Tree Ensembles:

Bagging, Random Forests and Boosting

Probability and Statistics for Data Science

Carlos Fernandez-Granda

These slides are based on the book Probability and Statistics for Data Science by Carlos Fernandez-Granda, available for purchase here. A free preprint, videos, code, slides and solutions to exercises are available at https://www.ps4ds.net

Trees

Interpretable nonlinear models for regression and classification

Problem: Overfitting

Example

Response: Temperature in Manhattan (Kansas)

Features:

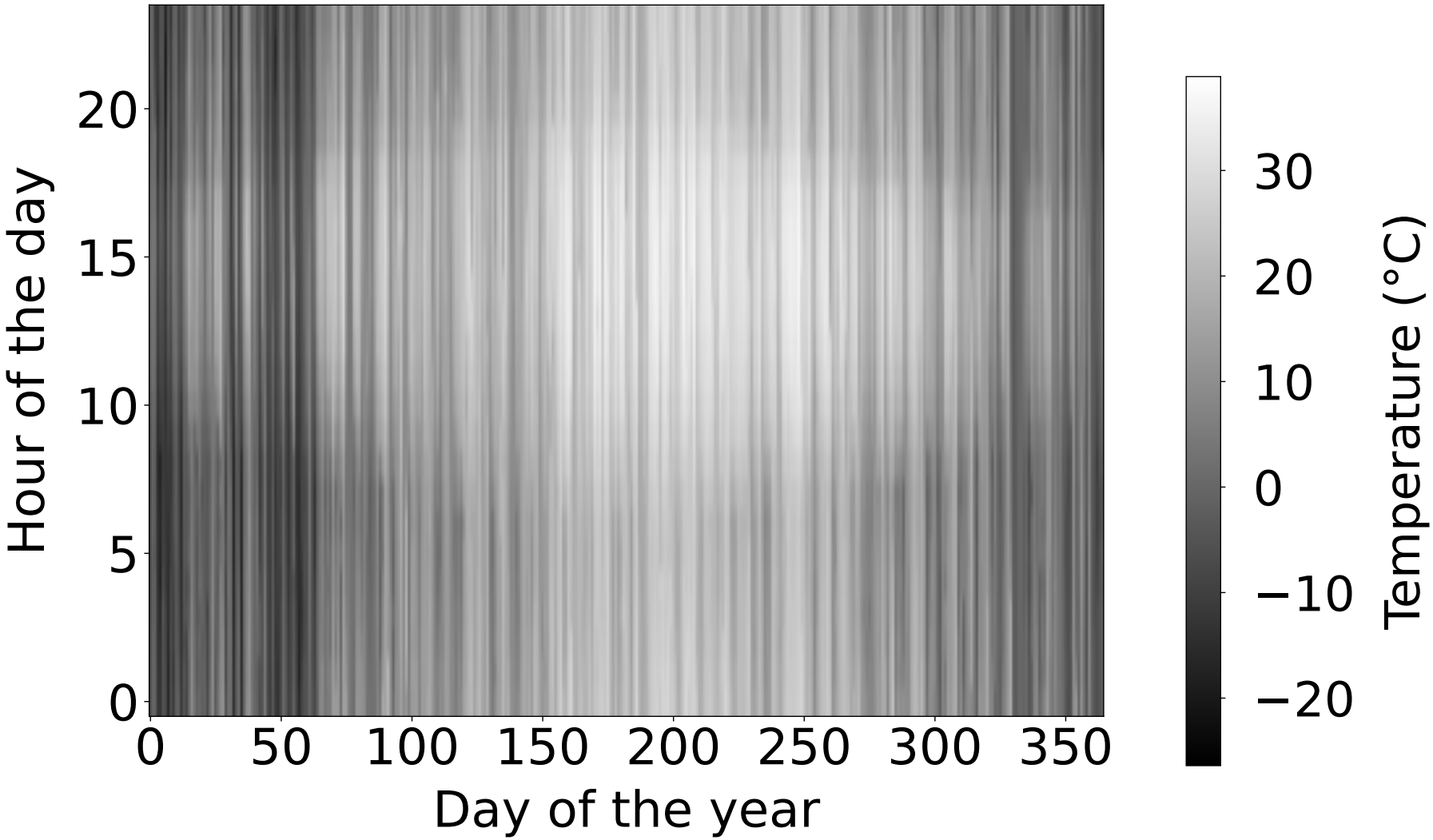
(1) Hour of the day (0-23)

(2) Day of the year (1-365)

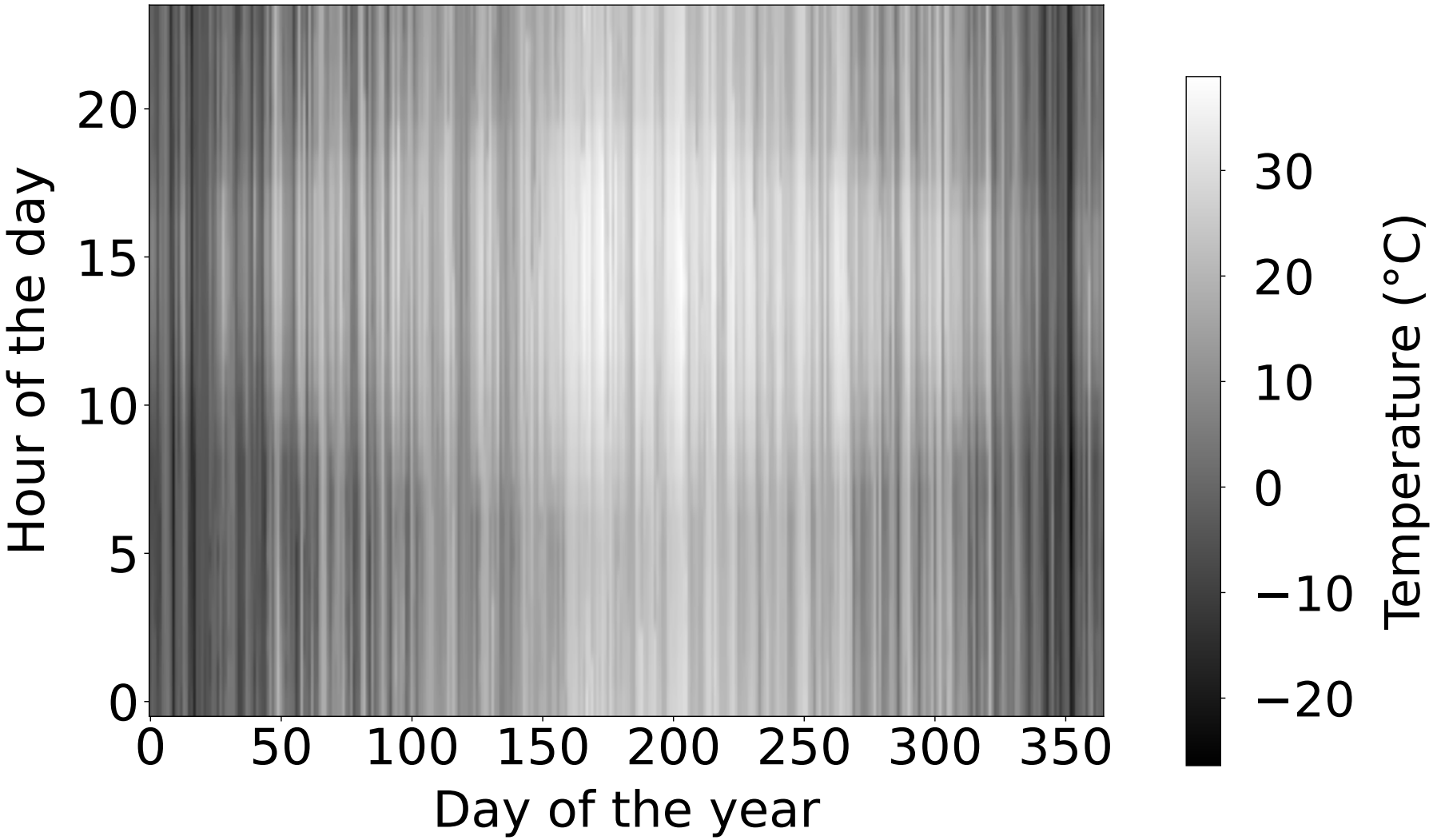
Training data: 2015

Test data: 2016

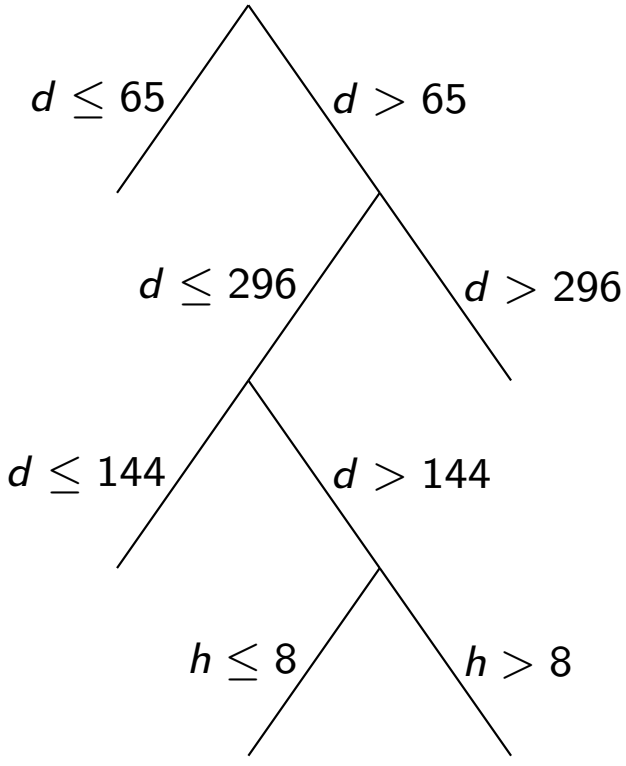
Training data

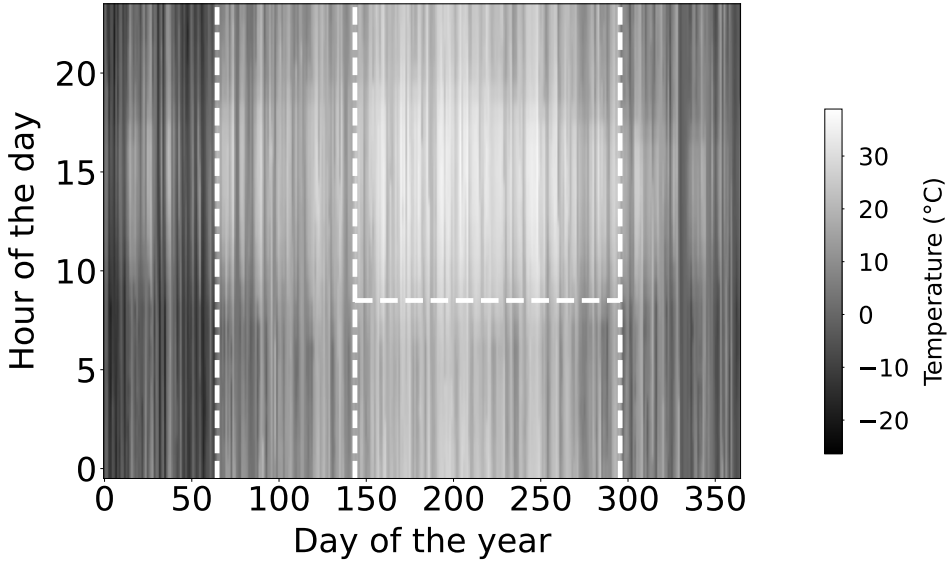


Test data

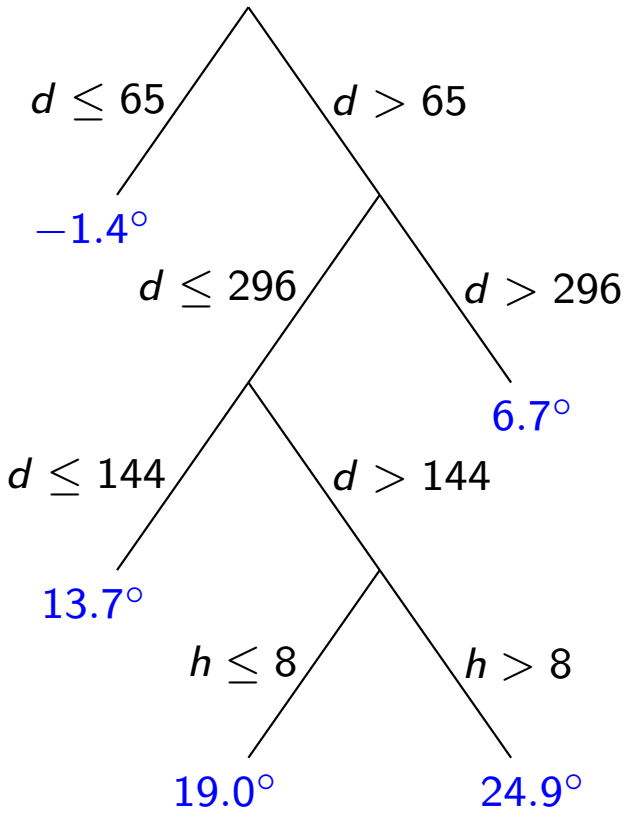

Hour of the day
0
5
10
15
20
0
50
100
150
200
250
300
350
Day of the year
30
20
10
0
−10
−20
Temperature (°C)

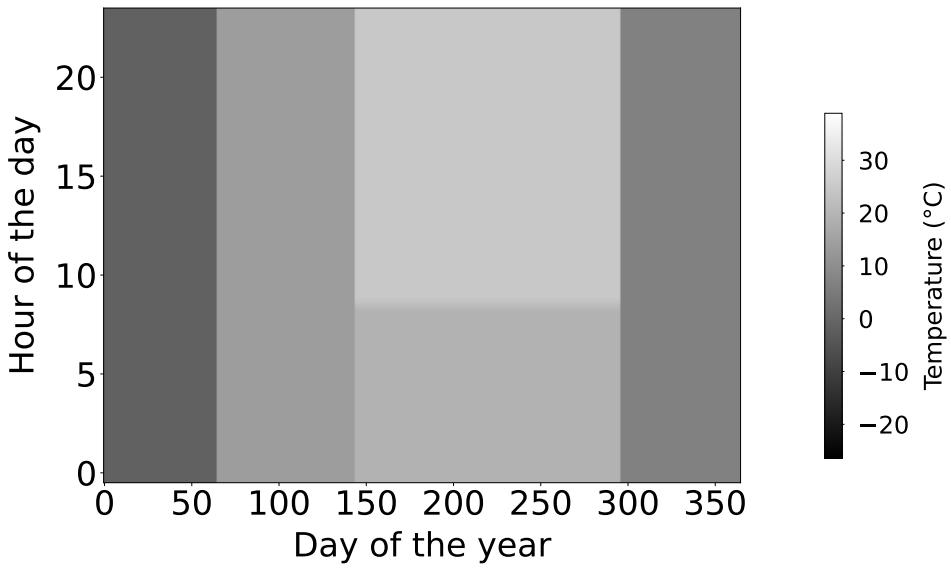
Regression tree



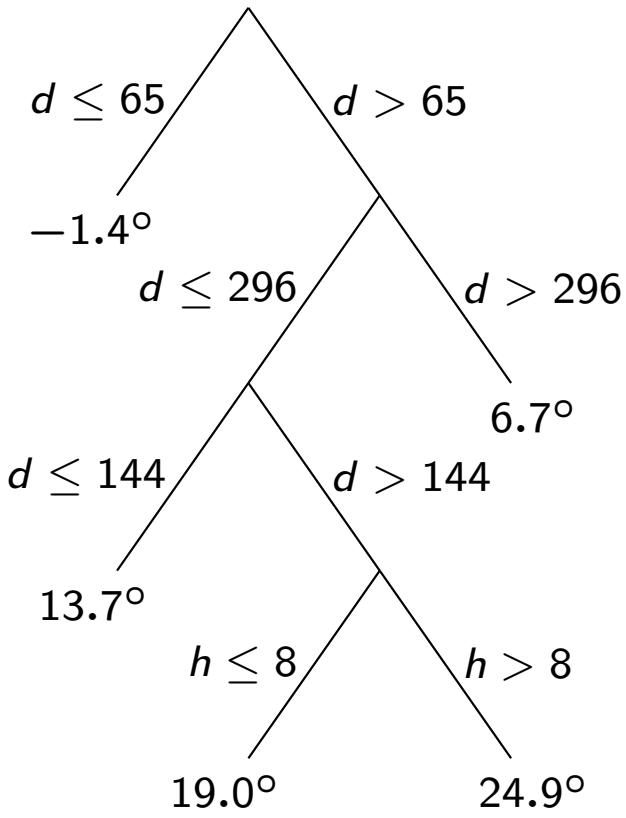


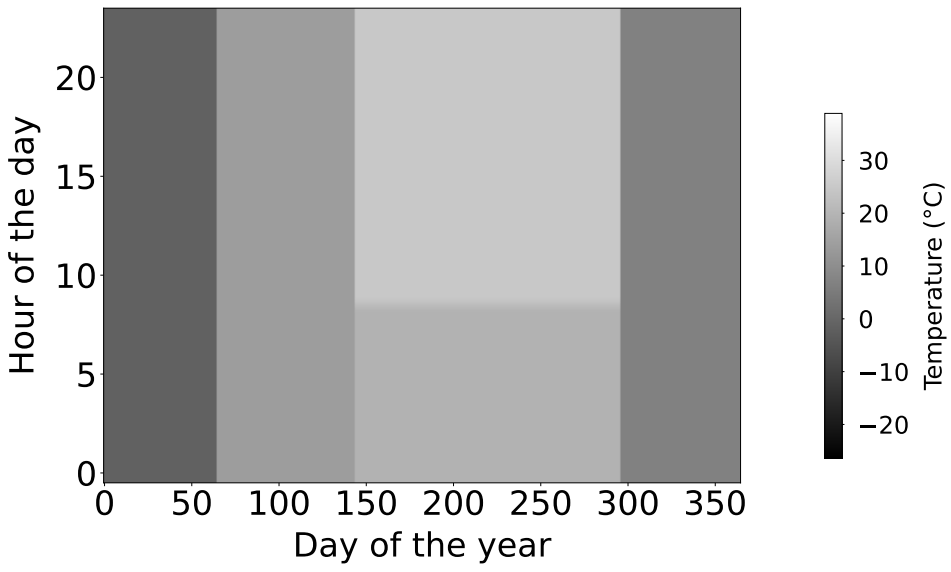
Regression tree



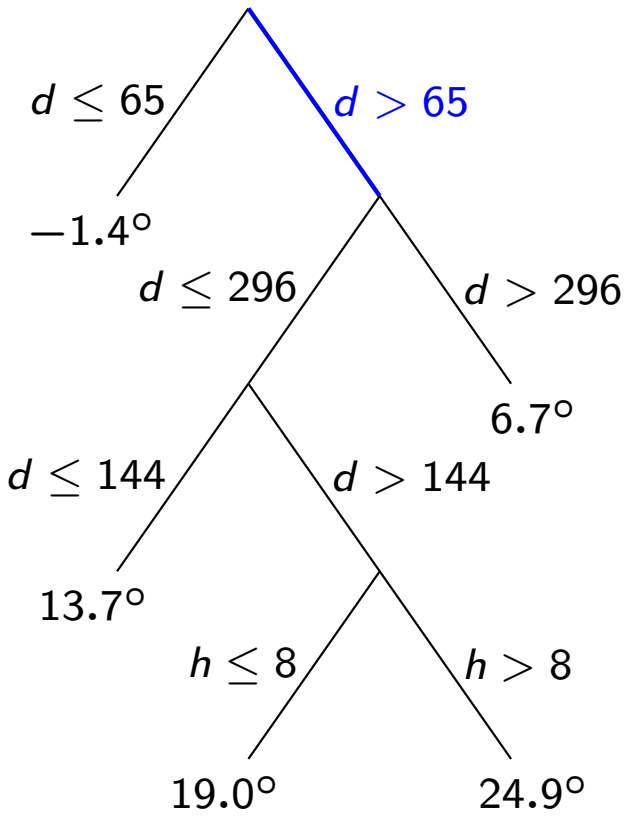


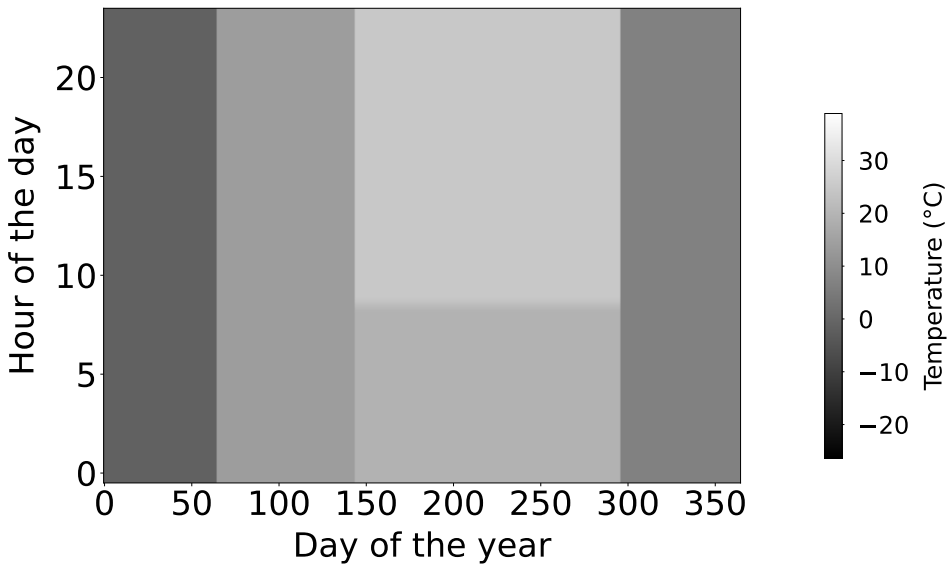
August 19 ($d := 251$) at 3 am ($h := 3$)?



