_test3,"Temperature")
df3.to_csv("FCDNN_results_lightonoffVStemperature.csv")
```

Epoch 1/100  
412/412 [=====] - 2s 2ms/step - loss: 0.4531 - accuracy: 0.8241 - precision: 0.8200 - recall: 0.6674 - auc: 0.8237 - val\_loss: 0.5187 - val\_accuracy: 0.7745 - val\_precision: 1.0000 - val\_recall: 0.4818 - val\_auc: 0.6655  
Epoch 2/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4197 - accuracy: 0.8312 - precision: 0.8533 - recall: 0.6525 - auc: 0.8453 - val\_loss: 0.5288 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.6613  
Epoch 3/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4186 - accuracy: 0.8304 - precision: 0.8489 - recall: 0.6546 - auc: 0.8479 - val\_loss: 0.5397 - val\_accuracy: 0.7754 - val\_precision: 1.0000 - val\_recall: 0.4839 - val\_auc: 0.6938  
Epoch 4/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4182 - accuracy: 0.8311 - precision: 0.8511 - recall: 0.6544 - auc: 0.8495 - val\_loss: 0.5430 - val\_accuracy: 0.7745 - val\_precision: 1.0000 - val\_recall: 0.4818 - val\_auc: 0.6996  
Epoch 5/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4179 - accuracy: 0.8310 - precision: 0.8519 - recall: 0.6534 - auc: 0.8497 - val\_loss: 0.5444 - val\_accuracy: 0.7754 - val\_precision: 1.0000 - val\_recall: 0.4839 - val\_auc: 0.6968  
Epoch 6/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4178 - accuracy: 0.8307 - precision: 0.8491 - recall: 0.6554 - auc: 0.8501 - val\_loss: 0.5407 - val\_accuracy: 0.7769 - val\_precision: 0.9915 - val\_recall: 0.4916 - val\_auc: 0.7389  
Epoch 7/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4178 - accuracy: 0.8322 - precision: 0.8495 - recall: 0.6600 - auc: 0.8498 - val\_loss: 0.5488 - val\_accuracy: 0.7754 - val\_precision: 1.0000 - val\_recall: 0.4839 - val\_auc: 0.7130  
Epoch 8/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4177 - accuracy: 0.8313 - precision: 0.8513 - recall: 0.6550 - auc: 0.8500 - val\_loss: 0.5580 - val\_accuracy: 0.7681 - val\_precision: 1.0000 - val\_recall: 0.4672 - val\_auc: 0.7041  
Epoch 9/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4172 - accuracy: 0.8314 - precision: 0.8512 - recall: 0.6556 - auc: 0.8508 - val\_loss: 0.5581 - val\_accuracy: 0.7739 - val\_precision: 1.0000 - val\_recall: 0.4804 - val\_auc: 0.7199  
Epoch 10/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4172 - accuracy: 0.8307 - precision: 0.8511 - recall: 0.6531 - auc: 0.8500 - val\_loss: 0.5495 - val\_accuracy: 0.7681 - val\_precision: 1.0000 - val\_recall: 0.4672 - val\_auc: 0.6998  
Epoch 11/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4177 - accuracy: 0.8310 - precision: 0.8500 - recall: 0.6554 - auc: 0.8510 - val\_loss: 0.5369 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.6897  
Epoch 12/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4173 - accuracy: 0.8307 - precision: 0.8483 - recall: 0.6562 - auc: 0.8512 - val\_loss: 0.5483 - val\_accuracy: 0.7739 - val\_precision: 1.0000 - val\_recall: 0.4804 - val\_auc: 0.6752  
Epoch 13/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4168 - accuracy: 0.8308 - precision: 0.8484 - recall: 0.6567 - auc: 0.8516 - val\_loss: 0.5669 - val\_accuracy: 0.7681 - val\_precision: 1.0000 - val\_recall: 0.4672 - val\_auc: 0.7109  
