Machine Learning B (2025)
Home Assignment 6

Carsten Jørgensen, student ID: skj730

Contents

1 Analysing AdaBoost **2**

2 Landcover Classification **4**

3 Majority vote **7**

4 Occam's kl-razor vs PAC-Bayes-kl **10**

1 Analysing AdaBoost

Let be a training set $T = \big\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\big\} \subset \mathbb{R}^d \times \{-1, +1\}$ of labelled instances. Denote by h_b the weak learner fitted in round b and $\alpha_b \in \mathbb{R}$ its importance coefficient. Finally $f_b = \sum_{p \le b} \alpha_p h_p$ with $f_0 := 0$.

Question 1

Prove, by induction over b, that the AdaBoost weight updates

$$w_i^{(b+1)} = \frac{w_i^{(b)} \, \exp\big(-\alpha_b \, y_i \, h_b(\mathbf{x}_i)\big)}{\sum_{j=1}^{n} w_j^{(b)} \, \exp\big(-\alpha_b \, y_j \, h_b(\mathbf{x}_j)\big)} \tag{1}$$

can be rewritten as

$$w_i^{(b+1)} = \frac{\exp\big(-y_i \, f_b(\mathbf{x}_i)\big)}{\sum_{j=1}^{n} \exp\big(-y_j \, f_b(\mathbf{x}_j)\big)}. \tag{2}$$

Proof. Base case $b = 0$. Before the first boosting step, the weights are uniform, $w_i^{(0)} = 1/n$. Now the r.h.s. of Equation 2 equals $e^0 / \sum_{j=1}^{n} e^0 = 1/n = w_i^{(0)}$, so the claim holds for $b = 0$.

Induction hypothesis. Assume that for some $b \ge 0$

$$w_i^{(b)} \;=\; \frac{\exp\big(-y_i \, f_{b-1}(\mathbf{x}_i)\big)}{\sum_{j=1}^{n} \exp\big(-y_j \, f_{b-1}(\mathbf{x}_j)\big)} \tag{3}$$

Induction step. Insert Equation 3 into the update rule Equation 1:

$$\begin{aligned}
w_i^{(b+1)} &= \frac{\exp\Big(-y_i \, f_{b-1}(\mathbf{x}_i)\Big) \, \exp\Big(-\alpha_b \, y_i \, h_b(\mathbf{x}_i)\Big)}{\sum_{j=1}^{n} \exp\Big(-y_j \, f_{b-1}(\mathbf{x}_j)\Big) \, \exp\Big(-\alpha_b \, y_j \, h_b(\mathbf{x}_j)\Big)} \\
&= \frac{\exp\Big(-y_i \big(f_{b-1}(\mathbf{x}_i) + \alpha_b h_b(\mathbf{x}_i)\big)\Big)}{\sum_{j=1}^{n} \exp\Big(-y_j \big(f_{b-1}(\mathbf{x}_j) + \alpha_b h_b(\mathbf{x}_j)\big)\Big)} \\
&= \frac{\exp\Big(-y_i \, f_b(\mathbf{x}_i)\Big)}{\sum_{j=1}^{n} \exp\Big(-y_j \, f_b(\mathbf{x}_j)\Big)}.
\end{aligned}$$

This is Equation 2 for round b+1, so by induction, Equation 2 holds for every $b \in \mathbb{N}$. □

Question 2

Assume that at every boosting round $b = 1, \ldots, B$ the weak learner returns a hypothesis h_b that fulfils

$$\mathbb{P}\big[\varepsilon_b \le \tfrac{1}{2} - \gamma\big] \;\ge\; 1 - \delta', \qquad \text{where } \varepsilon_b \;=\; \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}\big(h_b(x_i) \neq y_i\big),$$

where the probability is w.r.t. all the randomness of the learner. Here $\gamma \in (0, \frac{1}{2}]$ and $\delta' \in (0, 1)$ are fixed constants.

Find $B^\star \in \mathbb{Z}_+$ and $\delta^\star \in (0, 1)$ such that when Adaboost runs for $B^\star$ iterations, the error of the hypothesis h output by Adaboost is zero, i.e.

$$\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}(h(x_i) \neq y_i) = 0,$$

with probability at least $1 - \delta^\star$.