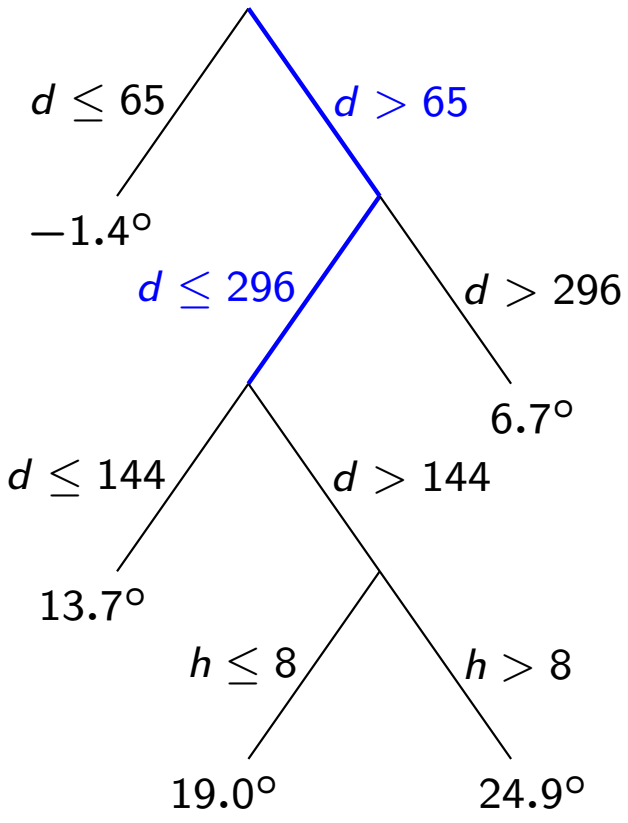


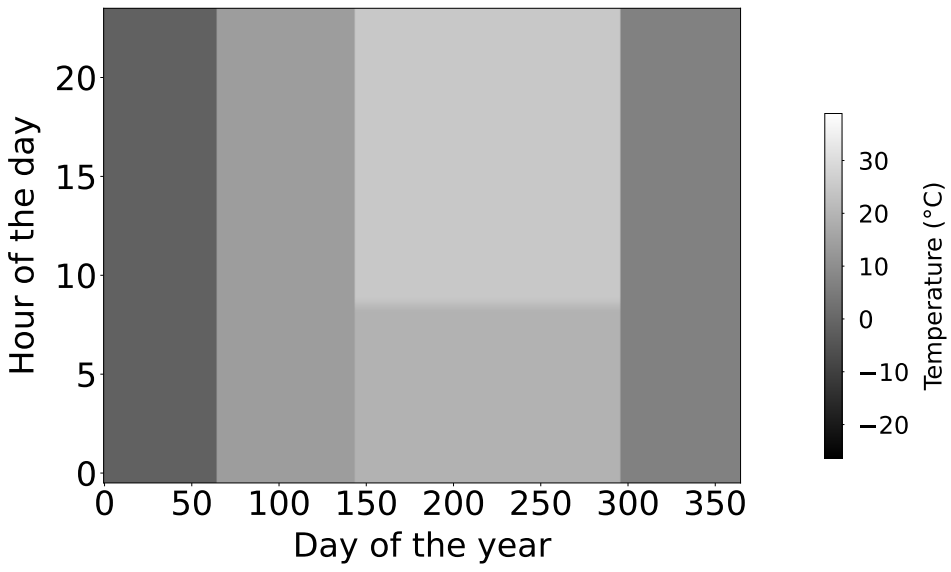
August 19 ($d := 251$) at 3 am ($h := 3$)?



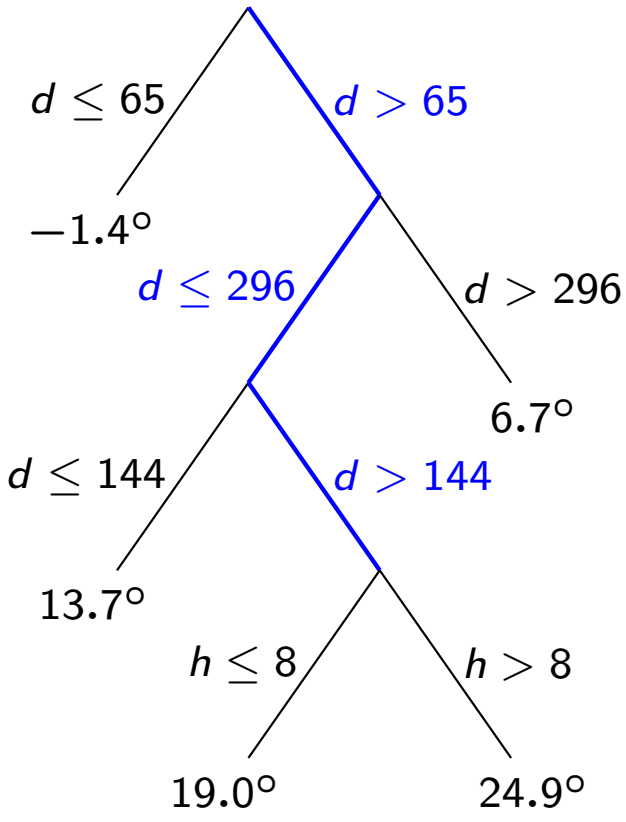


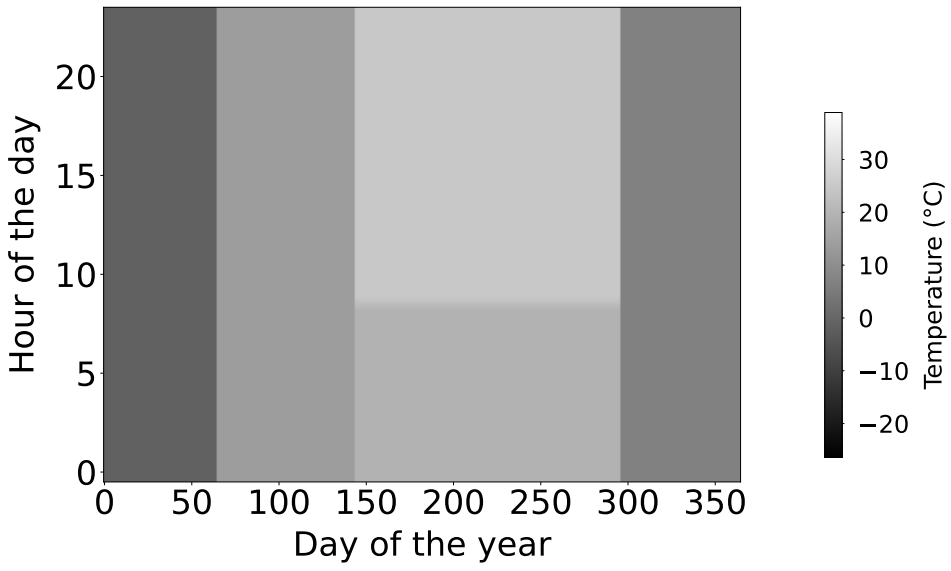
August 19 ($d := 251$) at 3 am ($h := 3$)?



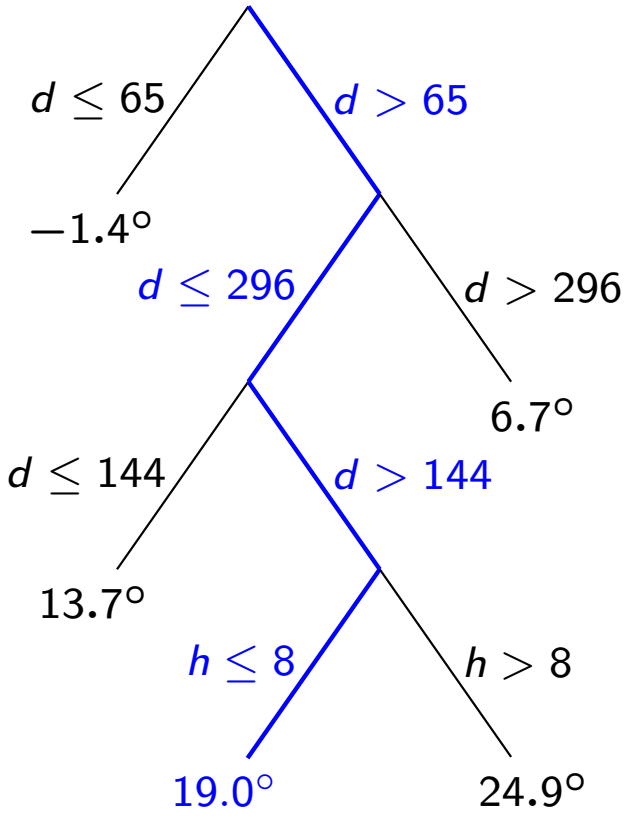


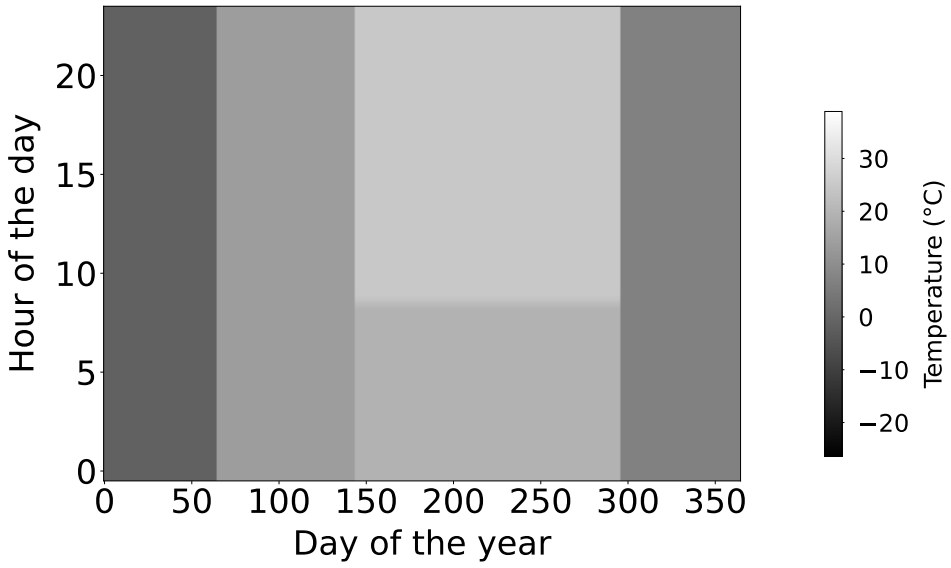
August 19 ($d := 251$) at 3 am ($h := 3$)?





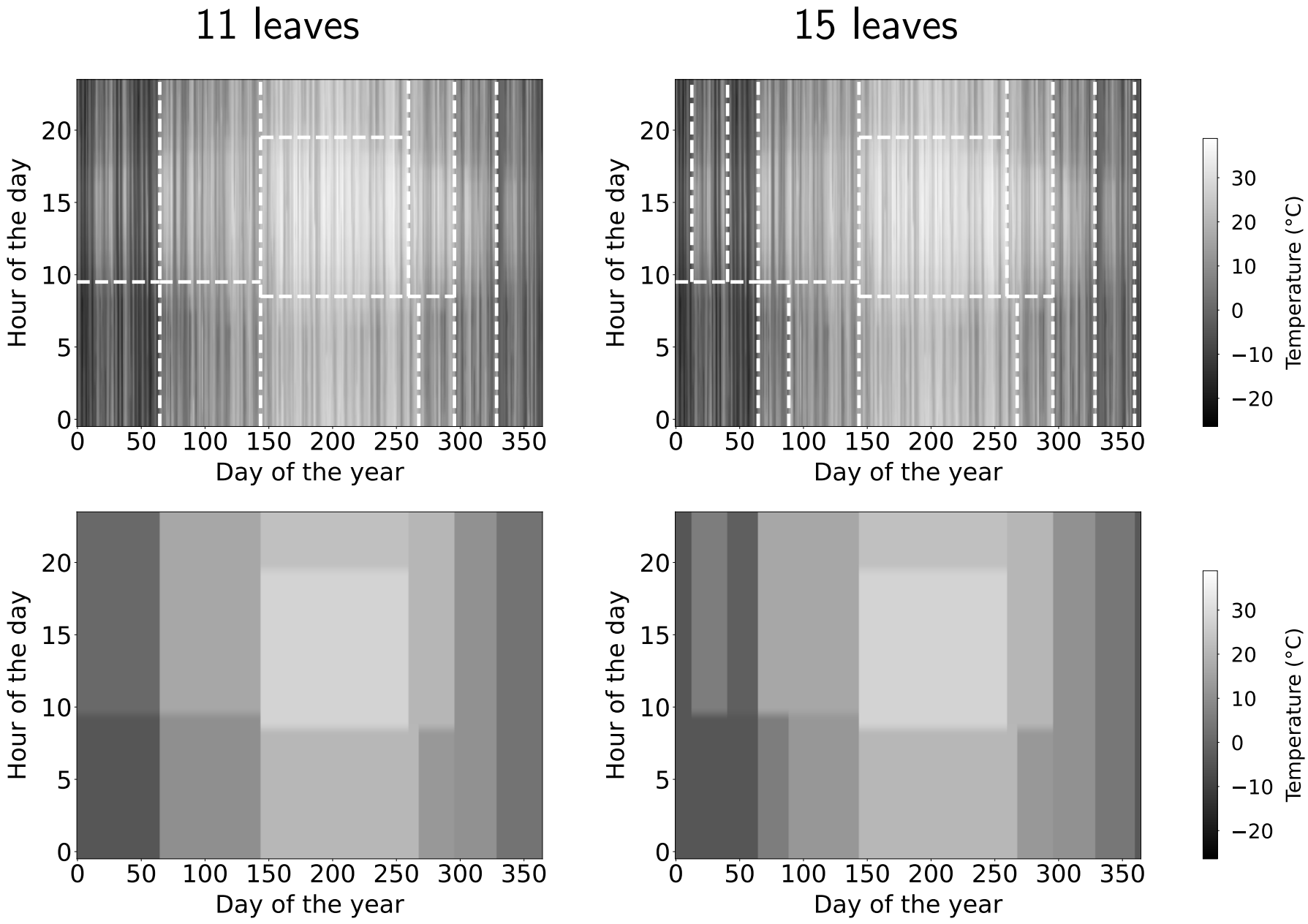
August 19 ($d := 251$) at 3 am ($h := 3$)?



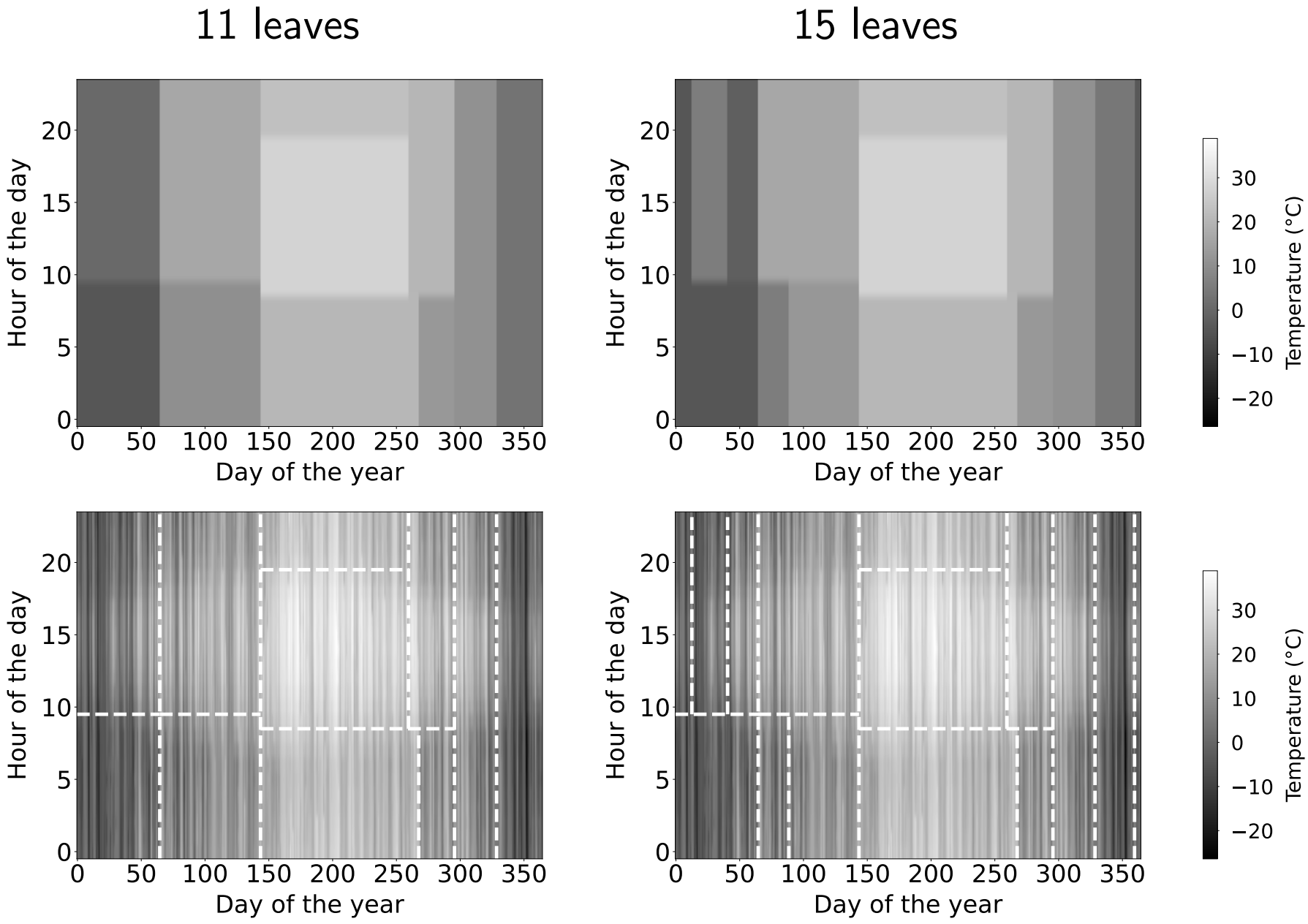


Interpretable!