Epoch 14/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4173 - accuracy: 0.8308 - precision: 0.8503 - recall: 0.6546 - auc: 0.8509 - val\_loss: 0.5708 - val\_accuracy: 0.7663 - val\_precision: 1.0000 - val\_recall: 0.4630 - val\_auc: 0.7036  
Epoch 15/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4173 - accuracy: 0.8304 - precision: 0.8491 - recall: 0.6544 - auc: 0.8507 - val\_loss: 0.5580 - val\_accuracy: 0.7745 - val\_precision: 1.0000 - val\_recall: 0.4818 - val\_auc: 0.7101  
Epoch 16/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4172 - accuracy: 0.8305 - precision: 0.8458 - recall: 0.6585 - auc: 0.8506 - val\_loss: 0.5513 - val\_accuracy: 0.7739 - val\_precision: 1.0000 - val\_recall: 0.4804 - val\_auc: 0.6823  
Epoch 17/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4172 - accuracy: 0.8310 - precision: 0.8490 - recall: 0.6565 - auc: 0.8511 - val\_loss: 0.5613 - val\_accuracy: 0.7739 - val\_precision: 1.0000 - val\_recall: 0.4804 - val\_auc: 0.7109  
Epoch 18/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4169 - accuracy: 0.8306 - precision: 0.8466 - recall: 0.6579 - auc: 0.8513 - val\_loss: 0.5663 - val\_accuracy: 0.7681 - val\_precision: 1.0000 - val\_recall: 0.4672 - val\_auc: 0.7215  
Epoch 19/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4173 - accuracy: 0.8314 - precision: 0.8487 - recall: 0.6581 - auc: 0.8502 - val\_loss: 0.5579 - val\_accuracy: 0.7745 - val\_precision: 1.0000 - val\_recall: 0.4818 - val\_auc: 0.6968

al\_accuracy: 0.7754 - val\_precision: 1.0000 - val\_recall: 0.4839 - val\_auc: 0.7285  
Epoch 20/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4165 - accuracy: 0.8314 - precision: 0.8492 - recall: 0.6575 - auc: 0.8520 - val\_loss: 0.5556 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7253  
Epoch 21/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4168 - accuracy: 0.8308 - precision: 0.8486 - recall: 0.6565 - auc: 0.8514 - val\_loss: 0.5538 - val\_accuracy: 0.7745 - val\_precision: 1.0000 - val\_recall: 0.4818 - val\_auc: 0.7229  
Epoch 22/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4169 - accuracy: 0.8305 - precision: 0.8471 - recall: 0.6571 - auc: 0.8517 - val\_loss: 0.5619 - val\_accuracy: 0.7739 - val\_precision: 1.0000 - val\_recall: 0.4804 - val\_auc: 0.7226  
Epoch 23/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4163 - accuracy: 0.8310 - precision: 0.8487 - recall: 0.6569 - auc: 0.8525 - val\_loss: 0.5475 - val\_accuracy: 0.7754 - val\_precision: 1.0000 - val\_recall: 0.4839 - val\_auc: 0.7396  
Epoch 24/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4162 - accuracy: 0.8306 - precision: 0.8481 - recall: 0.6562 - auc: 0.8529 - val\_loss: 0.5307 - val\_accuracy: 0.7769 - val\_precision: 0.9902 - val\_recall: 0.4923 - val\_auc: 0.7446  
Epoch 25/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4165 - accuracy: 0.8315 - precision: 0.8478 - recall: 0.6596 - auc: 0.8514 - val\_loss: 0.5684 - val\_accuracy: 0.7745 - val\_precision: 1.0000 - val\_recall: 0.4818 - val\_auc: 0.7274  