Step 1 – how many rounds are needed in the deterministic case? If *every* round fulfilled $\varepsilon_b \le \frac{1}{2} - \gamma$, then Theorem 10.2 of [SB14] gives for the final AdaBoost ensemble f_B

$$L_T(h) \;=\; \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}(h(x_i) \neq y_i) \le \exp(-2\gamma^2 B).$$

For the training error to be zero, we need the bound to be smaller than the smallest possible positive training error, which is $\frac{1}{n}$ (when exactly one example is misclassified). We require: $\exp(-2\gamma^2 B) < 1/n$. Hence, it suffices to choose

$$B^\star \;:=\; \left\lceil \frac{\ln(n)}{2\gamma^2} \right\rceil. \tag{1}$$

The empirical error is the fraction of mis-classified training points, hence can only take the discrete values $0, \frac{1}{n}, \frac{2}{n}, \ldots, 1$. With $B^\star$ as in Eq (1): $L_T(h) \le e^{-2\gamma^2 B^\star} < \frac{1}{n}$, the value cannot be any of $\frac{1}{n}, \frac{2}{n}, \ldots, 1$, leaving the single possibility $L_T(h) = 0$.

Step 2 – making the weak-learning condition hold in *all* rounds.
Let $E_b := \{\varepsilon_b \le \frac{1}{2} - \gamma\}$. By assumption $\mathbb{P}(E_b) \ge 1 - \delta'$, and the events $E_1, \ldots, E_{B^\star}$ are independent of one another (conditioned on T). Define

$$E := \bigcap_{b=1}^{B^\star} E_b.$$

Then

$$\mathbb{P}(E) \;=\; \prod_{b=1}^{B^\star} \mathbb{P}(E_b) \;\ge\; (1 - \delta')^{B^\star}.$$

Introduce

$$\delta^\star \;:=\; 1-(1-\delta')^{B^\star}. \tag{2}$$

Now put $F_b := E_b^{\mathrm{c}}$ and observe $\Pr(F_b) \;=\; \Pr\big[\varepsilon_b > \tfrac{1}{2}-\gamma\big] \le \delta'$. The bad event that *at least one* round fails is $\bigcup_{b=1}^{B^\star} F_b$, hence by the union

$$\Pr\Big[\bigcup_{b=1}^{B^\star} F_b\Big] \;\le\; \sum_{b=1}^{B^\star}\Pr(F_b) \;\le\; B^\star\delta'.$$

But the l.h.s. is exactly $\delta^\star$, so $\delta^\star \le B^\star\delta'$.

Step 3 – conclusion.
On the event E every round satisfies the deterministic weak guarantee, so the bound derived in Step 1 applies with $B = B^\star$ so $L_T(h) = 0$.

Hence, as *sets*, $E \subseteq \big\{L_T(h) = 0\big\}$. Because $\mathbb{P}(E) \ge 1-\delta^\star$, we have

$$\mathbb{P}(L_T(h) = 0) \ge \mathbb{P}(E) \ge\; 1-\delta^\star.$$

Answer. Run AdaBoost for

$$B^\star = \left\lceil \frac{\ln n}{2\gamma^2} \right\rceil$$

rounds. With

$$\delta^\star = 1-(1-\delta')^{B^\star} \le B^\star\delta'$$

the hypothesis returned by AdaBoost satisfies $\frac{1}{n}\sum_{i=1}^{n}\mathbf{1}\big(h(x_i) \neq y_i\big) = 0$ with probability at least $1-\delta^\star$.

2 Landcover Classification

Question 1

Justification for Central Pixel Selection

Why this is a reasonable choice:

- Computational efficiency: reduces the feature dimensionality from $12 \times 13 \times 13 \times 6 = 12,168$ to $12 \times 6 = 72$ features.
- Target relevance: the central pixel is the one we actually want to classify, so it is the most relevant.
- Temporal information: still captures temporal changes (12 timestamps) and spectral information (6 bands).
- Reduced overfitting: the lower dimensionality mitigates the risk of overfitting.