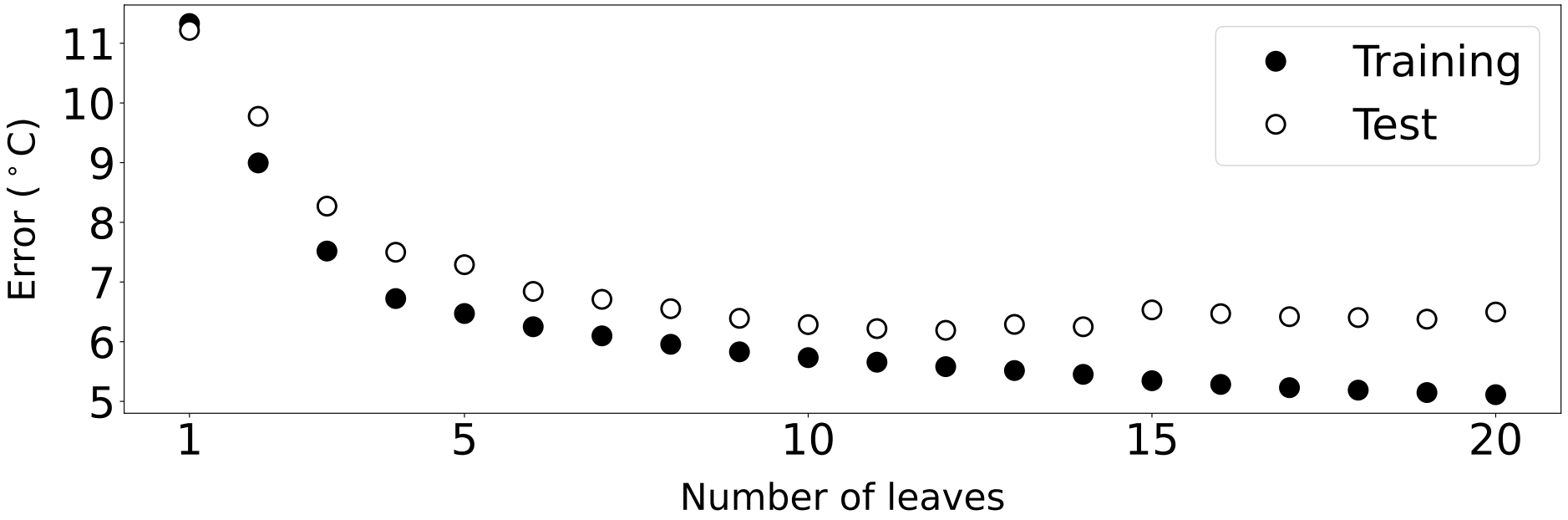
Fit to the training data



Generalization to the test data



Training and test error



Ensembles

Problem: Simple trees underfit / Complex trees overfit

Solution: Combine multiple simple trees

Three main strategies:

1. Bagging
2. Random forests
3. Boosting

Bagging

Motivation

Response: $\tilde{y}$

B different models: $\tilde{t}_b := \tilde{y} + \tilde{z}_b$, $1 \leq b \leq B$

Independent errors $\tilde{z}_1, \ldots, \tilde{z}_B$ with zero mean and variance σ^2

$$\text{MSE}_{\text{individual}} := \mathrm{E}\left[(\tilde{y} - \tilde{t}_b)^2\right] = \mathrm{E}\left[\tilde{z}_b^2\right] = \sigma^2$$

For an ensemble $\tilde{w} := \frac{1}{B}\sum_{b=1}^{B} \tilde{t}_b$

$$\text{MSE}_{\text{ensemble}} := \mathrm{E}\left[(\tilde{y} - \tilde{w})^2\right] = \mathrm{E}\left[\left(\frac{1}{B}\sum_{b=1}^{B} \tilde{z}_b\right)^2\right]$$

$$= \frac{1}{B^2}\mathrm{Var}\left[\sum_{b=1}^{B} \tilde{z}_b\right] = \frac{1}{B^2}\sum_{b=1}^{B} \mathrm{Var}\left[\tilde{z}_b\right] = \frac{\sigma^2}{B}$$

Bagging

Idea: Combine trees learned from different data

Problem: Where do we get different datasets?

The bootstrap

Resampling the available data!

Samples: $x_1, \ldots, x_n$

Bootstrap indices: $\tilde{k}_1, \tilde{k}_2, \ldots, \tilde{k}_n$

Sampled independently and uniformly with replacement

$$\mathrm{P}\left(\tilde{k}_j = i\right) = \frac{1}{n} \qquad 1 \leq i, j \leq n$$

Bootstrap samples: $\tilde{b}_1, \ldots, \tilde{b}_n$

$$\tilde{b}_j = x_{\tilde{k}_j} \qquad 1 \leq j \leq n$$

Bagging (bootstrap aggregating)

Idea: Fit trees to datasets produced via bootstrapping

Given training data (x_1, y_1), (x_2, y_2), ..., (x_n, y_n)

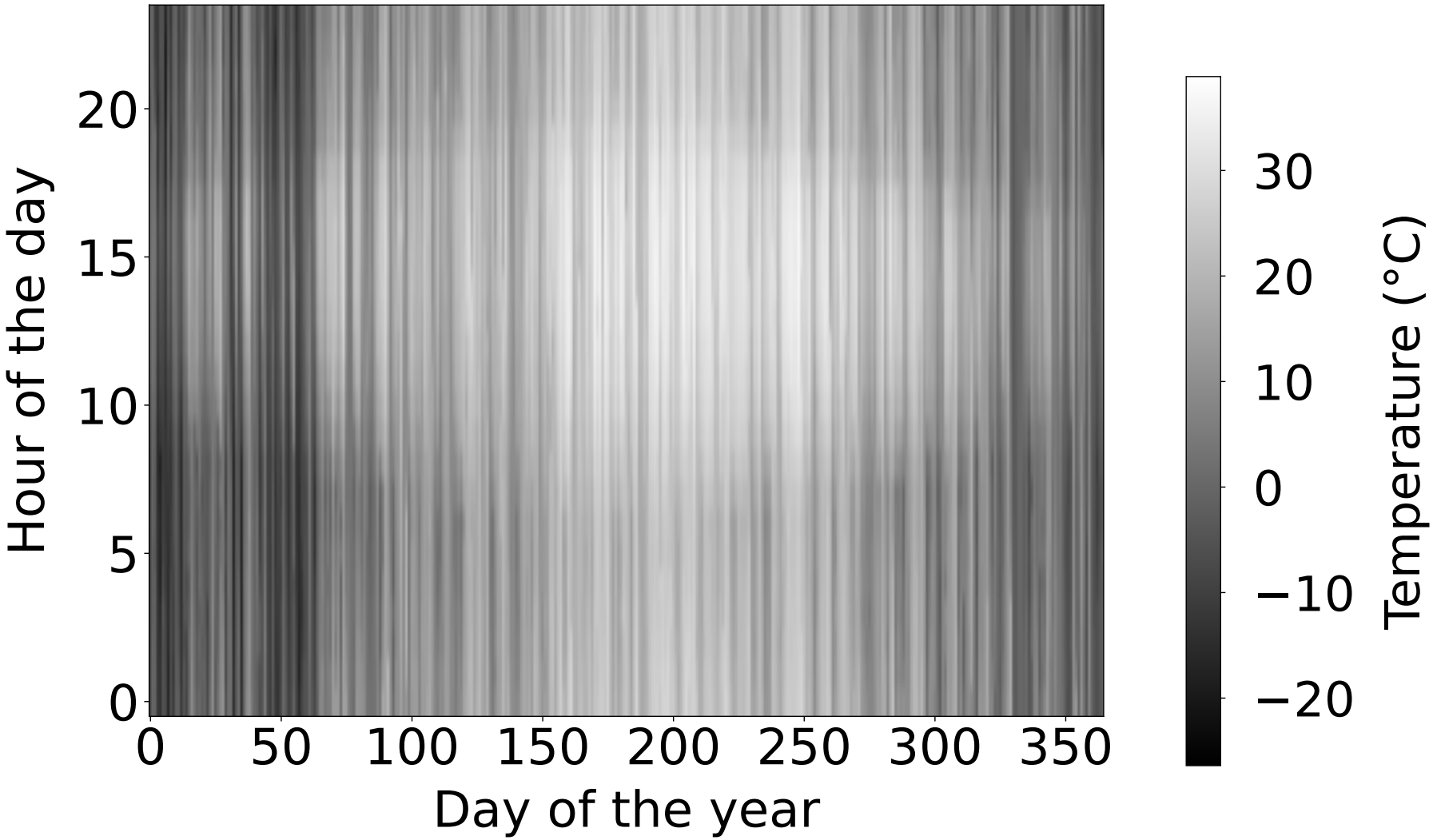
1. Bootstrapping: Resample n data points independently and uniformly B times

2. Fit a tree t_b to each bootstrap dataset for $1 \leq b \leq B$

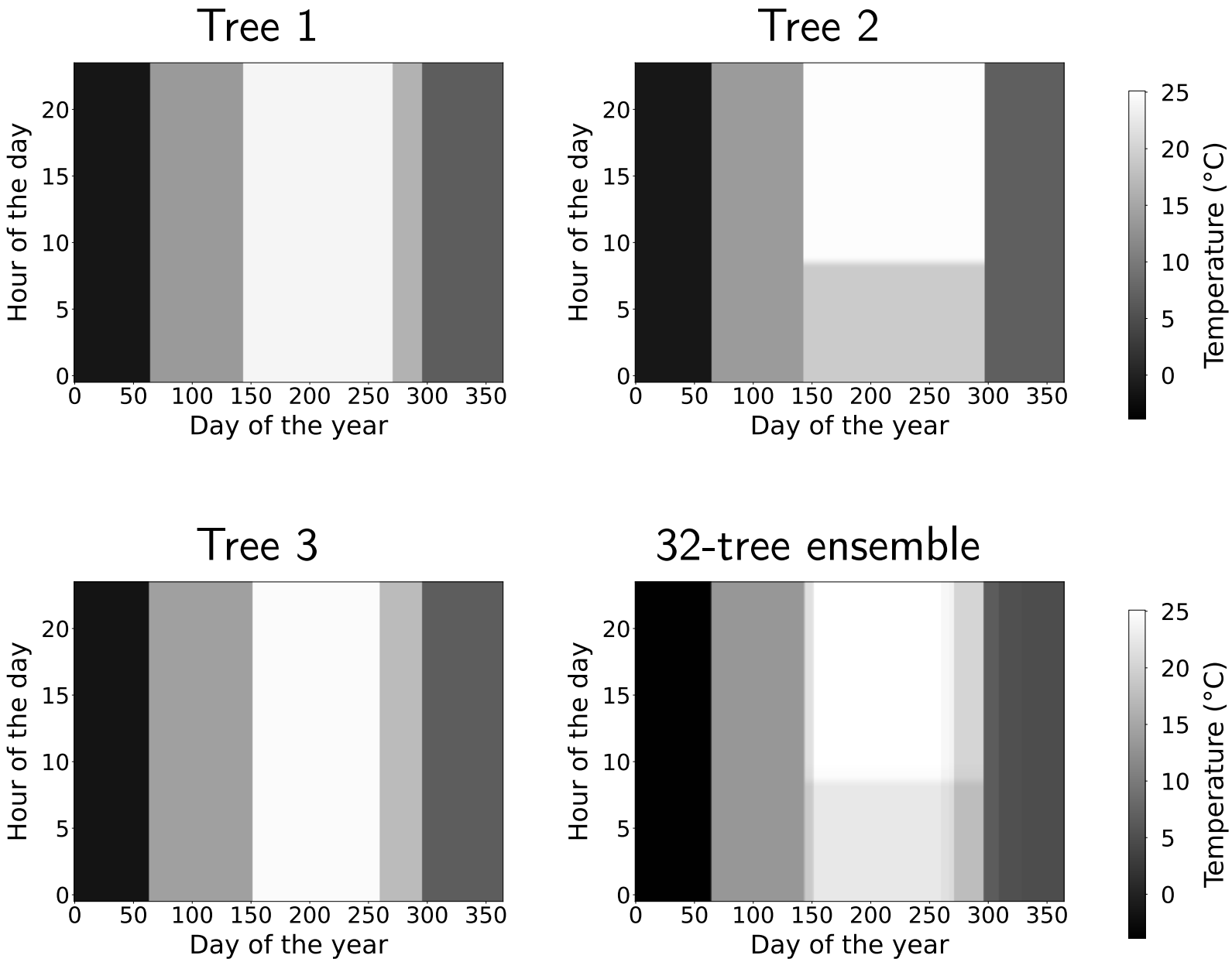
3. For new input feature vector, average output of the B trees

$$n\ell_{\text{bagging}}(x) := \frac{1}{B} \sum_{b=1}^{B} t_b(x)$$

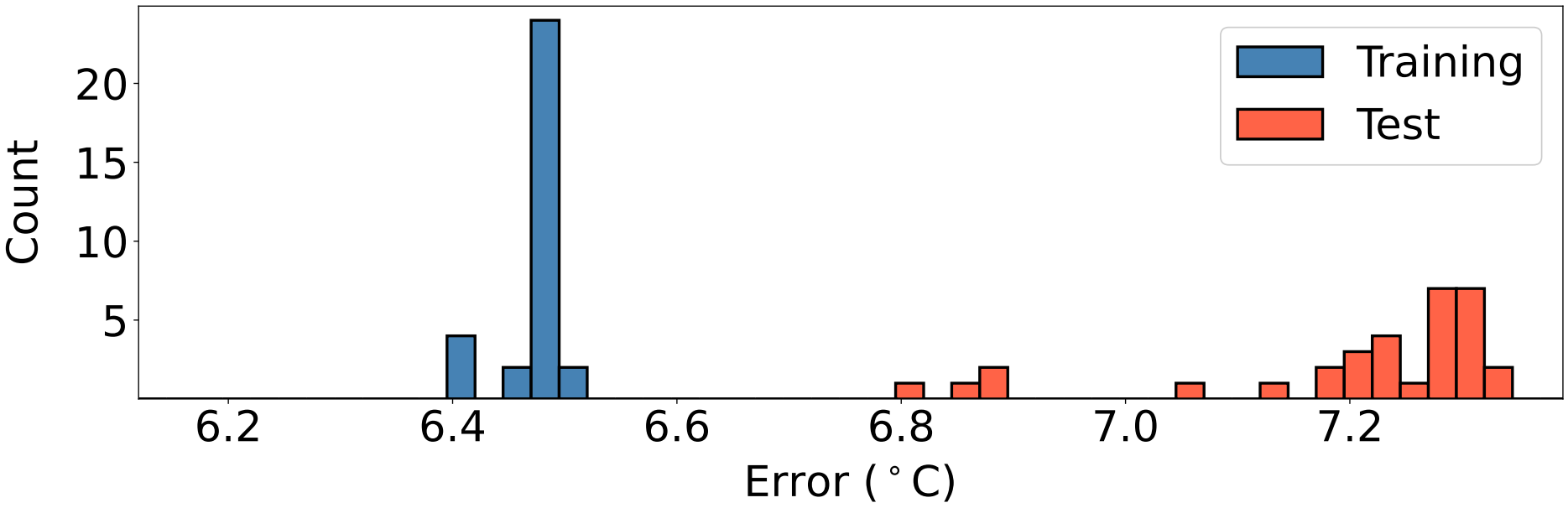
Training data



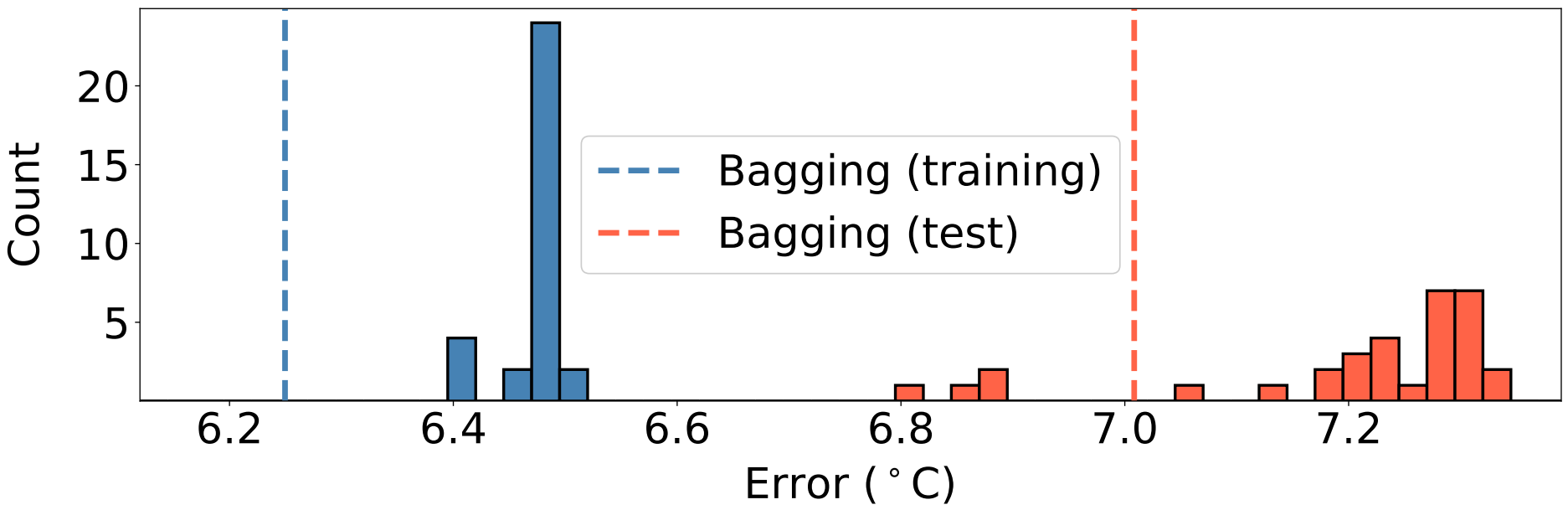
5-leaf trees


Tree 1
Tree 2
Tree 3
32-tree ensemble
Hour of the day
Day of the year
Temperature (°C)
0
5
10
15
20
0
50
100
150
200
250
300
350
25

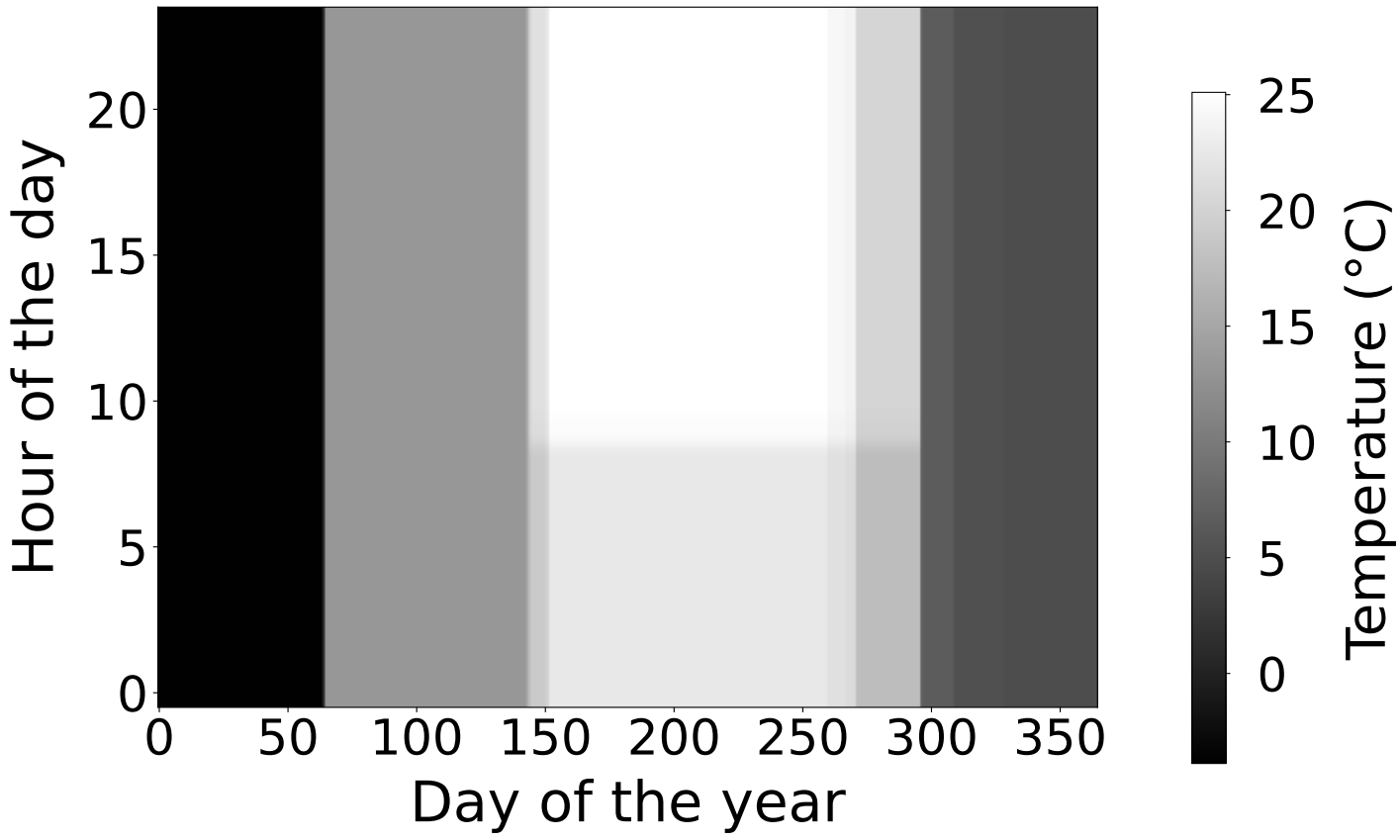
5-leaf trees



5-leaf trees


Count
20
15
10
5
Bagging (training)
Bagging (test)
6.2
6.4
6.6
6.8
7.0
7.2
Error (°C)

Less error, but...