Epoch 26/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4161 - accuracy: 0.8304 - precision: 0.8487 - recall: 0.6548 - auc: 0.8525 - val\_loss: 0.5598 - val\_accuracy: 0.7754 - val\_precision: 1.0000 - val\_recall: 0.4839 - val\_auc: 0.7550  
Epoch 27/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4160 - accuracy: 0.8308 - precision: 0.8475 - recall: 0.6577 - auc: 0.8525 - val\_loss: 0.5555 - val\_accuracy: 0.7754 - val\_precision: 1.0000 - val\_recall: 0.4839 - val\_auc: 0.7284  
Epoch 28/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4158 - accuracy: 0.8304 - precision: 0.8467 - recall: 0.6573 - auc: 0.8537 - val\_loss: 0.5577 - val\_accuracy: 0.7745 - val\_precision: 1.0000 - val\_recall: 0.4818 - val\_auc: 0.7141  
Epoch 29/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4156 - accuracy: 0.8311 - precision: 0.8445 - recall: 0.6618 - auc: 0.8535 - val\_loss: 0.5592 - val\_accuracy: 0.7754 - val\_precision: 1.0000 - val\_recall: 0.4839 - val\_auc: 0.7354  
Epoch 30/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4158 - accuracy: 0.8308 - precision: 0.8467 - recall: 0.6585 - auc: 0.8529 - val\_loss: 0.5514 - val\_accuracy: 0.7754 - val\_precision: 1.0000 - val\_recall: 0.4839 - val\_auc: 0.7287  
Epoch 31/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4158 - accuracy: 0.8305 - precision: 0.8464 - recall: 0.6579 - auc: 0.8519 - val\_loss: 0.5486 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7254  
Epoch 32/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4158 - accuracy: 0.8309 - precision: 0.8446 - recall: 0.6612 - auc: 0.8530 - val\_loss: 0.5538 - val\_accuracy: 0.7754 - val\_precision: 1.0000 - val\_recall: 0.4839 - val\_auc: 0.7190  
Epoch 33/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4155 - accuracy: 0.8310 - precision: 0.8457 - recall: 0.6602 - auc: 0.8522 - val\_loss: 0.5444 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7169  
Epoch 34/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4159 - accuracy: 0.8311 - precision: 0.8445 - recall: 0.6618 - auc: 0.8526 - val\_loss: 0.5774 - val\_accuracy: 0.7739 - val\_precision: 1.0000 - val\_recall: 0.4804 - val\_auc: 0.7400  
Epoch 35/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4155 - accuracy: 0.8299 - precision: 0.8446 - recall: 0.6579 - auc: 0.8534 - val\_loss: 0.5518 - val\_accuracy: 0.7769 - val\_precision: 0.9915 - val\_recall: 0.4916 - val\_auc: 0.7269  
Epoch 36/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4153 - accuracy: 0.8310 - precision: 0.8435 - recall: 0.6627 - auc: 0.8532 - val\_loss: 0.5658 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7335  
Epoch 37/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4161 - accuracy: 0.8301 - precision: 0.8407 - recall: 0.6631 - auc: 0.8526 - val\_loss: 0.5513 - val\_accuracy: 0.7745 - val\_precision: 1.0000 - val\_recall: 0.4818 - val\_auc: 0.7146  
Epoch 38/100

412/412 [=====] - 1s 2ms/step - loss: 0.4151 - accuracy: 0.8310 - precision: 0.8432 - recall: 0.6631 - auc: 0.8547 - val\_loss: 0.5395 - val\_accuracy: 0.7769 - val\_precision: 0.9915 - val\_recall: 0.4916 - val\_auc: 0.7311  
