Introduction to Machine Learning Assignment 2 Report

Polina Korobeinikova

April 2025

1 Task 1: Exploratory Data Analysis and Preprocessing

The raw dataset from `opsd_raw.csv` was loaded, containing hourly power-related measurements for Denmark from 2015 to 2020. Per the `README.md`, I extracted three features: total power load (`DK_load_actual_entsoe_transparency`), wind generation (`DK_wind_generation_actual`), and solar generation (`DK_solar_generation_actual`), all in MW. These features are the most suitable for the task, referring to README.

The dataset comprised 50,400 hourly records, with initial missing values: 2 in power load, 2 in wind generation, and 11 in solar generation. Missing values were handled as follows: power load and wind generation were imputed using forward and backward filling (`ffill` and `bfill`), suitable due to their edge placement, preserving temporal trends. Solar generation NaNs were filled with means grouped by month and hour, reflecting seasonal daylight patterns. Then hourly data was aggregated into daily 24-hour arrays per feature, verified to ensure each day had exactly 24 entries.

Seasons were assigned based on months. Average daily profiles (Figure 1) revealed: (1) power load peaks in winter (5000 MW at 17:00) due to heating demands; (2) wind generation is highest in winter, likely due to stormier weather; (3) biggest solar generation in summer with a bell-shaped curve, reflecting longer daylight hours.

Data was split into 70% training, 15% validation, and 15% test sets using stratified sampling to maintain season proportions. Each feature was standardized separately using `StandardScaler`, fitting on training data only to avoid leakage.

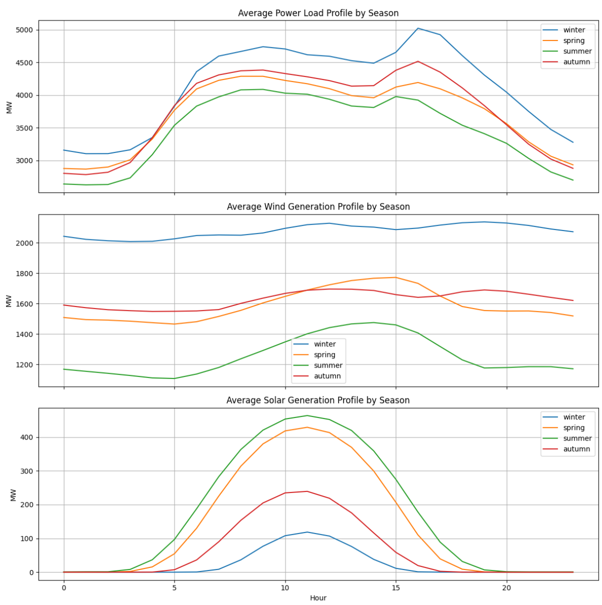


Figure 1: Average 24-hour profiles for power load, wind, and solar generation by season.

2 Task 2: Baseline MLP

A Multi-Layer Perceptron (MLP) with two hidden layers classified the flattened 72-dimensional input (3 features $\times$ 24 hours) into four seasons. Two hidden layers were chosen to capture complex, non-linear relationships beyond a single layer's capacity. Grid search tuned `hidden_size1, hidden_size2` $\in \{32, 64, 128, 256\}$ and `lr` $\in \{0.001, 0.005, 0.01\}$, with Adam optimization for its adaptive learning rate and faster convergence compared to SGD. Best parameters (`hidden_size1=128`, `hidden_size2=64`, `lr=0.001`) yielded a test accuracy of 85.08%. Training curves (Figure 2) and the confusion matrix (Figure 5) show stable learning and strong summer/autumn performance, with some summer-spring confusion likely due to weather similarities.