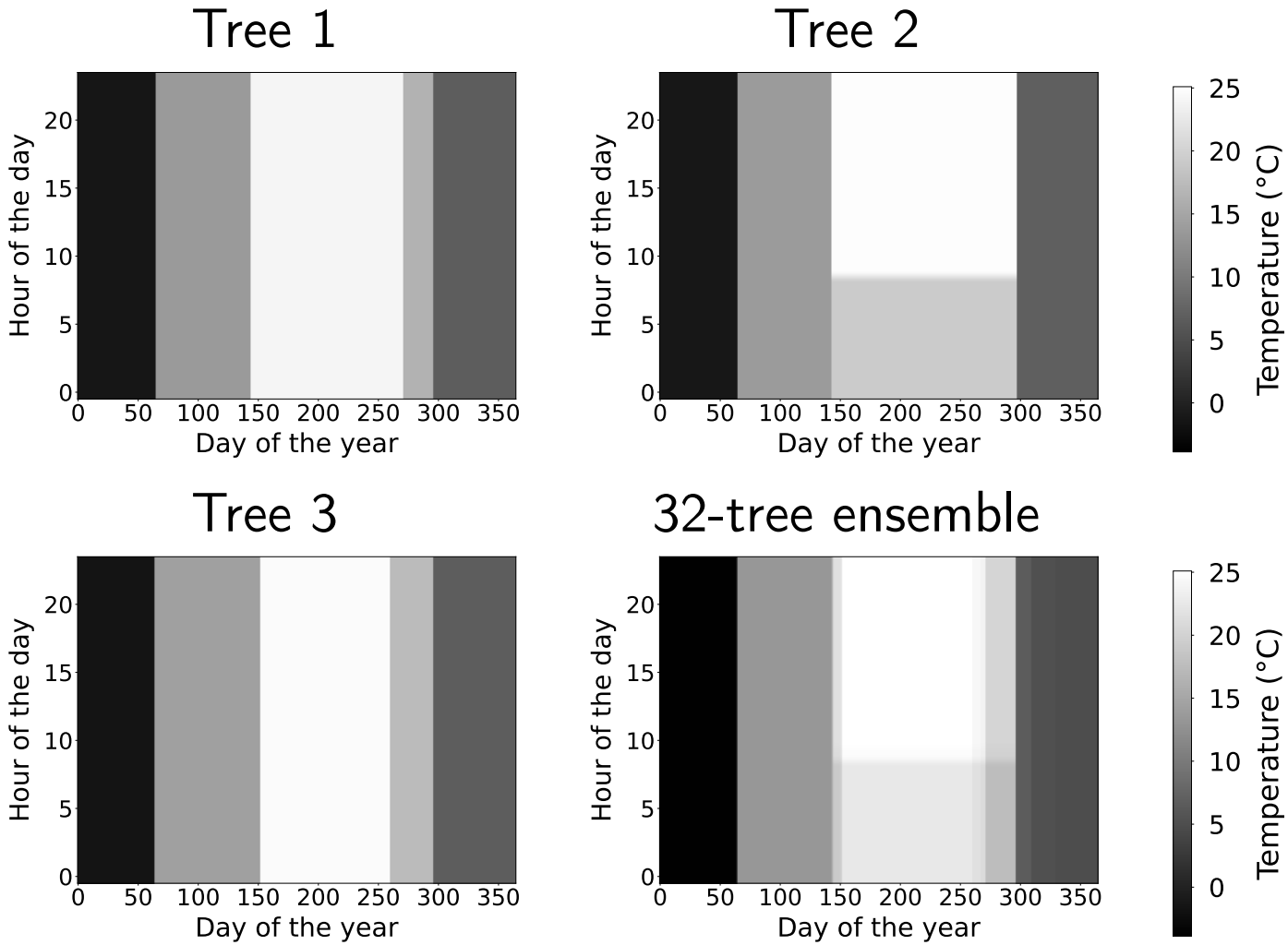


No longer interpretable!

Random Forests

Motivation

Bagging averages trees that are very similar



Idea: Modify tree construction to induce more variety

Random forests

Given training data (x_1, y_1), (x_2, y_2), ..., (x_n, y_n)

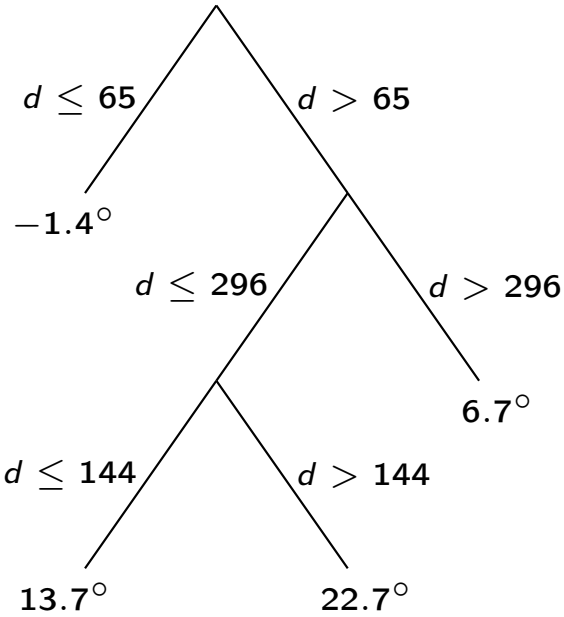
1. Bootstrapping: Resample n data points independently and uniformly B times

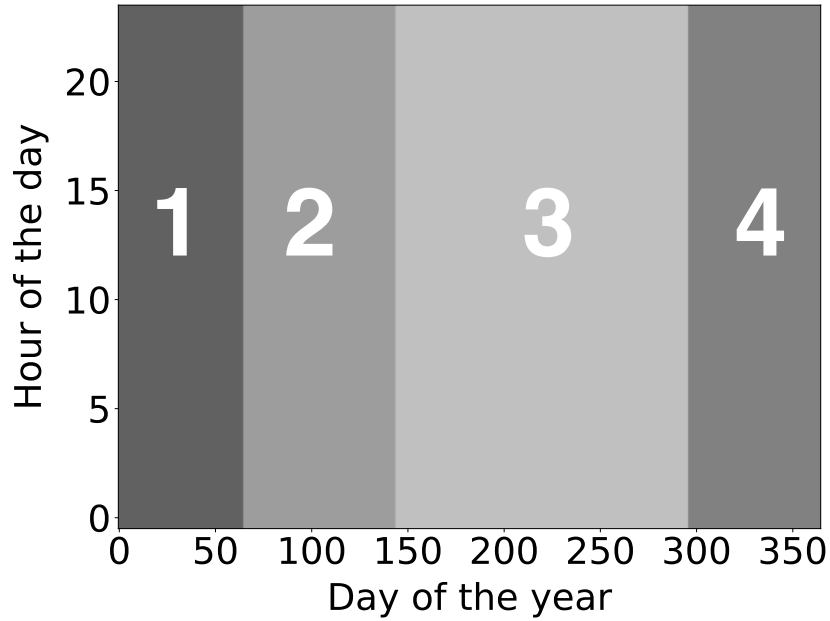
2. Fit a tree t_b to each bootstrap dataset for $1 \leq b \leq B$ via **randomized** recursive binary splitting

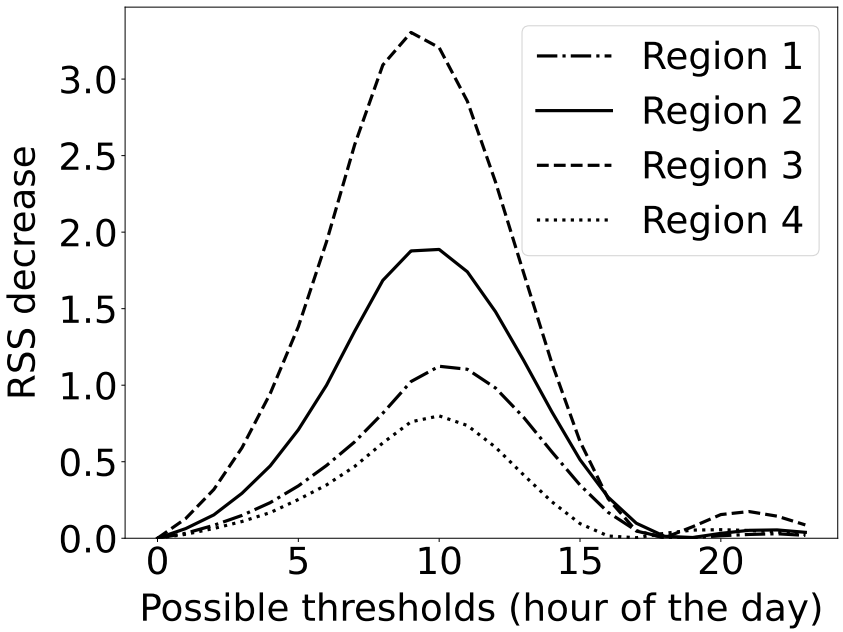
3. For new input feature vector, average output of the B trees

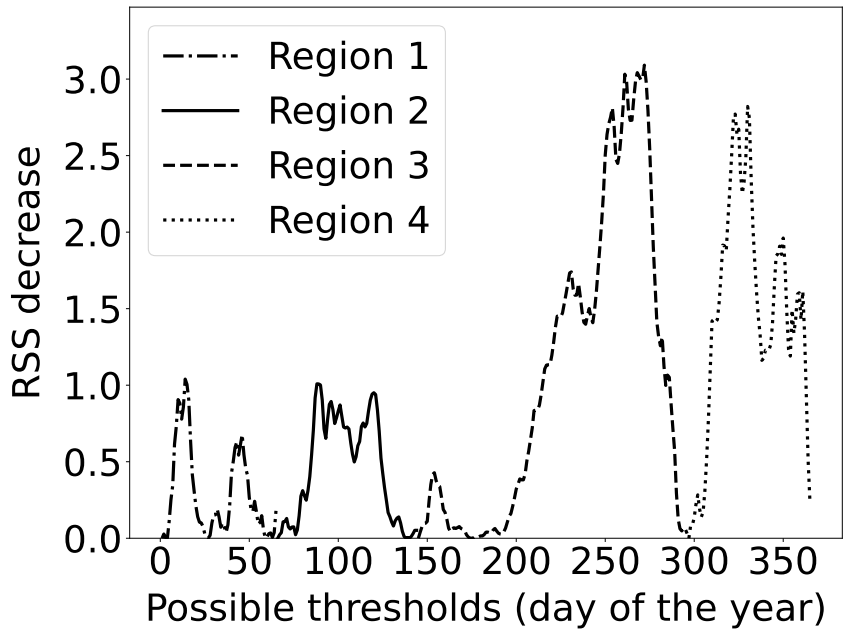
$$n\ell_{\mathsf{rf}}(x) := \frac{1}{B} \sum_{b=1}^{B} t_b(x)$$

Recursive binary splitting









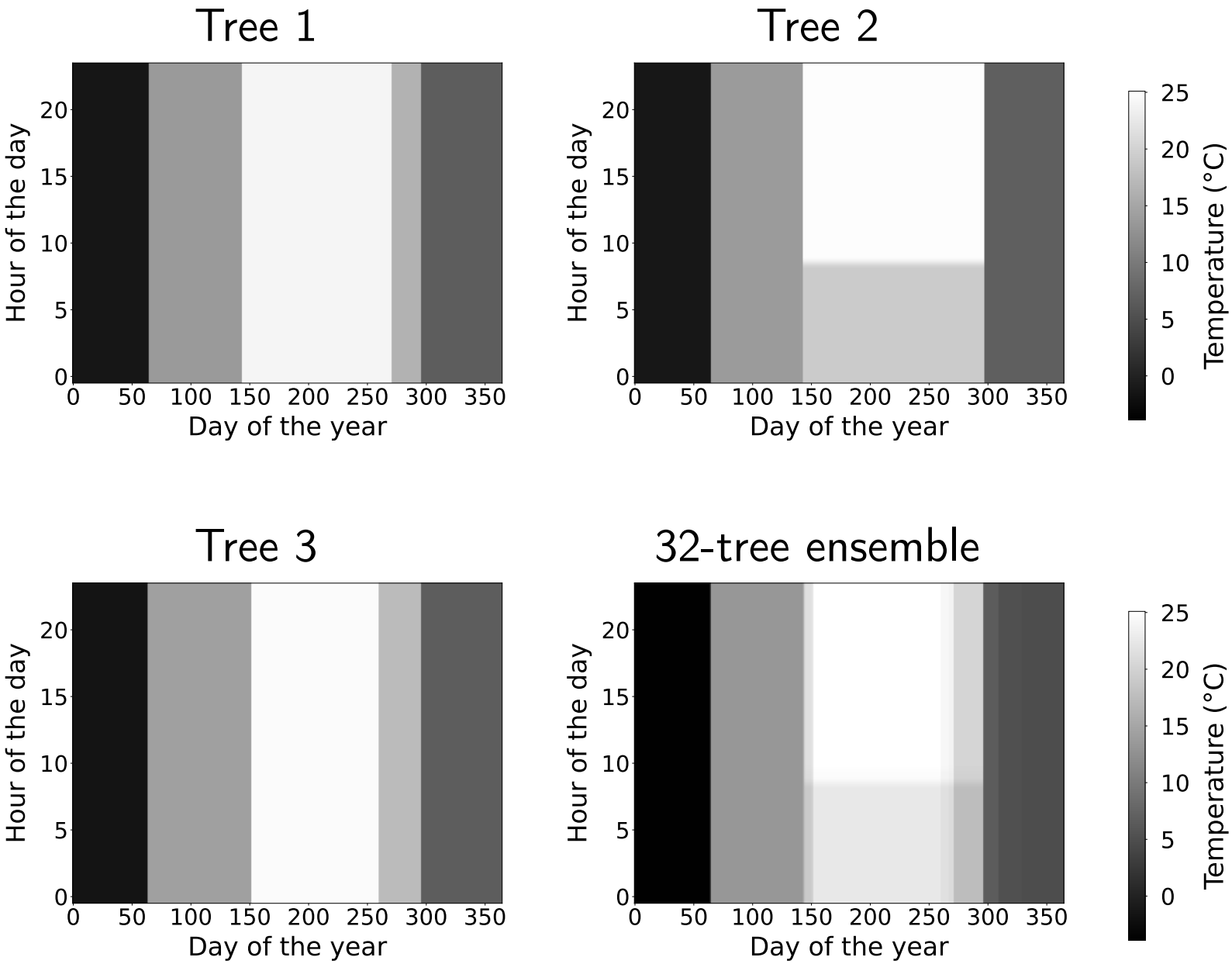
Randomized binary splitting

Only consider a random subset of features at each split

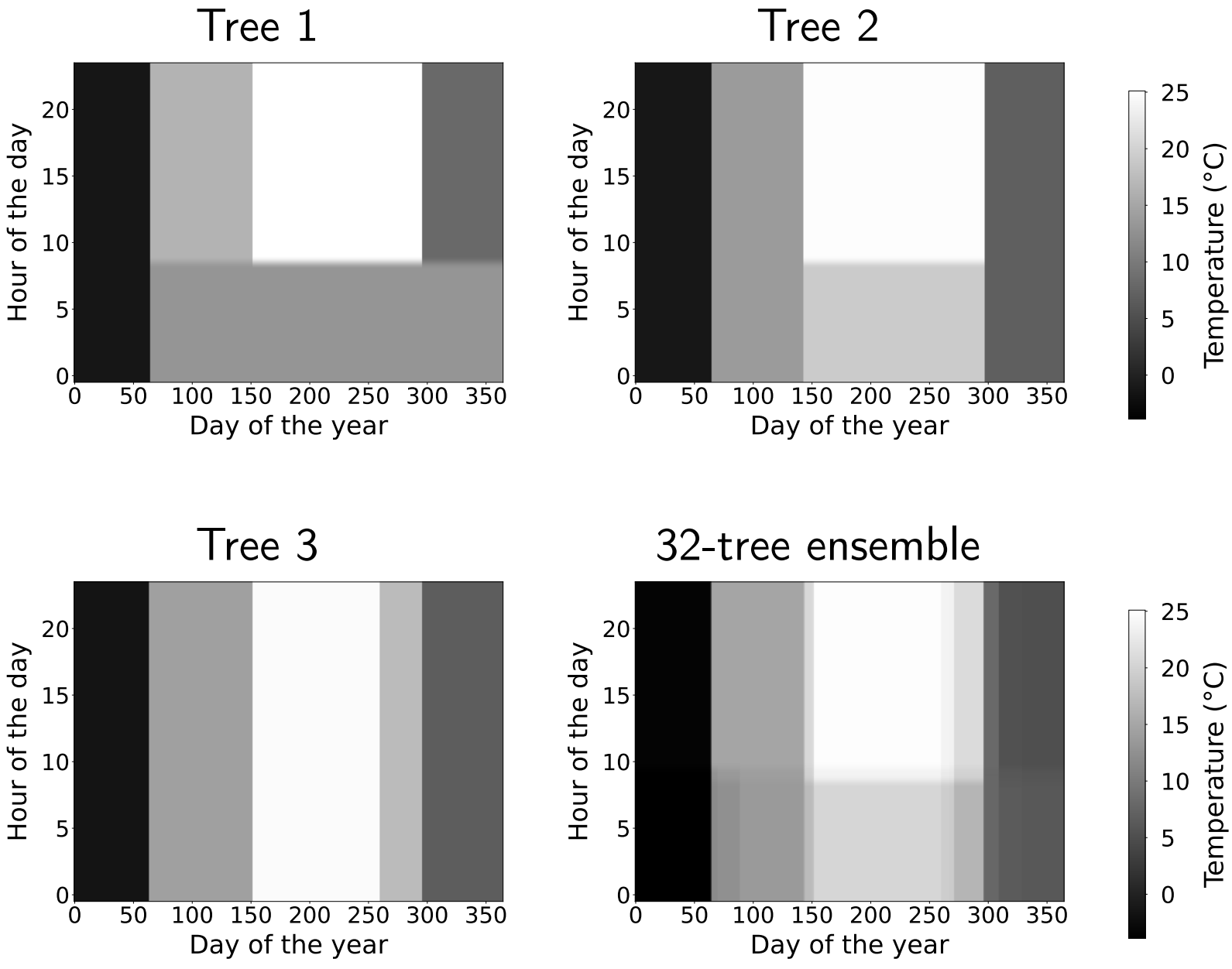
For our example, we choose a random feature 20% of the time