What might be lost:

- Spatial context: neighbouring pixels often provide valuable contextual information.
- Texture information: spatial patterns and textures can help distinguish classes.
- Edge effects: boundaries between different land-cover types are not represented.

Question 2

Grid search

A grid search over the following hyper-parameters was performed:

- Depth of decision trees: $[1, 2, 3]$
- Number of boosted trees: $[50, 100, 200]$

The best combination is a maximum depth of **3** and **200** boosted trees. See section 1 for code listing. Note we are using "SAMME" algorithm [RZH06] for AdaBoost as in the lectures.

Test accuracy

Using the best parameters from 2, we refitted the AdaBoost model on the entire training set. The resulting test accuracy is **0.7188**.

Confusion matrix

Figure 1 shows the confusion matrix. The model struggles to separate forest, grassland, and shrubland—three classes that are indeed quite similar spectrally and temporally.

Question 3

Suggestions for improving model quality

The Sentinel-2 data set contains 13 spectral bands. Vegetation indices such as the Normalized Difference Vegetation Index (NDVI) are known to improve land-cover classification but require precise knowledge of which six bands are included in the current data set.

We lack additional information about the timestamps, so exploiting seasonality is not currently feasible.

Thus, the following avenues remain for improvement:

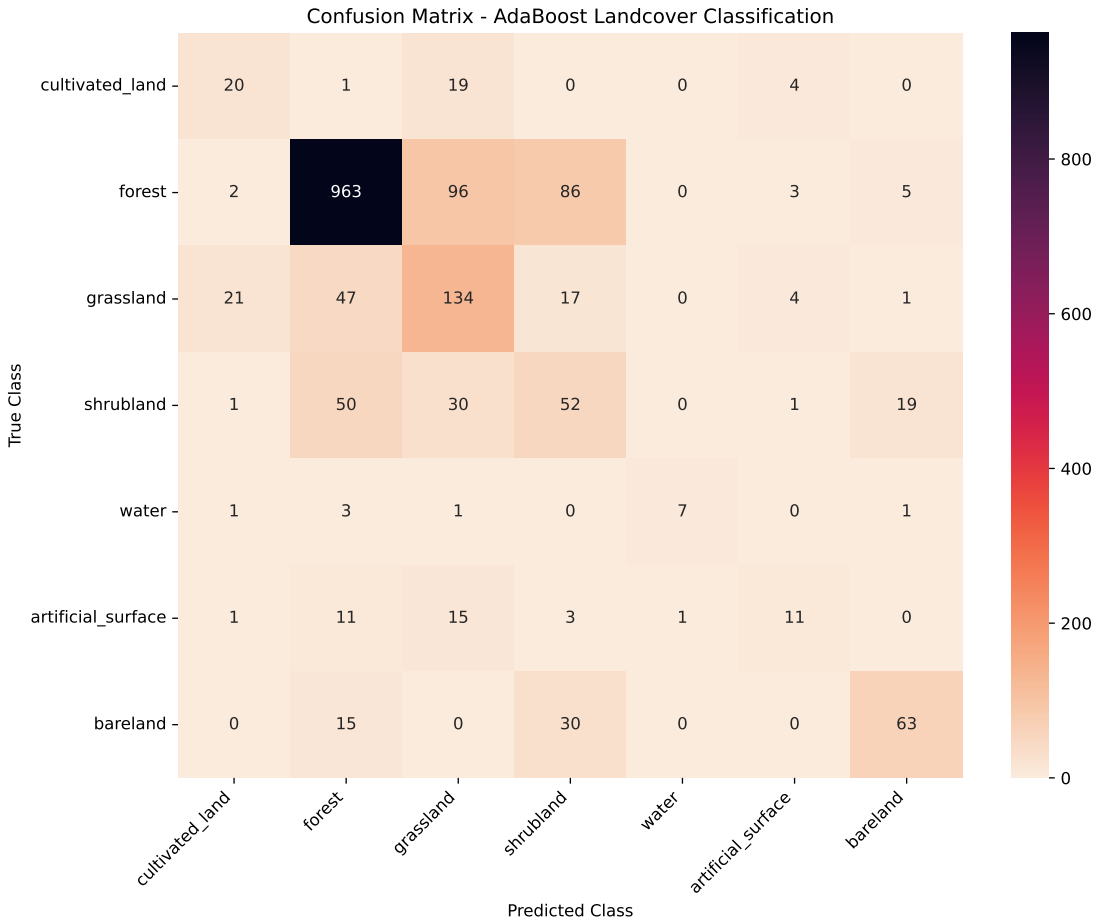

Figure 1: Confusion matrix for AdaBoost land-cover classification

- Spatial feature enhancements. By selecting only the centre pixel, we lose contextual information from neighbouring pixels.
- Including a 5×5 window around the centre pixel would enlarge the feature vector from 72 to $12 \times 5 \times 5 \times 6 = 1,800$.
- A lighter option is to compute the *mean* and *standard deviation* within the 5×5 window, yielding $12 \times 3 \times 6 = 216$ features.
- Increase the maximum depth of each decision tree.
- Increase the number of boosted trees.

Table 1 summarises several combinations of these improvements. Adding spatial features to the configuration with max depth = 3 and 200 boosted trees yields an increase of roughly 4 percentage points in test accuracy. Increasing both depth and the number of trees boosts accuracy further, but at the cost of much longer training times.[1]