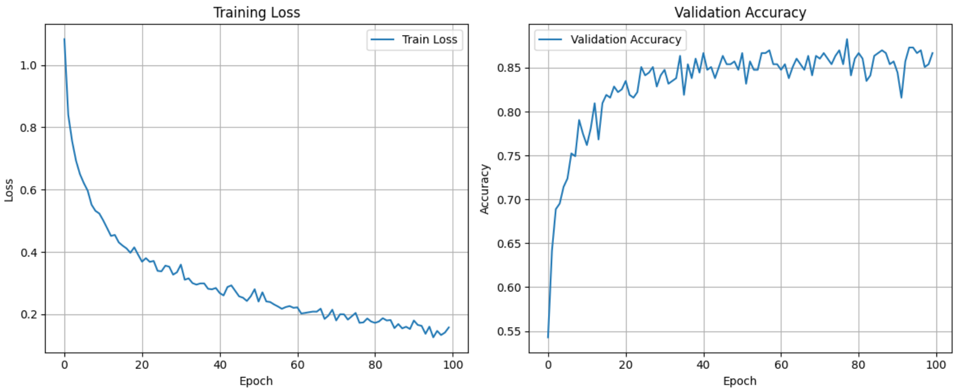


Figure 2: Training loss and validation accuracy for the MLP.

3 Task 3: 1D-CNN on Raw Time-Series

I implemented a 1D Convolutional Neural Network (1D-CNN) to classify 24-hour time series data (shape: $n_{\text{days}} \times 3 \times 24$) into four seasons. The architecture included one `Conv1d` layer (3 input channels, tunable `num_filters` and `kernel_size`), a `MaxPool1d` layer (kernel size 2), and two fully connected layers (`hidden_size1`, `hidden_size2`), with ReLU activations and dropout for regularization. Hyperparameters were tuned via grid search: `num_filters` $\in \{8, 16, 32\}$, `kernel_size` $\in \{3, 5\}$, `hidden_size1, hidden_size2` $\in \{32, 64, 128, 256\}$, `dropout_rate` $\in \{0.1, 0.3, 0.5\}$, and `lr` $\in \{0.0001, 0.001, 0.01\}$, with Adam optimization with its adaptive learning.

The best configuration (`num_filters`=16, `kernel_size`=3, `hidden_size1`=256, `hidden_size2`=128, `dropout_rate`=0.3, `lr`=0.001) achieved a validation accuracy of 85.08% and a test accuracy of 86.98% after 100 epochs. Figure 3 shows training loss and validation accuracy trends, while the confusion matrix (Figure 5) highlights improved differentiation of seasons compared to the MLP. The 1D-CNN's performance gain of 1.9% over the MLP suggests it effectively captures local temporal patterns.

4 Task 4: 2D Transform & 2D-CNN

I transformed the 24-hour time-series data (shape: $n_{\text{days}} \times 3 \times 24$) into a 2D "image" by reshaping each feature's 24 time steps into a 4×6 matrix, resulting in a shape of $(n_{\text{days}}, 3, 4, 6)$, where the 3 channels represent power load, wind, and solar generation. This approach, inspired by Option B in the assignment (1), enables the 2D-CNN to apply spatial convolutions, dividing the day into four 6-hour segments (e.g., morning, afternoon) to capture intra-day patterns.

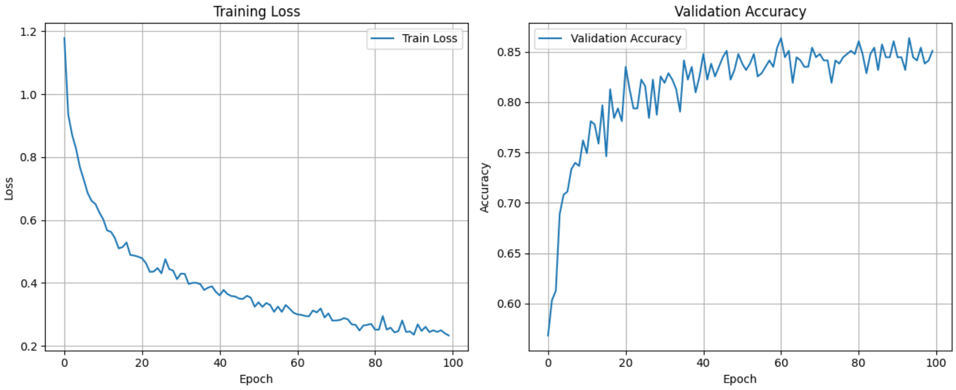


Figure 3: Training loss and validation accuracy for the 1D-CNN.

A 2D-CNN with two `Conv2d` layers achieved a test accuracy of 87.30%, slightly outperforming the 1D-CNN (86.98%) and MLP (85.08%). The 2D transformation helped to gain more information for the model, as shown in the accuracy results. In my opinion, transformation to 4 rows and 6 columns helped model to capture patterns in different parts of the day, like morning, afternoon, evening, night.