If there are d features, $\sqrt{d}$ is a popular choice

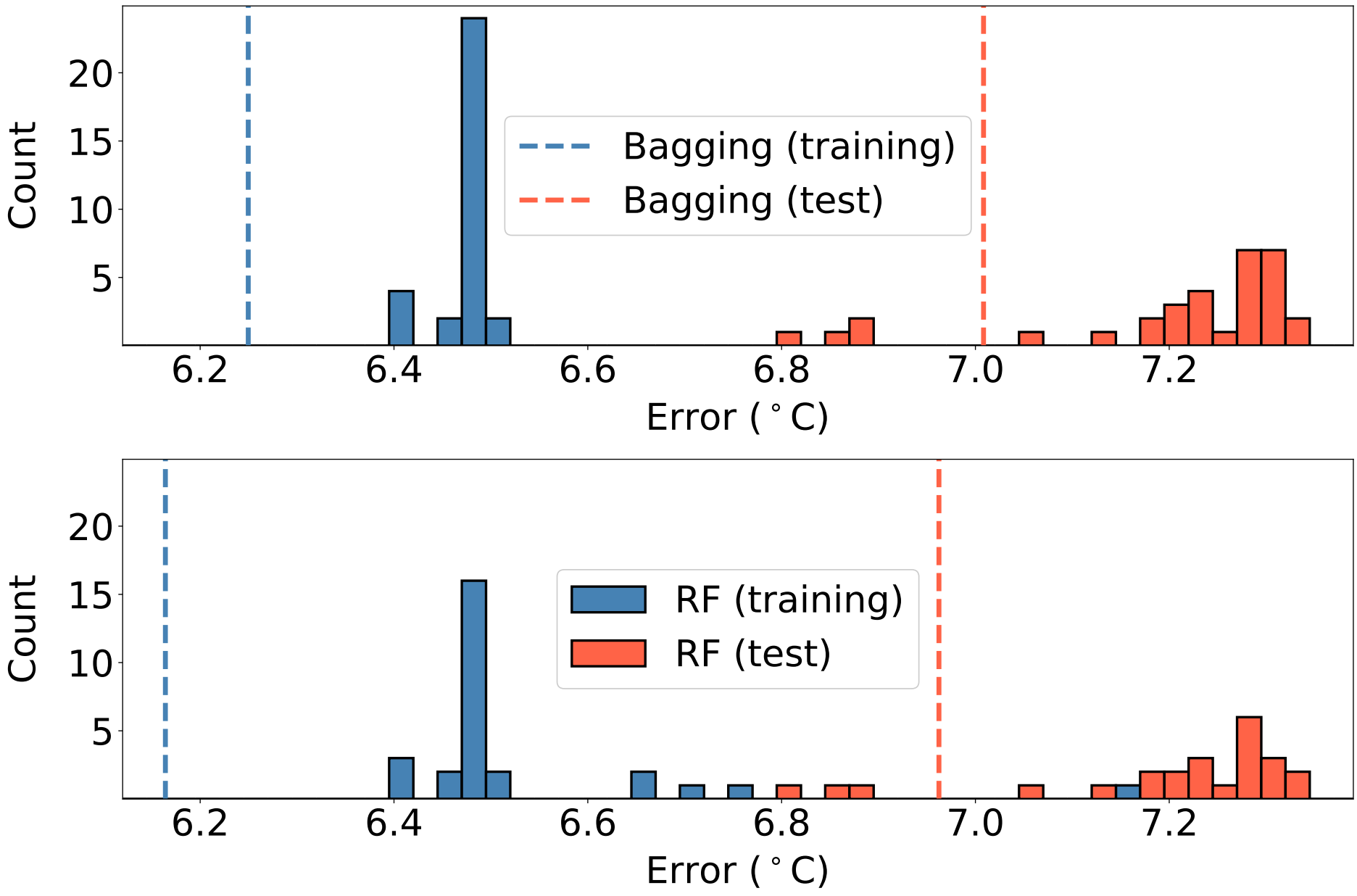
5-leaf trees: Bagging



5-leaf trees: Random forest



5-leaf trees


Count
20
15
10
5
Bagging (training)
Bagging (test)
6.2
6.4
6.6
6.8
7.0
7.2
Error (°C)
Count
20
15
10
5
RF (training)
RF (test)
6.2
6.4
6.6
6.8
7.0
7.2
Error (°C)

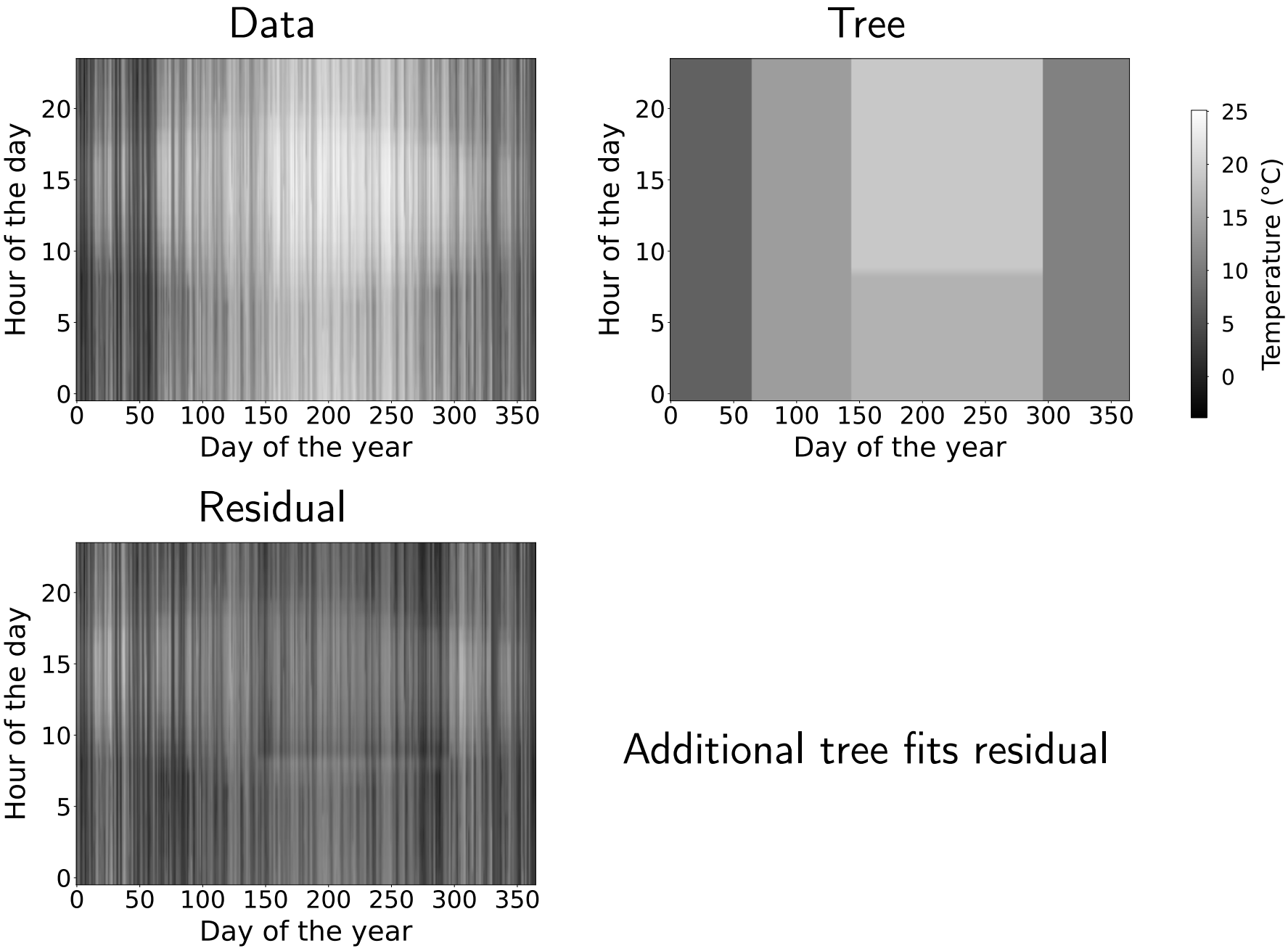
Boosting

Ensembling strategies

Bagging and random forests combine independent trees

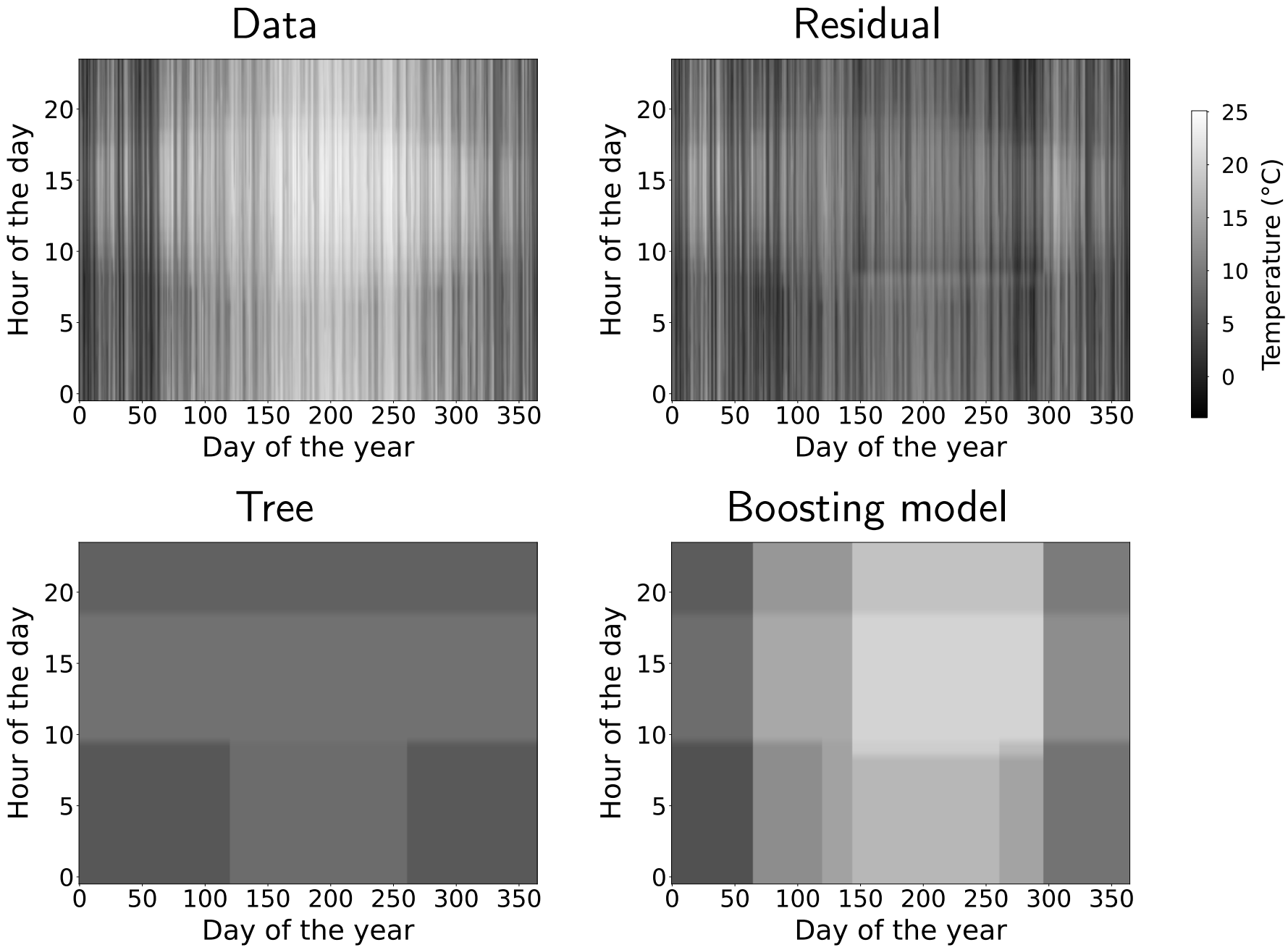
Boosting combines complementary trees

Naive boosting

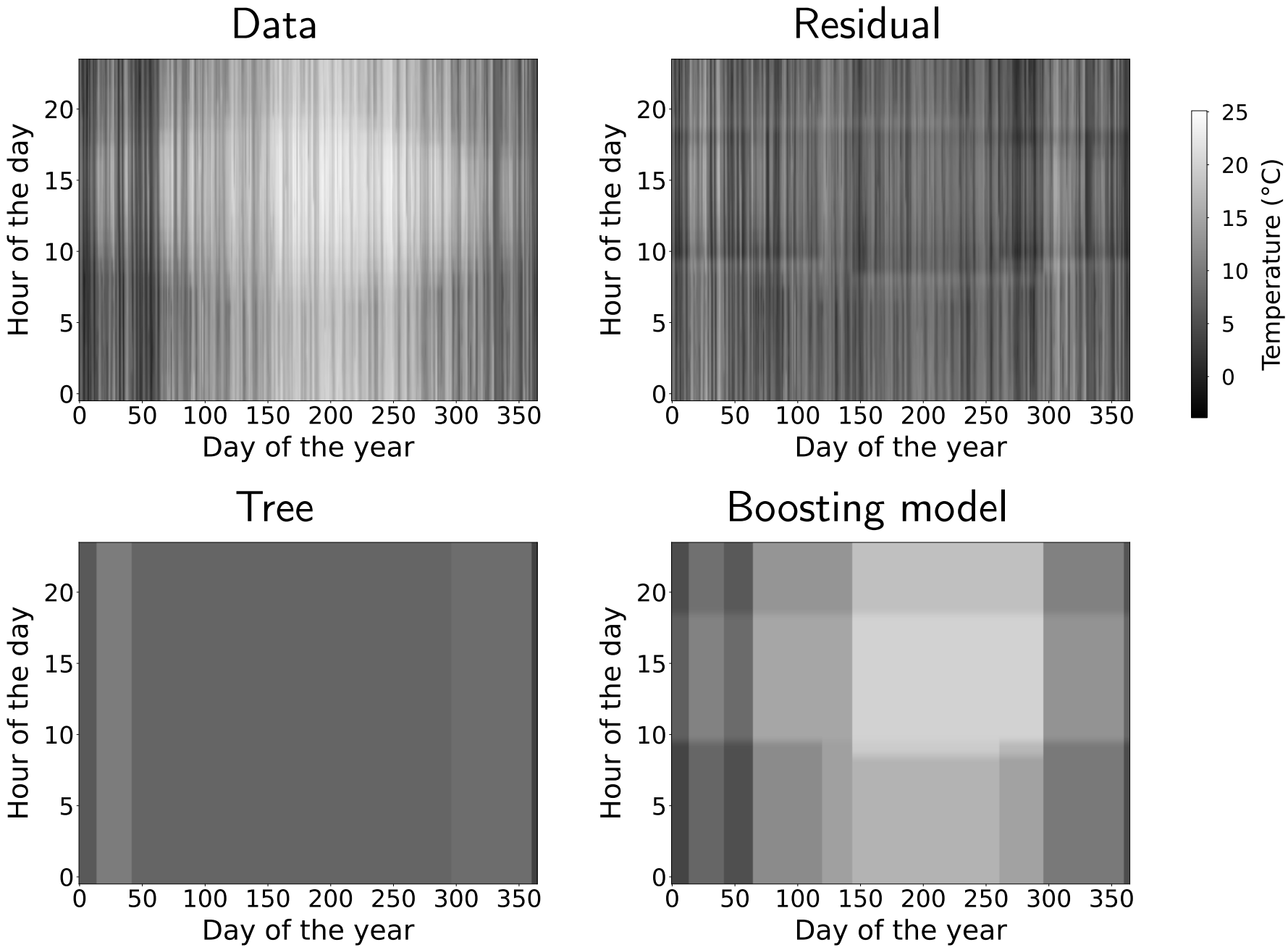


Additional tree fits residual

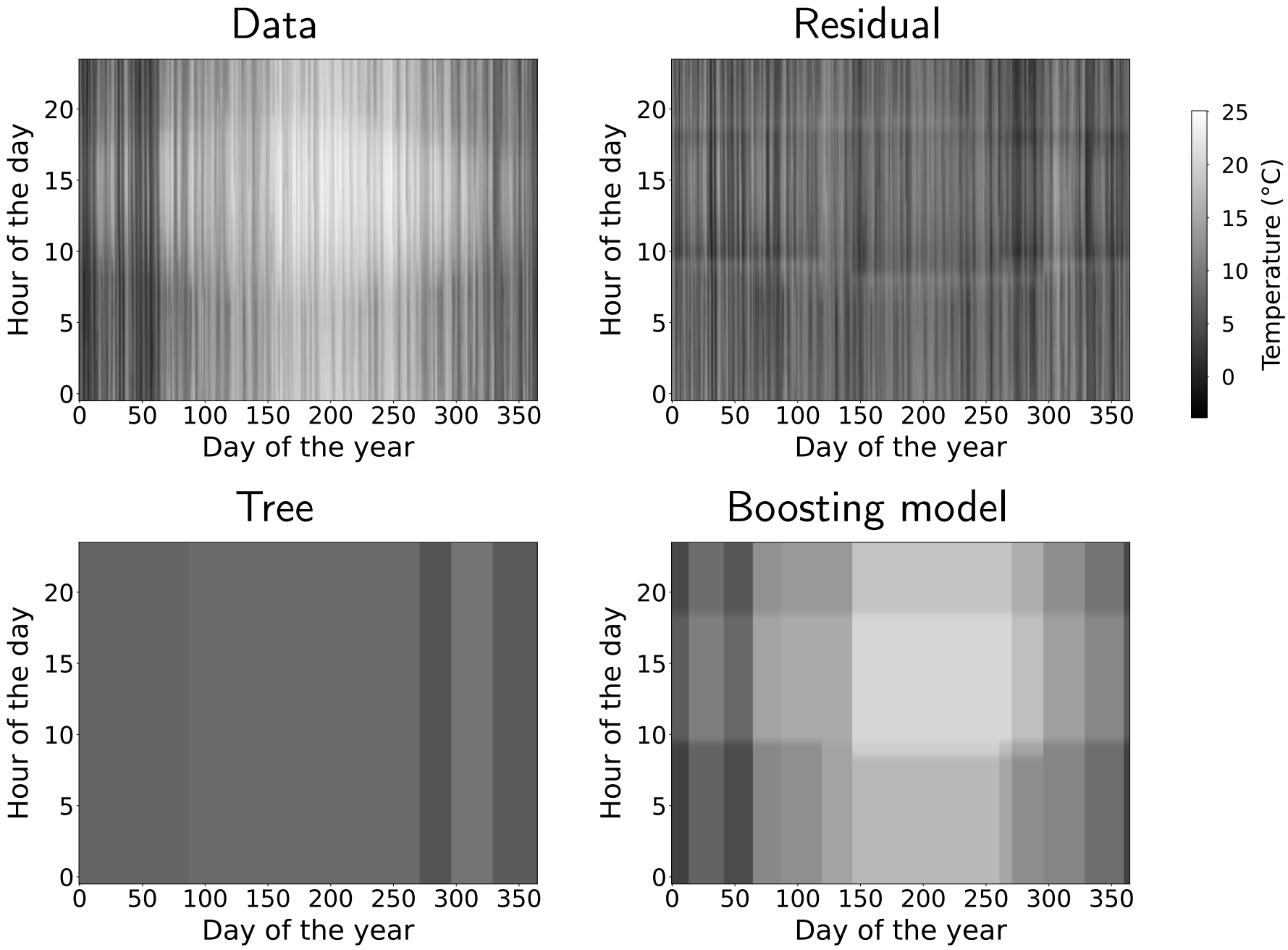
Tree 2


Data
Residual
Tree
Boosting model
Hour of the day
Day of the year
0
5
10
15
20
0
50
100
150
200
250
300
350
Temperature (°C)
25
20
15
10
5
0