Epoch 39/100

412/412 [=====] - 1s 2ms/step - loss: 0.4149 - accuracy: 0.8313 - precision: 0.8426 - recall: 0.6647 - auc: 0.8548 - val\_loss: 0.5498 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7238  
Epoch 40/100

412/412 [=====] - 1s 2ms/step - loss: 0.4147 - accuracy: 0.8319 - precision: 0.8444 - recall: 0.6647 - auc: 0.8545 - val\_loss: 0.5772 - val\_accuracy: 0.7681 - val\_precision: 1.0000 - val\_recall: 0.4672 - val\_auc: 0.7441  
Epoch 41/100

412/412 [=====] - 1s 2ms/step - loss: 0.4148 - accuracy: 0.8311 - precision: 0.8432 - recall: 0.6635 - auc: 0.8551 - val\_loss: 0.5527 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7254  
Epoch 42/100

412/412 [=====] - 1s 2ms/step - loss: 0.4149 - accuracy: 0.8309 - precision: 0.8419 - recall: 0.6643 - auc: 0.8545 - val\_loss: 0.5738 - val\_accuracy: 0.7754 - val\_precision: 1.0000 - val\_recall: 0.4839 - val\_auc: 0.7405  
Epoch 43/100

412/412 [=====] - 1s 2ms/step - loss: 0.4145 - accuracy: 0.8314 - precision: 0.8417 - recall: 0.6660 - auc: 0.8555 - val\_loss: 0.5972 - val\_accuracy: 0.7754 - val\_precision: 1.0000 - val\_recall: 0.4839 - val\_auc: 0.7402  
Epoch 44/100

412/412 [=====] - 1s 2ms/step - loss: 0.4145 - accuracy: 0.8314 - precision: 0.8428 - recall: 0.6647 - auc: 0.8559 - val\_loss: 0.5565 - val\_accuracy: 0.7769 - val\_precision: 0.9915 - val\_recall: 0.4916 - val\_auc: 0.7193  
Epoch 45/100

412/412 [=====] - 1s 2ms/step - loss: 0.4142 - accuracy: 0.8312 - precision: 0.8401 - recall: 0.6674 - auc: 0.8557 - val\_loss: 0.5506 - val\_accuracy: 0.7769 - val\_precision: 0.9915 - val\_recall: 0.4916 - val\_auc: 0.7290  
Epoch 46/100

412/412 [=====] - 1s 2ms/step - loss: 0.4148 - accuracy: 0.8304 - precision: 0.8409 - recall: 0.6639 - auc: 0.8548 - val\_loss: 0.5425 - val\_accuracy: 0.7769 - val\_precision: 0.9915 - val\_recall: 0.4916 - val\_auc: 0.7027  
Epoch 47/100

412/412 [=====] - 1s 2ms/step - loss: 0.4147 - accuracy: 0.8311 - precision: 0.8410 - recall: 0.6658 - auc: 0.8545 - val\_loss: 0.5548 - val\_accuracy: 0.7754 - val\_precision: 1.0000 - val\_recall: 0.4839 - val\_auc: 0.7161  
Epoch 48/100

412/412 [=====] - 1s 2ms/step - loss: 0.4147 - accuracy: 0.8307 - precision: 0.8407 - recall: 0.6651 - auc: 0.8554 - val\_loss: 0.5441 - val\_accuracy: 0.7769 - val\_precision: 0.9902 - val\_recall: 0.4923 - val\_auc: 0.7125  
Epoch 49/100

412/412 [=====] - 1s 2ms/step - loss: 0.4152 - accuracy: 0.8304 - precision: 0.8405 - recall: 0.6643 - auc: 0.8541 - val\_loss: 0.5499 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7389  
Epoch 50/100

412/412 [=====] - 1s 2ms/step - loss: 0.4149 - accuracy: 0.8315 - precision: 0.8422 - recall: 0.6660 - auc: 0.8543 - val\_loss: 0.5304 - val\_accuracy: 0.7769 - val\_precision: 0.9902 - val\_recall: 0.4923 - val\_auc: 0.7124  
Epoch 51/100

412/412 [=====] - 1s 2ms/step - loss: 0.4139 - accuracy: 0.8316 - precision: 0.8421 - recall: 0.6664 - auc: 0.8567 - val\_loss: 0.5388 - val\_accuracy: 0.7769 - val\_precision: 0.9902 - val\_recall: 0.4923 - val\_auc: 0.7333  
Epoch 52/100

