

```

ann_params:
  input_dim: 10
  output_dim: 4
  hidden_dims: [16, 32, 32, 16]          # 降低网络复杂度, 减少参数数量
  dropout: 0.05                          # 增加 Dropout 防止过拟合
  learning_rate: 0.55e-3                 # 降低学习率, 减少过拟合
  epochs: 6000                           # 适当增加训练轮数
  batch_size: 200                        # 增大 batch_size, 提高稳定性
  weight_decay: 3e-4                     # 增强 L2 正则
  checkpoint_path: "./models/ANN/best_ann.pt"
  early_stopping: true                   # 启用早停
  patience: 200                          # 允许更多的耐心等待
  optimizer: "AdamW"                    # 使用 AdamW, 改善正则化
  activation: "leakyrelu"                # 使用 ReLU, leakyrelu, tanh, sigmoid
  random_seed: 777

```

train={'MSE': 60.36608822386789, 'MAE': 5.319451937882833, 'R2': 0.8063889737019316}

val={'MSE': 91.8336134255949, 'MAE': 6.741102436230641, 'R2': 0.7137809741147367}

```

ann_params:
  input_dim: 10
  output_dim: 4
  hidden_dims: [16, 32, 64, 32, 16]      # 降低网络复杂度, 减少参数数量
  dropout: 0.05                          # 增加 Dropout 防止过拟合
  learning_rate: 0.55e-3                 # 降低学习率, 减少过拟合
  epochs: 6000                           # 适当增加训练轮数
  batch_size: 200                        # 增大 batch_size, 提高稳定性
  weight_decay: 3e-4                     # 增强 L2 正则
  checkpoint_path: "./models/ANN/best_ann.pt"
  early_stopping: true                   # 启用早停
  patience: 200                          # 允许更多的耐心等待
  optimizer: "AdamW"                    # 使用 AdamW, 改善正则化
  activation: "leakyrelu"                # 使用 ReLU, leakyrelu, tanh, sigmoid
  random_seed: 777

```

train={'MSE': 66.11442821067068, 'MAE': 5.566619210276454, 'R2': 0.7691222251567145}

val={'MSE': 91.34977696010675, 'MAE': 6.875681021967495, 'R2': 0.7269002345832805}

```

ann_params:
  input_dim: 10
  output_dim: 4
  hidden_dims: [16, 32, 64, 32, 16]      # 降低网络复杂度, 减少参数数量
  dropout: 0.05                          # 增加 Dropout 防止过拟合
  learning_rate: 0.1e-3                  # 降低学习率, 减少过拟合
  epochs: 6000                           # 适当增加训练轮数
  batch_size: 200                        # 增大 batch_size, 提高稳定性
  weight_decay: 3e-4                     # 增强 L2 正则

```

```

checkpoint_path: "./models/ANN/best_ann.pt"
early_stopping: true          # 启用早停
patience: 200                # 允许更多的耐心等待
optimizer: "AdamW"           # 使用 AdamW, 改善正则化
activation: "leakyrelu"       # 使用 ReLU, leakyrelu, tanh, sigmoid
random_seed: 777

```

```
train={'MSE': 65.03397266457536, 'MAE': 5.548815660554142, 'R2': 0.7797340718923341}
```

```
val={'MSE': 93.86720643961843, 'MAE': 6.843583805669387, 'R2': 0.7252401860074781}
```

```

ann_params:
    input_dim: 10
    output_dim: 4
    hidden_dims: [16, 32, 64, 32, 16]      # 降低网络复杂度, 减少参数数量
    dropout: 0.05                          # 增加 Dropout 防止过拟合
    learning_rate: 1e-3                    # 降低学习率, 减少过拟合
    epochs: 6000                            # 适当增加训练轮数
    batch_size: 200                        # 增大 batch_size, 提高稳定性
    weight_decay: 3e-4                     # 增强 L2 正则
    checkpoint_path: "./models/ANN/best_ann.pt"
    early_stopping: true                   # 启用早停
    patience: 200                          # 允许更多的耐心等待
    optimizer: "AdamW"                     # 使用 AdamW, 改善正则化
    activation: "leakyrelu"                # 使用 ReLU, leakyrelu, tanh, sigmoid
    random_seed: 777

```

```
train={'MSE': 51.634220296616675, 'MAE': 4.9142836244546775, 'R2': 0.8360283017603021}
```

```
val={'MSE': 90.33419023507255, 'MAE': 6.735513810974202, 'R2': 0.7313401364319789}
```

```

# ----- ANN 参数 (优化泛化能力) -----
ann_params:
    input_dim: 10
    output_dim: 4
    hidden_dims: [16, 32, 64, 32, 16]      # 降低网络复杂度, 减少参数数量
    dropout: 0.01                          # 增加 Dropout 防止过拟合
    learning_rate: 1e-3                    # 降低学习率, 减少过拟合
    epochs: 6000                            # 适当增加训练轮数
    batch_size: 200                        # 增大 batch_size, 提高稳定性
    weight_decay: 3e-4                     # 增强 L2 正则
    checkpoint_path: "./models/ANN/best_ann.pt"
    early_stopping: true                   # 启用早停
    patience: 200                          # 允许更多的耐心等待
    optimizer: "AdamW"                     # 使用 AdamW, 改善正则化
    activation: "leakyrelu"                # 使用 ReLU, leakyrelu, tanh, sigmoid
    random_seed: 777

```

```
train={'MSE': 62.264453541073436, 'MAE': 5.312417629904133, 'R2': 0.7881050815508368}
```

val={'MSE': 81.83691805984037, 'MAE': 6.3448813198413445, 'R2': 0.7510512656843131}

-----**Best**

```
# ----- ANN 参数 (优化泛化能力) -----
ann_params:
    input_dim: 10
    output_dim: 4
    hidden_dims: [16, 32, 32, 16]          # 降低网络复杂度, 减少参数数量
    dropout: 0.01                          # 增加 Dropout 防止过拟合
    learning_rate: 1e-3                    # 降低学习率, 减少过拟合
    epochs: 6000                           # 适当增加训练轮数
    batch_size: 200                        # 增大 batch_size, 提高稳定性
    weight_decay: 3e-4                     # 增强 L2 正则
    checkpoint_path: "./models/ANN/best_ann.pt"
    early_stopping: true                   # 启用早停
    patience: 200                          # 允许更多的耐心等待
    optimizer: "AdamW"                    # 使用 AdamW, 改善正则化
    activation: "leakyrelu"               # 使用 ReLU, leakyrelu, tanh, sigmoid
    random_seed: 777
```

train={'MSE': 51.55462099857504, 'MAE': 4.829048871786391, 'R2': 0.8274579409559509}

val={'MSE': 82.15022948238376, 'MAE': 6.225170265532564, 'R2': 0.7450928155887689}