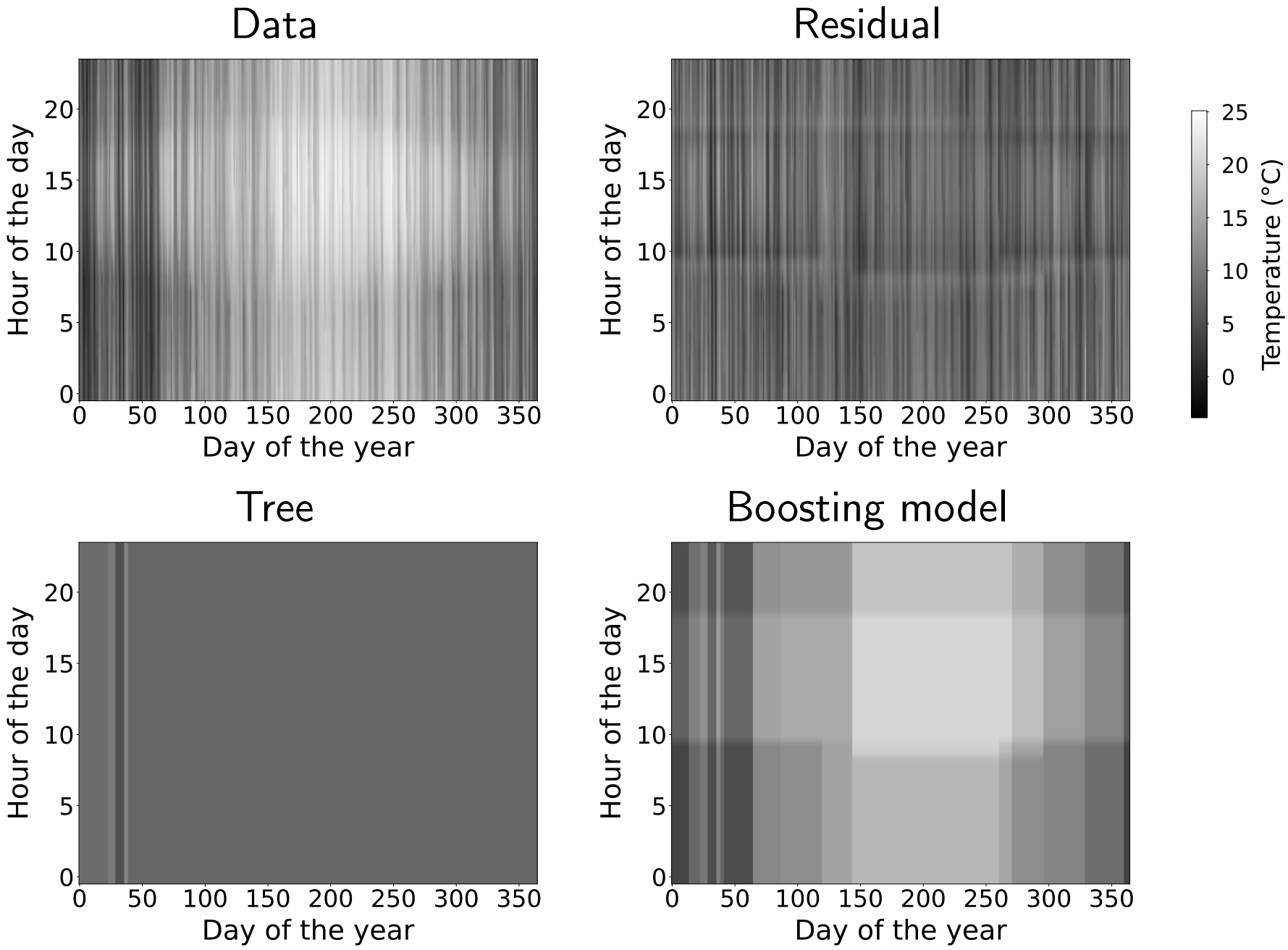
Tree 3


Data
Residual
Tree
Boosting model
Hour of the day
Day of the year
Temperature (°C)
0
5
10
15
20
25
0
50
100
150
200
250
300
350

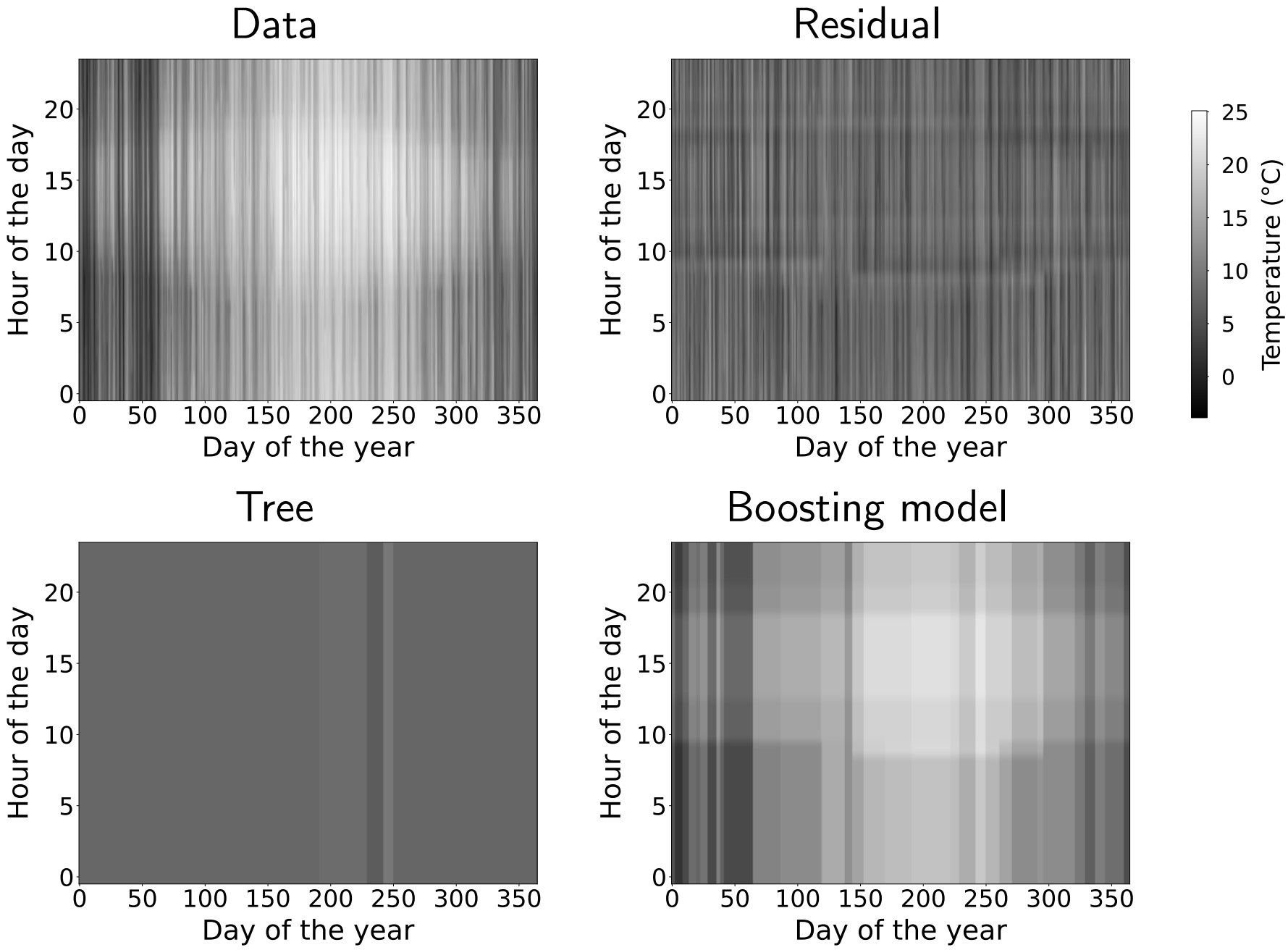
Tree 4


Data
Residual
Tree
Boosting model
Hour of the day
Day of the year
Temperature (°C)
0
5
10
15
20
25
0
50
100
150
200
250
300
350

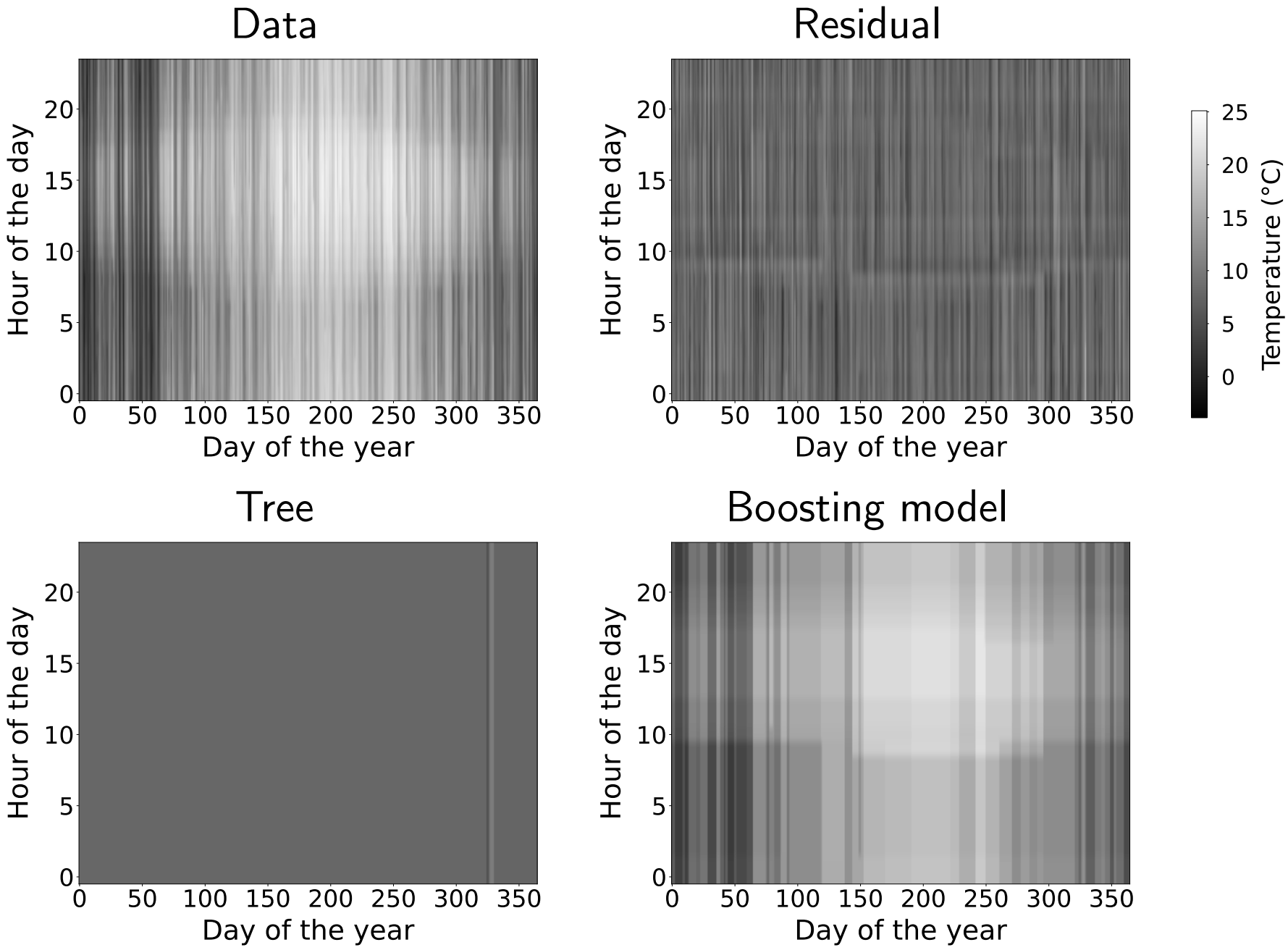
Tree 5


Data
Residual
Tree
Boosting model
Hour of the day
Day of the year
0
5
10
15
20
0
50
100
150
200
250
300
350
Temperature (°C)
25

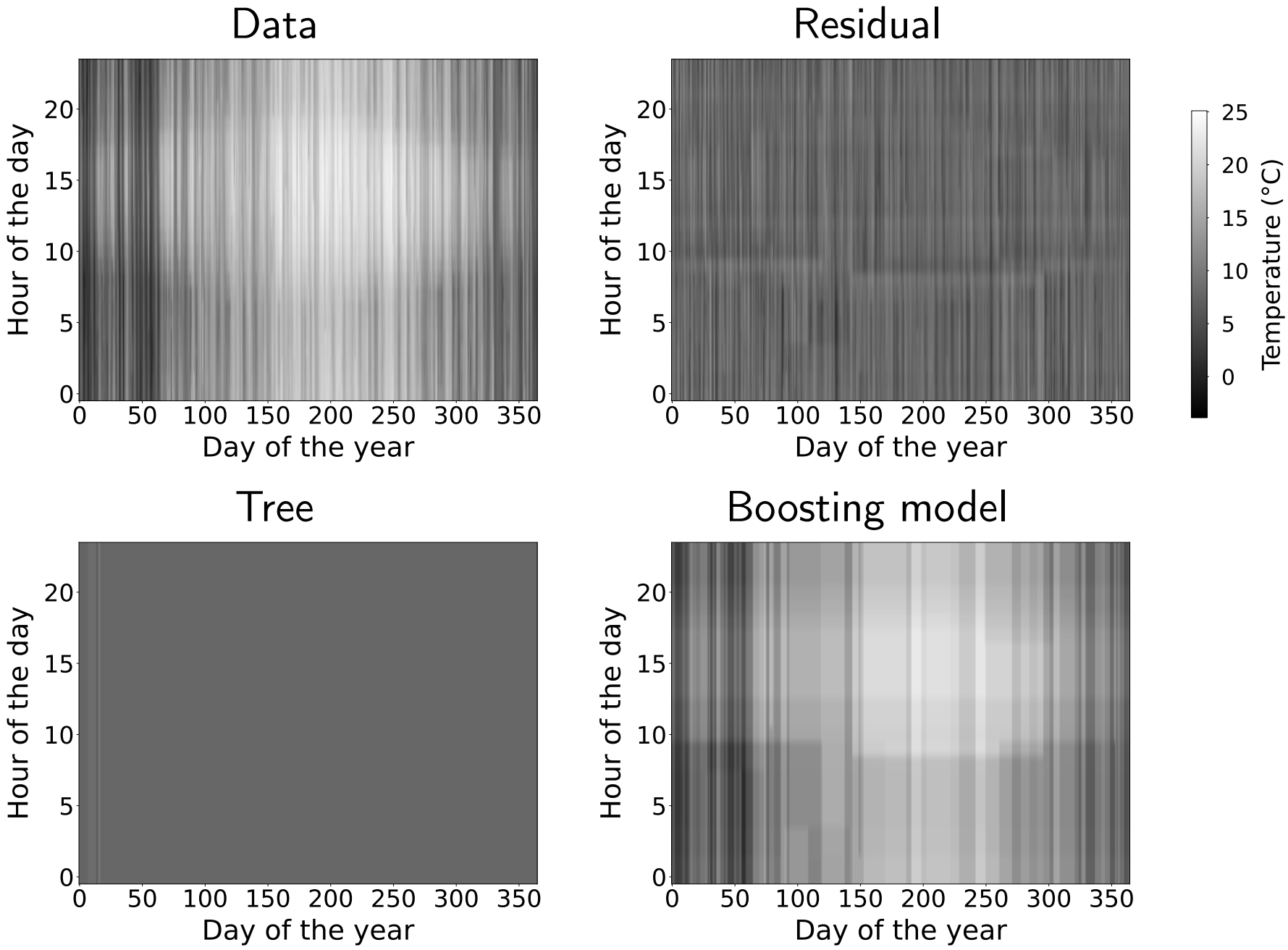
Tree 10


Data
Residual
Tree
Boosting model
Hour of the day
Day of the year
Temperature (°C)
0
5
10
15
20
25
0
50
100
150
200
250
300
350

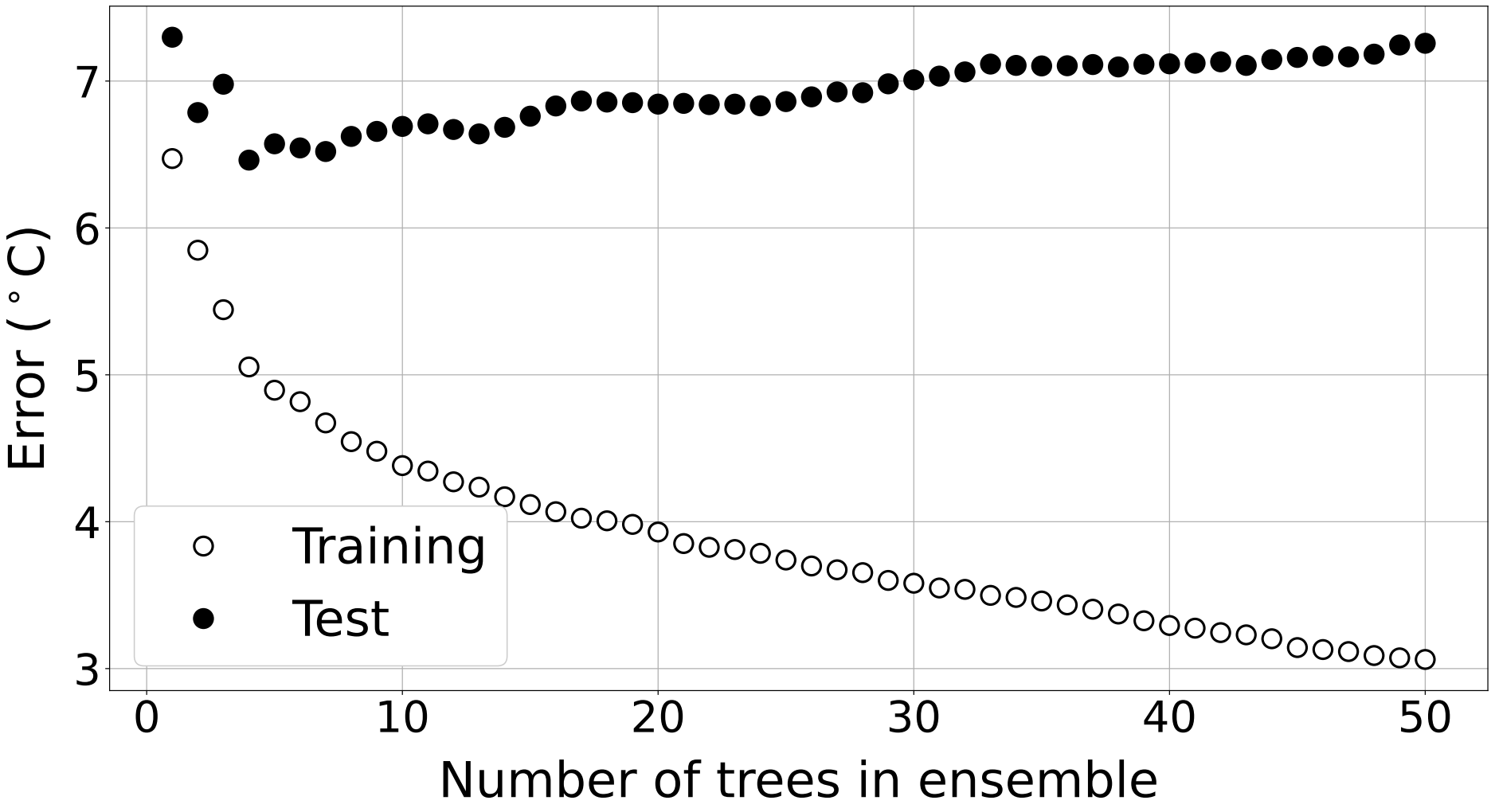
Tree 20


Data
Residual
Tree
Boosting model
Hour of the day
Day of the year
0
5
10
15
20
0
50
100
150
200
250
300
350
Temperature (°C)
0
5
10
15
20
25