412/412 [=====] - 1s 2ms/step - loss: 0.4142 - accuracy: 0.8323 - precision: 0.8396 - recall: 0.6716 - auc: 0.8550 - val\_loss: 0.5646 - val\_accuracy: 0.7745 - val\_precision: 1.0000 - val\_recall: 0.4818 - val\_auc: 0.7282  
Epoch 53/100

412/412 [=====] - 1s 2ms/step - loss: 0.4146 - accuracy: 0.8311 - precision: 0.8403 - recall: 0.6666 - auc: 0.8551 - val\_loss: 0.5539 - val\_accuracy: 0.7769 - val\_precision: 0.9915 - val\_recall: 0.4916 - val\_auc: 0.7430  
Epoch 54/100

412/412 [=====] - 1s 2ms/step - loss: 0.4141 - accuracy: 0.8316 - precision: 0.8394 - recall: 0.6695 - auc: 0.8562 - val\_loss: 0.5472 - val\_accuracy: 0.7754 - val\_precision: 1.0000 - val\_recall: 0.4839 - val\_auc: 0.7102  
Epoch 55/100

412/412 [=====] - 1s 2ms/step - loss: 0.4144 - accuracy: 0.8318 - precision: 0.8411 - recall: 0.6683 - auc: 0.8553 - val\_loss: 0.5398 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7156  
Epoch 56/100

412/412 [=====] - 1s 2ms/step - loss: 0.4141 - accuracy: 0.8306 - precision: 0.8366 - recall: 0.6695 - auc: 0.8561 - val\_loss: 0.5366 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7215

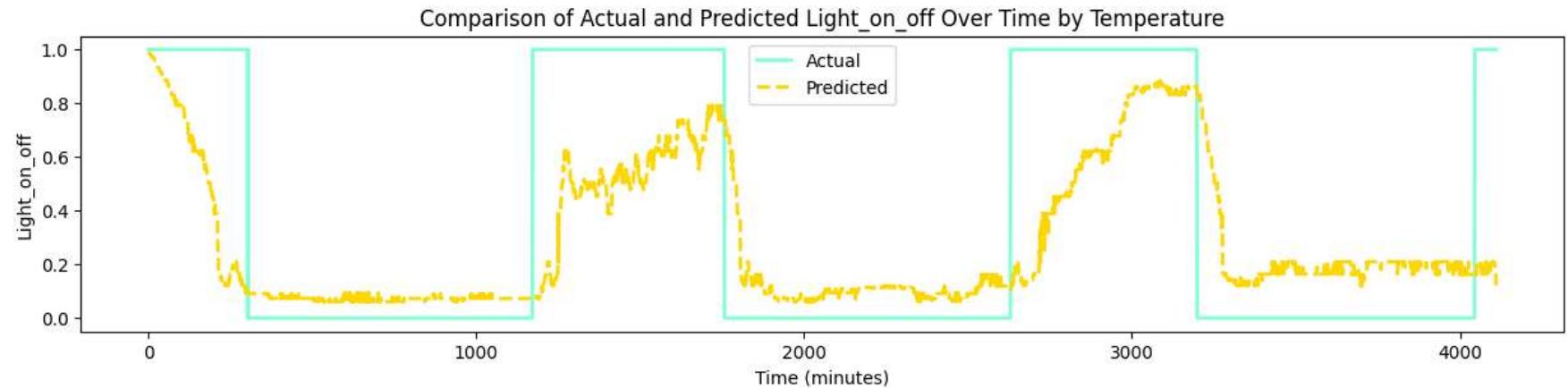
Epoch 57/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4139 - accuracy: 0.8310 - precision: 0.8392 - recall: 0.6676 - auc: 0.8561 - val\_loss: 0.5621 - val\_accuracy: 0.7754 - val\_precision: 1.0000 - val\_recall: 0.4839 - val\_auc: 0.7354  
Epoch 58/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4139 - accuracy: 0.8310 - precision: 0.8366 - recall: 0.6707 - auc: 0.8554 - val\_loss: 0.5477 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7345  
Epoch 59/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4140 - accuracy: 0.8322 - precision: 0.8413 - recall: 0.6693 - auc: 0.8563 - val\_loss: 0.5482 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7338  
Epoch 60/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4140 - accuracy: 0.8311 - precision: 0.8395 - recall: 0.6678 - auc: 0.8563 - val\_loss: 0.5525 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7424  
Epoch 61/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4140 - accuracy: 0.8311 - precision: 0.8402 - recall: 0.6670 - auc: 0.8562 - val\_loss: 0.5633 - val\_accuracy: 0.7769 - val\_precision: 0.9915 - val\_recall: 0.4916 - val\_auc: 0.7211  
Epoch 62/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4140 - accuracy: 0.8303 - precision: 0.8362 - recall: 0.6689 - auc: 0.8566 - val\_loss: 0.5689 - val\_accuracy: 0.7754 - val\_precision: 1.0000 - val\_recall: 0.4839 - val\_auc: 0.7253  
Epoch 63/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4141 - accuracy: 0.8313 - precision: 0.8413 - recall: 0.6662 - auc: 0.8557 - val\_loss: 0.5588 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7463  
Epoch 64/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4144 - accuracy: 0.8317 - precision: 0.8401 - recall: 0.6689 - auc: 0.8551 - val\_loss: 0.5493 - val\_accuracy: 0.7769 - val\_precision: 0.9915 - val\_recall: 0.4916 - val\_auc: 0.7314  