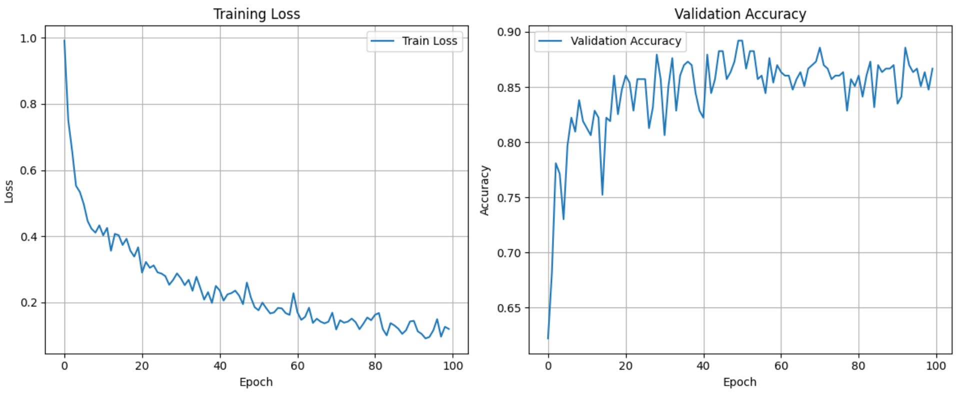
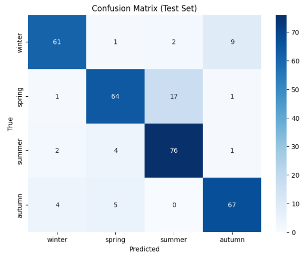


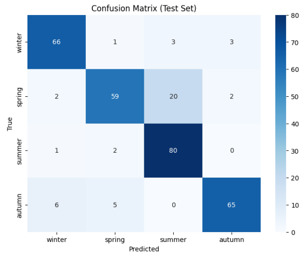
Figure 4: Training loss and validation accuracy for the 2D-CNN.

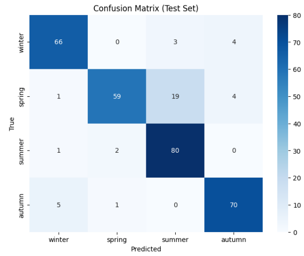
5 Models Comparison

Model	Test Accuracy
MLP	85.08%
1D-CNN	86.98%
2D-CNN	87.30%

The 1D-CNN's direct temporal processing provided a more balanced performance across seasons, while the MLP, lacking temporal structure, had the lowest accuracy. The results suggest that convolutional models outperform the MLP by capturing local patterns, with the 2D-CNN's slight edge indicating potential in 2D transformations.



(MLP), 

(1D-CNN), 

(2D-CNN).

The 2D-CNN achieved the highest test accuracy (87.30%), followed by the 1D-CNN (86.98%) and MLP (85.08%). Per-class analysis reveals distinct strengths:

- **Winter:** The 2D-CNN excels (F1-score: 0.92 vs. 0.89 for 1D-CNN, 0.87 for MLP), likely due to the 4×6 transformation highlighting winter's high power load and wind generation patterns across different day time.

- **Spring:** The 1D-CNN performs best (recall: 0.80 vs. 0.75 for 2D-CNN), as its temporal processing captures transitional patterns better than the 2D reshape, which confuses spring with summer (16 vs. 14 misclassifications).

- **Summer:** The 2D-CNN again leads (F1-score: 0.88), leveraging the reshape to emphasize solar generation peaks.

- **Autumn:** Both CNNs perform well (F1-score: 0.90 for 1D-CNN, 0.89 for 2D-CNN), capturing wind generation patterns.

The 2D-CNN's training curves (Figure 4) show a steady loss decrease but fluctuating validation accuracy (0.80–0.85), suggesting potential overfitting. The 1D-CNN's balanced performance across seasons highlights the value of direct temporal processing, while the MLP's lack of temporal structure limits its accuracy. Future improvements could include early stopping or learning rate scheduling.

References

[1] Assignment 2 Instructions, April 2025.