Tree 30


Data
Residual
Tree
Boosting model
Hour of the day
Day of the year
0
5
10
15
20
0
50
100
150
200
250
300
350
Temperature (°C)
25

Naive boosting

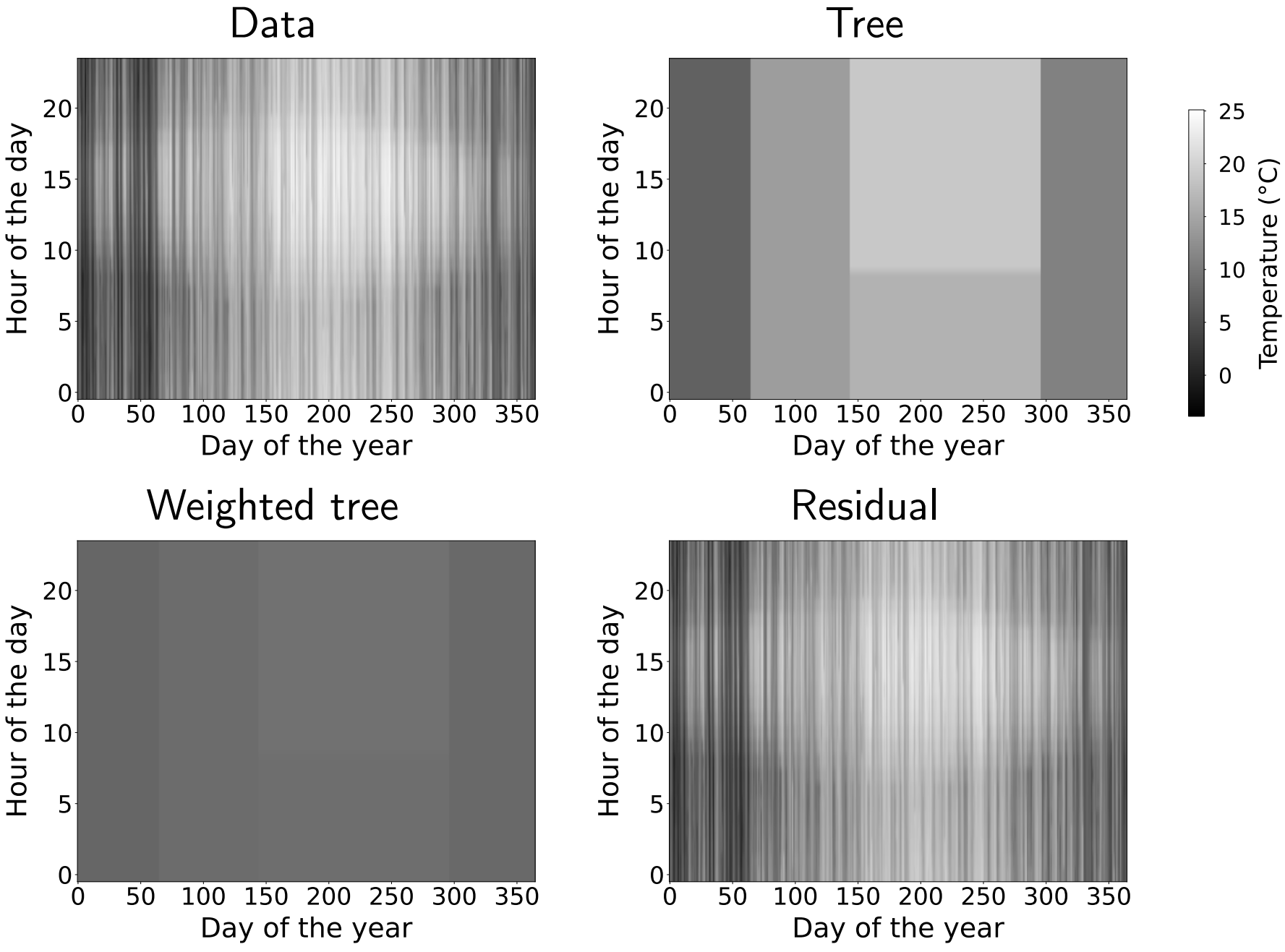


Challenge: How to avoid overfitting?

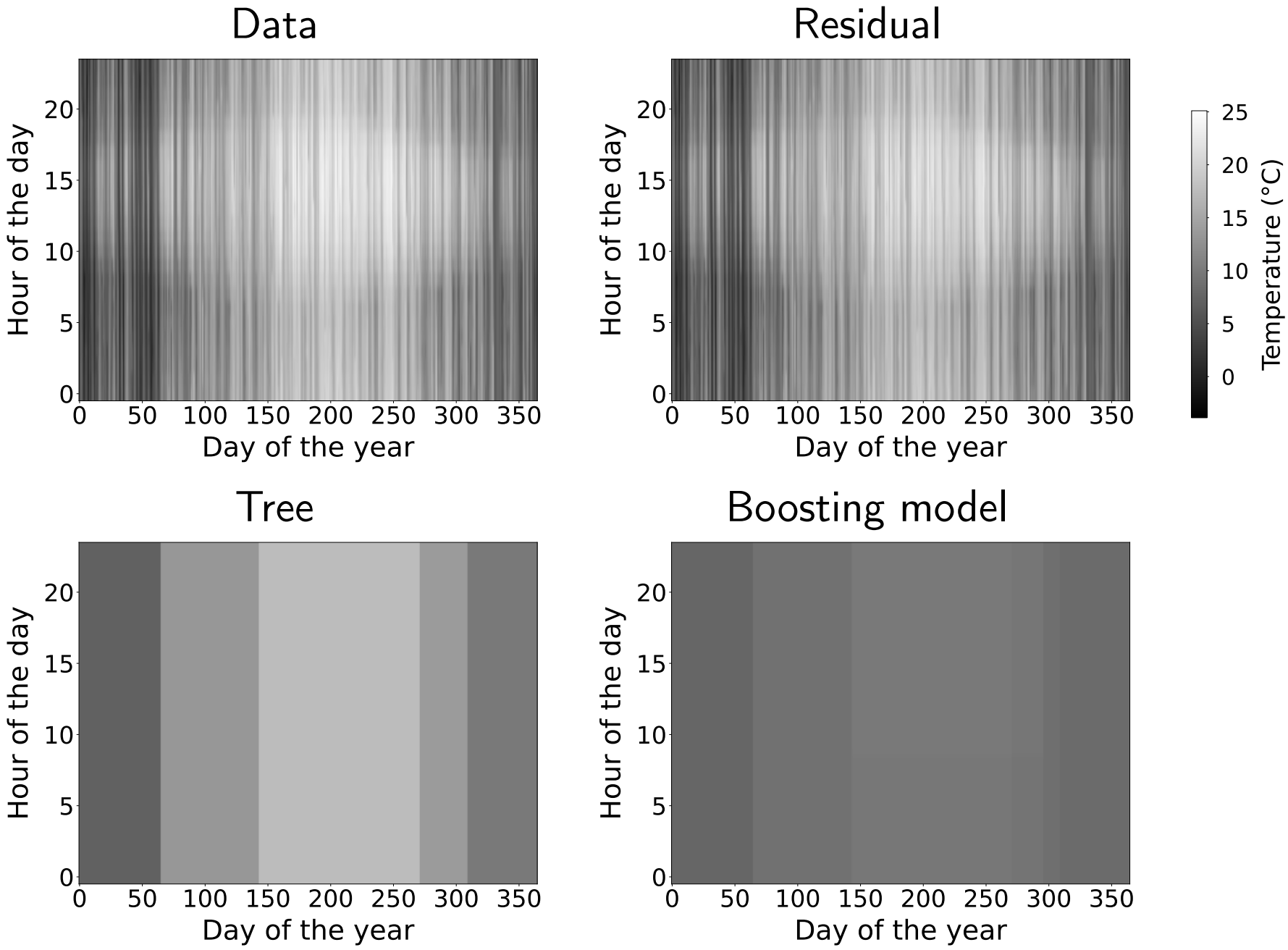
Scale down contribution of new trees

Simple approach: Multiply by a small constant γ

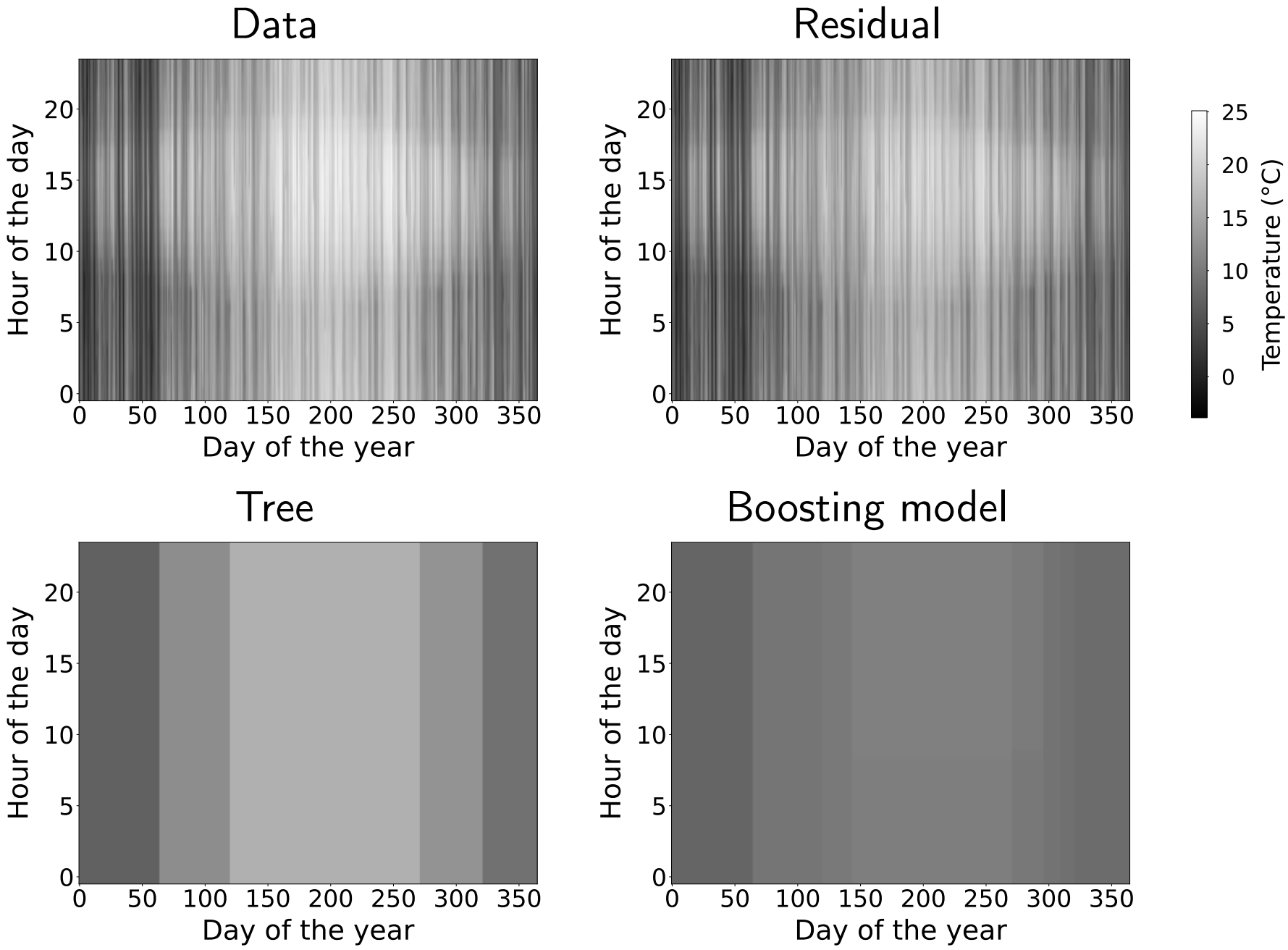
Boosting ($\gamma = 0.1$)



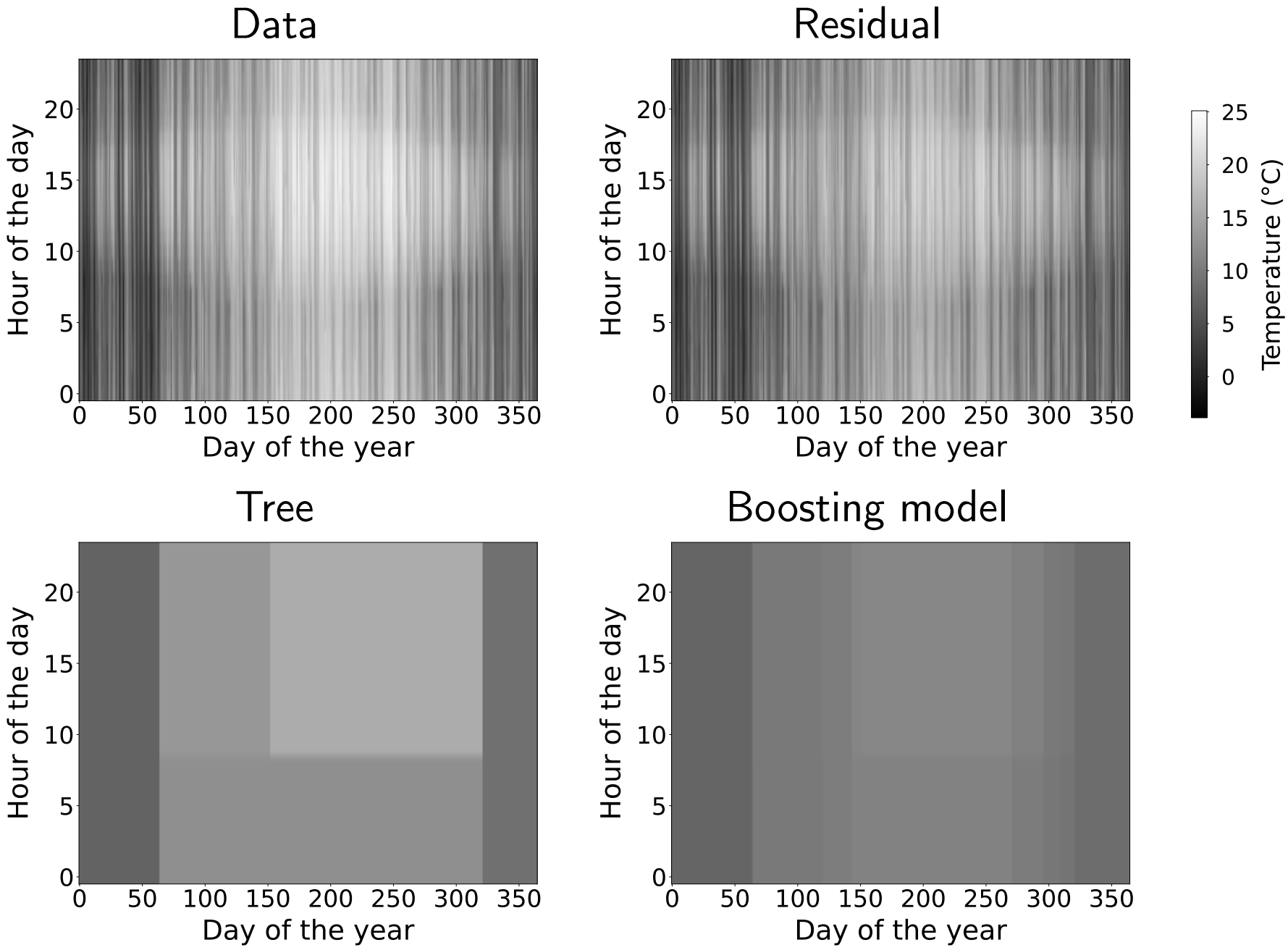
Tree 2


Data
Residual
Tree
Boosting model
Hour of the day
Day of the year
Temperature (°C)
0
5
10
15
20
25
0
50
100
150
200
250
300
350

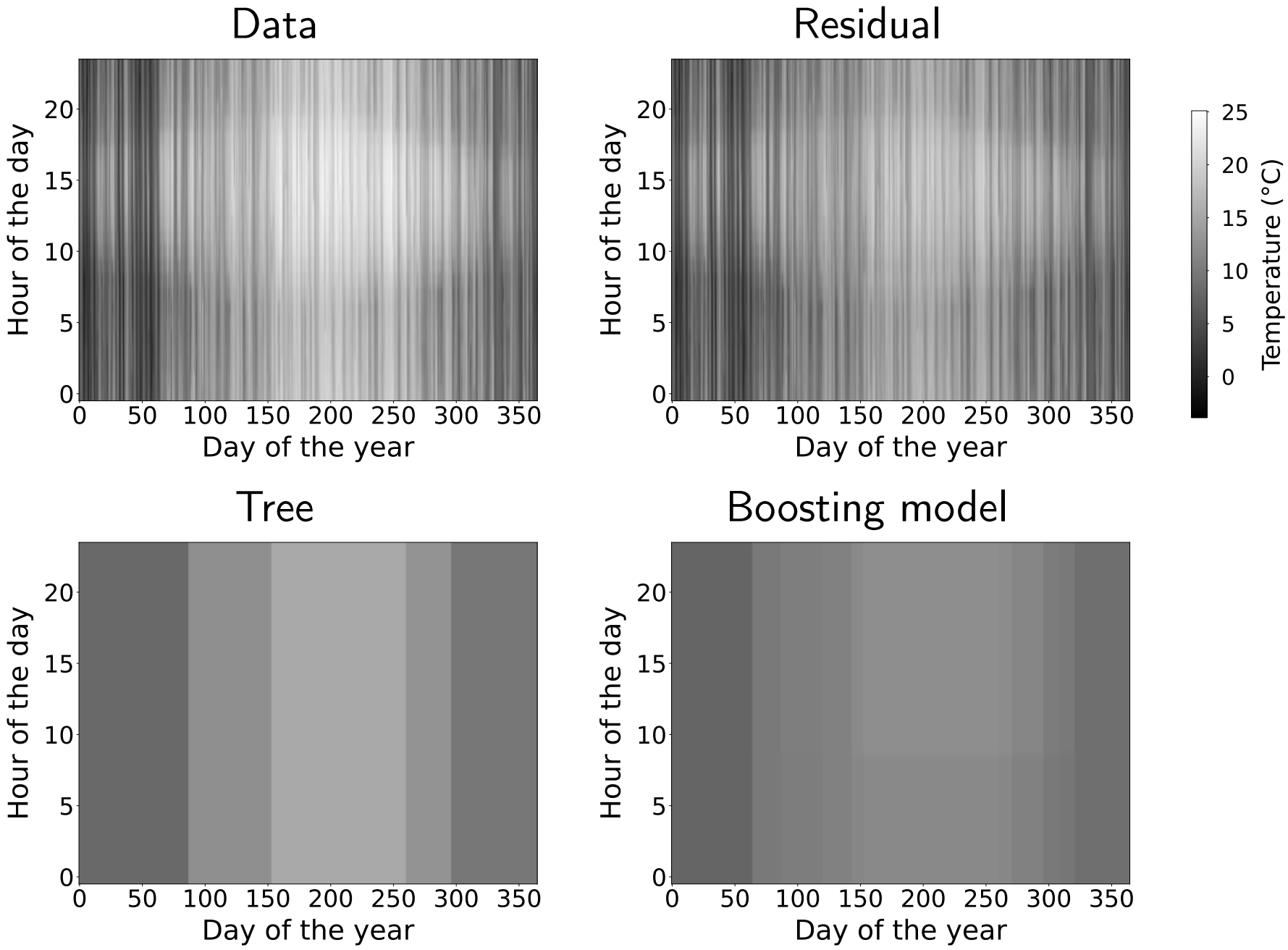
Tree 3


Data
Residual
Tree
Boosting model
Hour of the day
Day of the year
Temperature (°C)
0
5
10
15
20
25
0
50
100
150
200
250
300
350