Epoch 65/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4135 - accuracy: 0.8312 - precision: 0.8385 - recall: 0.6693 - auc: 0.8566 - val\_loss: 0.5573 - val\_accuracy: 0.7769 - val\_precision: 0.9915 - val\_recall: 0.4916 - val\_auc: 0.7298  
Epoch 66/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4136 - accuracy: 0.8311 - precision: 0.8377 - recall: 0.6697 - auc: 0.8566 - val\_loss: 0.5556 - val\_accuracy: 0.7754 - val\_precision: 1.0000 - val\_recall: 0.4839 - val\_auc: 0.7326  
Epoch 67/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4139 - accuracy: 0.8314 - precision: 0.8400 - recall: 0.6683 - auc: 0.8559 - val\_loss: 0.5423 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7294  
Epoch 68/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4141 - accuracy: 0.8311 - precision: 0.8400 - recall: 0.6672 - auc: 0.8564 - val\_loss: 0.5538 - val\_accuracy: 0.7745 - val\_precision: 1.0000 - val\_recall: 0.4818 - val\_auc: 0.7087  
Epoch 69/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4141 - accuracy: 0.8321 - precision: 0.8414 - recall: 0.6689 - auc: 0.8567 - val\_loss: 0.5310 - val\_accuracy: 0.7769 - val\_precision: 0.9902 - val\_recall: 0.4923 - val\_auc: 0.7329  
Epoch 70/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4136 - accuracy: 0.8311 - precision: 0.8383 - recall: 0.6693 - auc: 0.8566 - val\_loss: 0.5595 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7300  
Epoch 71/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4140 - accuracy: 0.8305 - precision: 0.8393 - recall: 0.6660 - auc: 0.8558 - val\_loss: 0.5373 - val\_accuracy: 0.7757 - val\_precision: 0.9846 - val\_recall: 0.4923 - val\_auc: 0.7291  
Epoch 72/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4134 - accuracy: 0.8319 - precision: 0.8399 - recall: 0.6699 - auc: 0.8575 - val\_loss: 0.5603 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7315  
Epoch 73/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4138 - accuracy: 0.8318 - precision: 0.8407 - recall: 0.6687 - auc: 0.8564 - val\_loss: 0.5533 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7278  
Epoch 74/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4142 - accuracy: 0.8310 - precision: 0.8385 - recall: 0.6685 - auc: 0.8565 - val\_loss: 0.5598 - val\_accuracy: 0.7769 - val\_precision: 0.9915 - val\_recall: 0.4916 - val\_auc: 0.7343  
Epoch 75/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4138 - accuracy: 0.8317 - precision: 0.8414 - recall: 0.6676 - auc: 0.8559 - val\_loss: 0.5481 - val\_accuracy: 0.7769 - val\_precision: 0.9915 - val\_recall: 0.4916 - val\_auc: 0.7211

al\_accuracy: 0.7769 - val\_precision: 0.9915 - val\_recall: 0.4916 - val\_auc: 0.7219  
Epoch 76/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4133 - accuracy: 0.8317 - precision: 0.8402 - recall: 0.6691 - auc: 0.8570 - val\_loss: 0.5637 - val\_accuracy: 0.7754 - val\_precision: 1.0000 - val\_recall: 0.4839 - val\_auc: 0.7320  
Epoch 77/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4137 - accuracy: 0.8307 - precision: 0.8405 - recall: 0.6651 - auc: 0.8562 - val\_loss: 0.5483 - val\_accuracy: 0.7769 - val\_precision: 0.9902 - val\_recall: 0.4923 - val\_auc: 0.7359  
Epoch 78/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4139 - accuracy: 0.8326 - precision: 0.8382 - recall: 0.6743 - auc: 0.8562 - val\_loss: 0.5689 - val\_accuracy: 0.7745 - val\_precision: 1.0000 - val\_recall: 0.4818 - val\_auc: 0.7382  
Epoch 79/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4139 - accuracy: 0.8313 - precision: 0.8408 - recall: 0.6668 - auc: 0.8558 - val\_loss: 0.5554 - val\_accuracy: 0.7754 - val\_precision: 1.0000 - val\_recall: 0.4839 - val\_auc: 0.7316  
Epoch 80/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4134 - accuracy: 0.8308 - precision: 0.8395 - recall: 0.6668 - auc: 0.8569 - val\_loss: 0.5491 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7320  
Epoch 81/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4134 - accuracy: 0.8311 - precision: 0.8376 - recall: 0.6701 - auc: 0.8566 - val\_loss: 0.5490 - val\_accuracy: 0.7754 - val\_precision: 1.0000 - val\_recall: 0.4839 - val\_auc: 0.7286  
Epoch 82/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4135 - accuracy: 0.8318 - precision: 0.8372 - recall: 0.6728 - auc: 0.8564 - val\_loss: 0.5451 - val\_accuracy: 0.7769 - val\_precision: 0.9915 - val\_recall: 0.4916 - val\_auc: 0.7393  
Epoch 83/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4132 - accuracy: 0.8325 - precision: 0.8417 - recall: 0.6699 - auc: 0.8572 - val\_loss: 0.5560 - val\_accuracy: 0.7769 - val\_precision: 0.9915 - val\_recall: 0.4916 - val\_auc: 0.7392  
Epoch 84/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4133 - accuracy: 0.8309 - precision: 0.8355 - recall: 0.6718 - auc: 0.8567 - val\_loss: 0.5465 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7302  
Epoch 85/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4133 - accuracy: 0.8313 - precision: 0.8394 - recall: 0.6685 - auc: 0.8572 - val\_loss: 0.5613 - val\_accuracy: 0.7745 - val\_precision: 1.0000 - val\_recall: 0.4818 - val\_auc: 0.7320  
Epoch 86/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4132 - accuracy: 0.8309 - precision: 0.8397 - recall: 0.6668 - auc: 0.8574 - val\_loss: 0.5574 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7349  
Epoch 87/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4135 - accuracy: 0.8306 - precision: 0.8357 - recall: 0.6705 - auc: 0.8564 - val\_loss: 0.5360 - val\_accuracy: 0.7757 - val\_precision: 0.9846 - val\_recall: 0.4923 - val\_auc: 0.7386  
Epoch 88/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4130 - accuracy: 0.8318 - precision: 0.8386 - recall: 0.6712 - auc: 0.8575 - val\_loss: 0.5478 - val\_accuracy: 0.7745 - val\_precision: 1.0000 - val\_recall: 0.4818 - val\_auc: 0.7285  
Epoch 89/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4137 - accuracy: 0.8316 - precision: 0.8376 - recall: 0.6716 - auc: 0.8566 - val\_loss: 0.5409 - val\_accuracy: 0.7769 - val\_precision: 0.9902 - val\_recall: 0.4923 - val\_auc: 0.7316  
Epoch 90/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4131 - accuracy: 0.8318 - precision: 0.8388 - recall: 0.6709 - auc: 0.8569 - val\_loss: 0.5433 - val\_accuracy: 0.7769 - val\_precision: 0.9915 - val\_recall: 0.4916 - val\_auc: 0.7407  