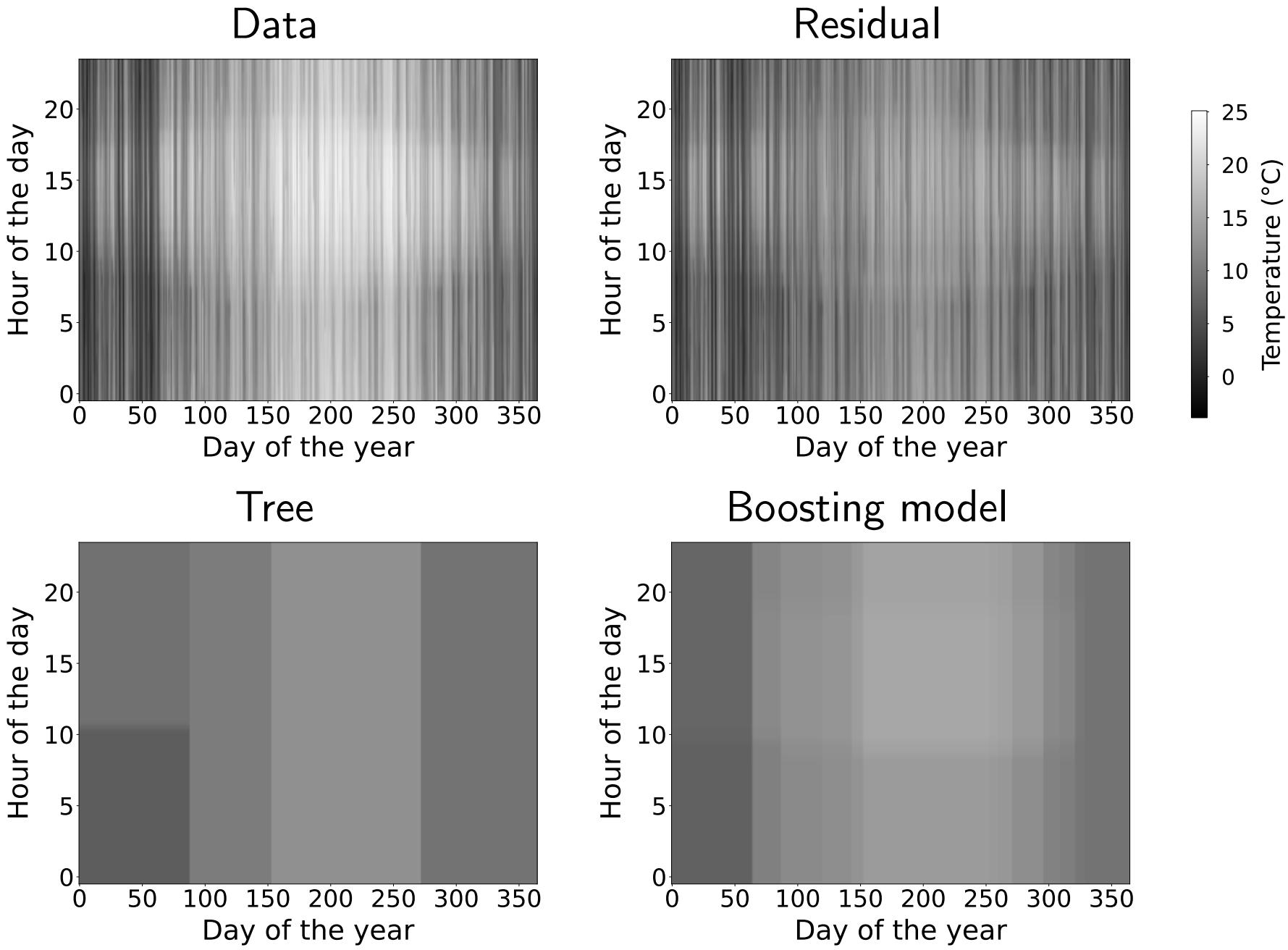
Tree 4


Data
Residual
Tree
Boosting model
Hour of the day
Day of the year
Temperature (°C)
0
5
10
15
20
0
50
100
150
200
250
300
350
25

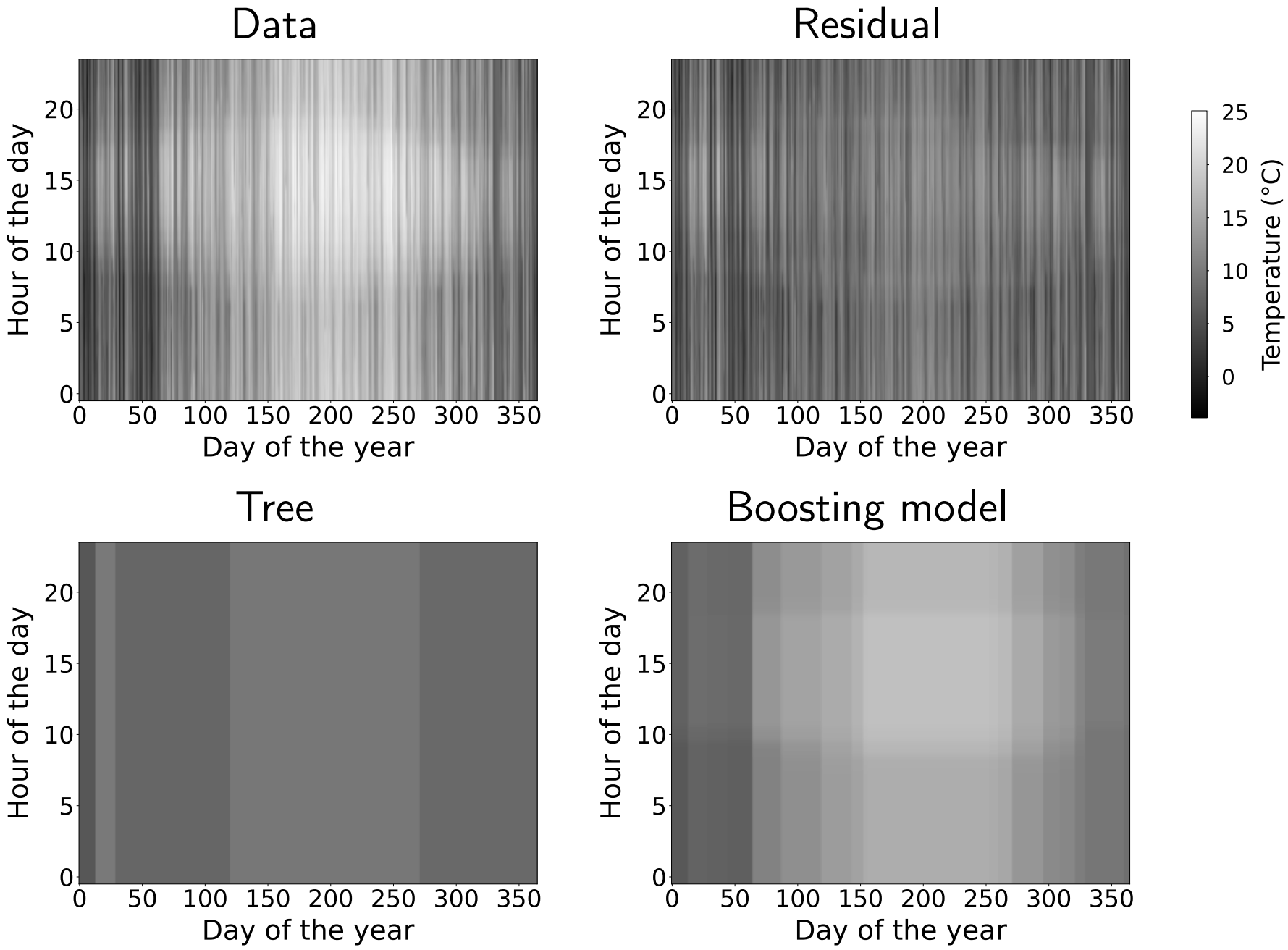
Tree 5


Data
Residual
Tree
Boosting model
Hour of the day
Day of the year
Temperature (°C)
0
5
10
15
20
25
0
50
100
150
200
250
300
350

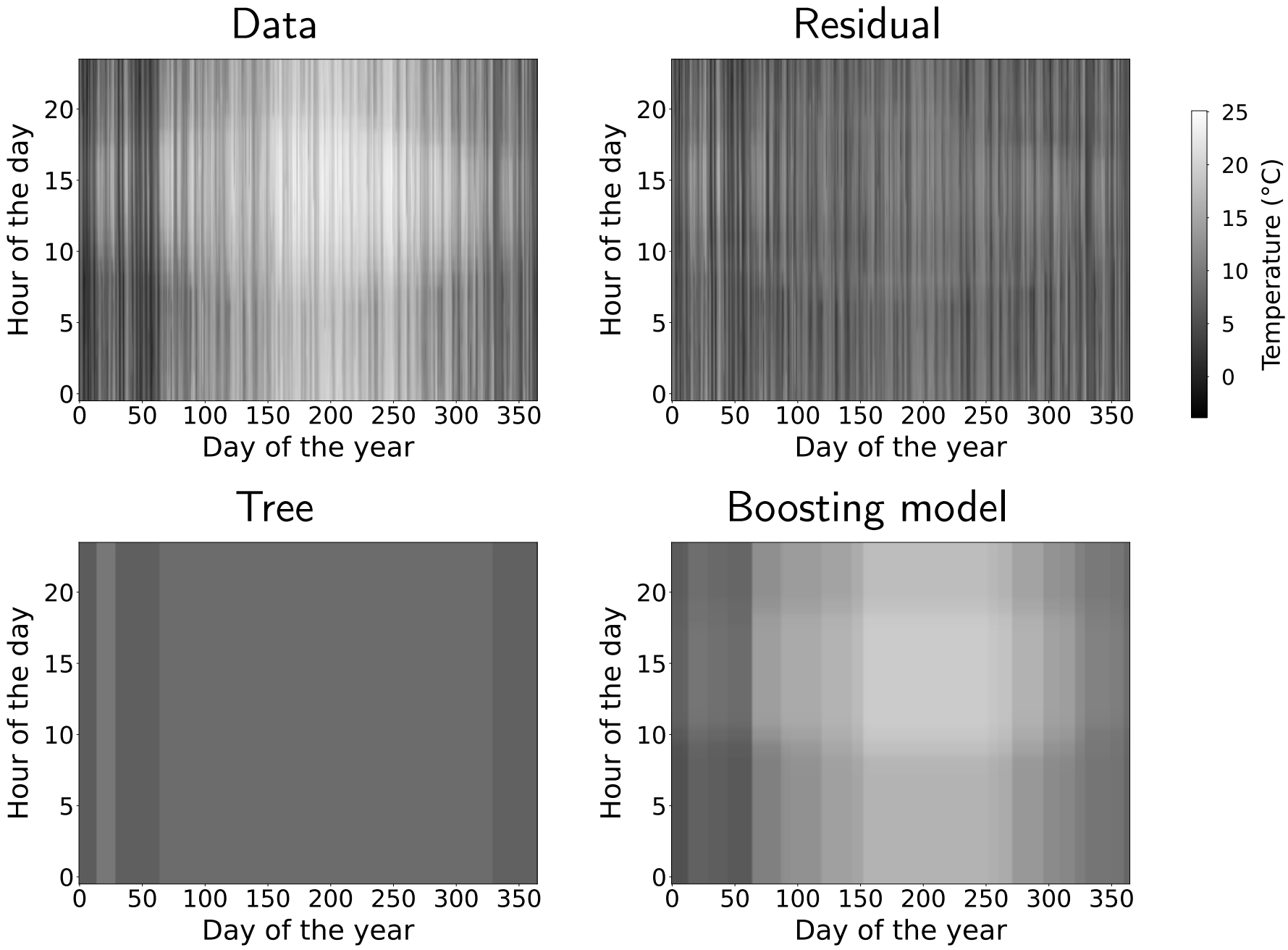
Tree 10


Data
Residual
Tree
Boosting model
Hour of the day
Day of the year
Temperature (°C)
0
5
10
15
20
0
50
100
150
200
250
300
350
25

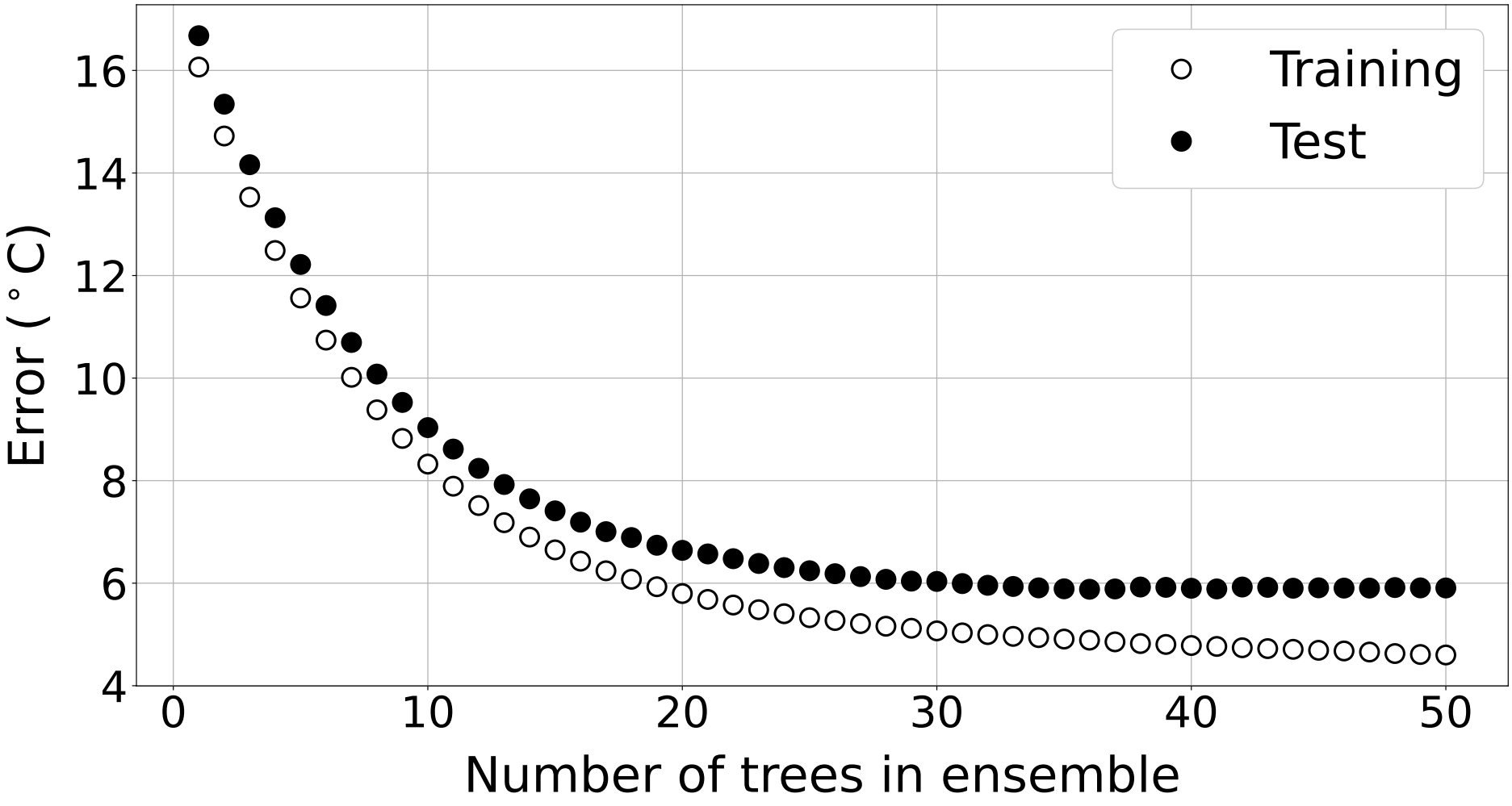
Tree 20


Data
Residual
Tree
Boosting model
Hour of the day
Day of the year
Temperature (°C)
0
5
10
15
20
25
0
50
100
150
200
250
300
350

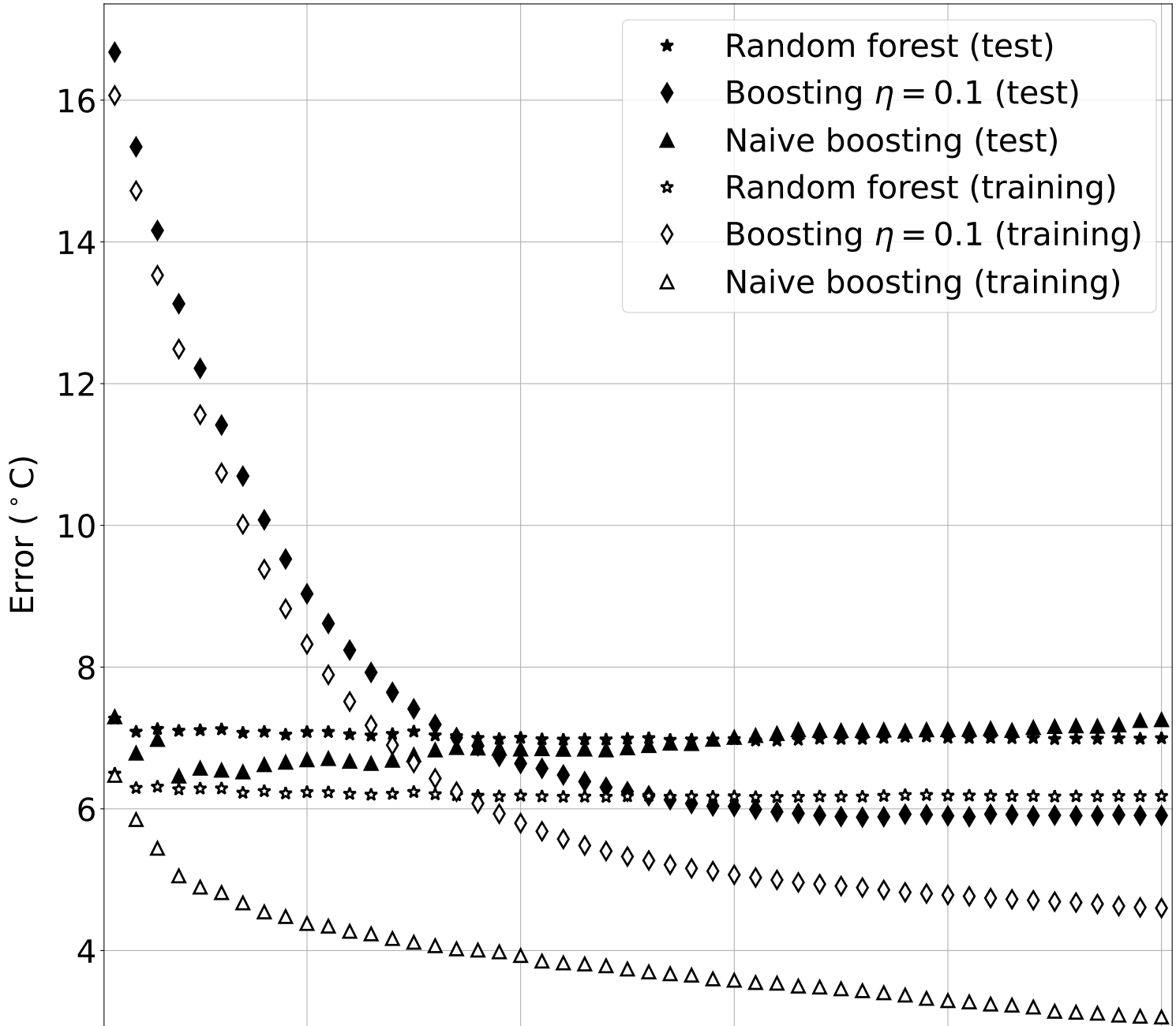
Tree 30


Data
Residual
Tree
Boosting model
Hour of the day
Day of the year
Temperature (°C)
0
5
10
15
20
25
0
50
100
150
200
250
300
350

Boosting


Training
Test
Error (°C)
Number of trees in ensemble
4
6
8
10
12
14
16
0
10
20
30
40
50

Bagging vs random forests vs boosting



What have we learned?

How to build complex models combining simple models (trees)

Three ensembling strategies:

1. Bagging: Averaging models fit to bootstrapped data
2. Random forests: Averaging randomized models fit to bootstrapped data
3. Boosting: Building complementary models by (carefully) fitting residuals

Tradeoff: We gain accuracy but lose interpretability

Overfitting is a constant threat!