Epoch 91/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4131 - accuracy: 0.8318 - precision: 0.8395 - recall: 0.6701 - auc: 0.8575 - val\_loss: 0.5514 - val\_accuracy: 0.7757 - val\_precision: 0.9846 - val\_recall: 0.4923 - val\_auc: 0.7455  
Epoch 92/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4132 - accuracy: 0.8303 - precision: 0.8347 - recall: 0.6707 - auc: 0.8570 - val\_loss: 0.5570 - val\_accuracy: 0.7769 - val\_precision: 0.9915 - val\_recall: 0.4916 - val\_auc: 0.7385  
Epoch 93/100  
412/412 [=====] - 1s 2ms/step - loss: 0.4134 - accuracy: 0.8311 - precision: 0.8369 - recall: 0.6709 - auc: 0.8565 - val\_loss: 0.5589 - val\_accuracy: 0.7757 - val\_precision: 1.0000 - val\_recall: 0.4846 - val\_auc: 0.7188  
Epoch 94/100

```

412/412 [=====] - 1s 2ms/step - loss: 0.4135 - accuracy: 0.8309 - precision: 0.8355 - recall: 0.6718 - auc: 0.8575 - val_loss: 0.5452 - val_accuracy: 0.7757 - val_precision: 1.0000 - val_recall: 0.4846 - val_auc: 0.7261
Epoch 95/100
412/412 [=====] - 1s 2ms/step - loss: 0.4128 - accuracy: 0.8320 - precision: 0.8396 - recall: 0.6705 - auc: 0.8577 - val_loss: 0.5636 - val_accuracy: 0.7769 - val_precision: 0.9915 - val_recall: 0.4916 - val_auc: 0.7228
Epoch 96/100
412/412 [=====] - 1s 2ms/step - loss: 0.4128 - accuracy: 0.8318 - precision: 0.8379 - recall: 0.6720 - auc: 0.8575 - val_loss: 0.5452 - val_accuracy: 0.7769 - val_precision: 0.9915 - val_recall: 0.4916 - val_auc: 0.7057
Epoch 97/100
412/412 [=====] - 1s 2ms/step - loss: 0.4132 - accuracy: 0.8307 - precision: 0.8392 - recall: 0.6666 - auc: 0.8567 - val_loss: 0.5568 - val_accuracy: 0.7769 - val_precision: 0.9915 - val_recall: 0.4916 - val_auc: 0.7382
Epoch 98/100
412/412 [=====] - 1s 2ms/step - loss: 0.4132 - accuracy: 0.8309 - precision: 0.8362 - recall: 0.6709 - auc: 0.8567 - val_loss: 0.5424 - val_accuracy: 0.7769 - val_precision: 0.9915 - val_recall: 0.4916 - val_auc: 0.7280
Epoch 99/100
412/412 [=====] - 1s 2ms/step - loss: 0.4132 - accuracy: 0.8317 - precision: 0.8359 - recall: 0.6738 - auc: 0.8572 - val_loss: 0.5616 - val_accuracy: 0.7745 - val_precision: 1.0000 - val_recall: 0.4818 - val_auc: 0.7316
Epoch 100/100
412/412 [=====] - 1s 2ms/step - loss: 0.4129 - accuracy: 0.8323 - precision: 0.8372 - recall: 0.6747 - auc: 0.8577 - val_loss: 0.5731 - val_accuracy: 0.7739 - val_precision: 1.0000 - val_recall: 0.4804 - val_auc: 0.7167
129/129 [=====] - 0s 1ms/step - loss: 0.3903 - accuracy: 0.8473 - precision: 0.9103 - recall: 0.6522 - auc: 0.9178
129/129 [=====] - 0s 860us/step

```



```
In [9]: # create dataframe to contain all results
templists = [result1,result2,result3]
fcdnn_results = pd.DataFrame(templists)
fcdnn_results.columns=['Test Loss', 'Test Accuracy', 'Test Precision', 'Test Recall', 'Test AUC']
fcdnn_results = fcdnn_results.rename(index={0: 'Working_Hours',1: 'Occupancy',2: 'Temperature'})
fcdnn_results
```

	Test Loss	Test Accuracy	Test Precision	Test Recall	Test AUC
Working_Hours	0.310436	0.852140	0.714822	1.000000	0.882535
Occupancy	0.299999	0.896158	1.000000	0.719816	0.859908
Temperature	0.390315	0.847276	0.910256	0.652231	0.917770