[1]Experiments were running on a MacBook Pro 14-inch (2024) equipped with an Apple M3 Pro chip (11-cores), 18 GB unified memory, 1 TB SSD storage, running macOS Sequoia 15.5.

max depth	boosted trees	feature config	num features	train accuracy	test accuracy	fit time seconds
3	200	Central Pixels	72	0.7742	0.7188	18.9444
3	200	Central + Spatial	216	0.8253	0.7585	65.8750
8	400	Central Pixels	72	1.0000	0.8235	97.6872
8	400	Central + Spatial	216	1.0000	0.8269	324.9704

Table 1: Performance of various combinations of the suggested improvements

3 Majority vote

Question 1

Design an example of $\mathcal{H}$ and decision space $\mathbf{X}$, where $L(MV) = 0$ and $L(h) \geq \frac{1}{3}$ for all h.

Construction:

- Decision space: $\mathbf{X} = \{x_1, x_2, x_3\}$
- True labels: $Y(x_1) = +1$, $Y(x_2) = +1$, $Y(x_3) = +1$
- Hypothesis space: $\mathcal{H} = \{h_1, h_2, h_3\}$

Hypothesis Space:

Hypothesis	$h(x_1)$	$h(x_2)$	$h(x_3)$	Loss $L(h)$
h_1	$+1$	$+1$	-1	$\frac{1}{3}$
h_2	$+1$	-1	$+1$	$\frac{1}{3}$
h_3	-1	$+1$	$+1$	$\frac{1}{3}$

Table 2: Individual hypothesis predictions and losses

Majority Vote Analysis:

Point	h_1	h_2	h_3	Majority Vote	Correct?
x_1	$+1$	$+1$	-1	$+1$	✓
x_2	$+1$	-1	$+1$	$+1$	✓
x_3	-1	$+1$	$+1$	$+1$	✓

Table 3: Majority vote predictions

Result:

- Each individual hypothesis h_i makes exactly one error out of three predictions: $L(h_i) = \frac{1}{3}$ for all i
- The majority vote correctly predicts all three points: $L(\text{MV}) = 0$
- This demonstrates that $L(\text{MV}) = 0 < \frac{1}{3} \leq L(h)$ for all $h \in \mathcal{H}$

Question 2

Design an example of $\mathcal{H}$ and $\mathbf{X}$, where $L(\text{MV}) > L(h)$ for all h.

Construction:

- Decision space: Single point x with true label $Y(x) = +1$
- Hypothesis space: Three different probabilistic classifiers $\mathcal{H} = \{h_1, h_2, h_3\}$

Hypothesis Specifications: see Table 4

Hypothesis	$P(h(x) = +1)$	$P(h(x) = -1)$	Loss $L(h)$
h_1	0.25	0.75	0.75
h_2	0.30	0.70	0.70
h_3	0.35	0.65	0.65

Table 4: Individual hypothesis prediction probabilities and losses

Majority Vote Calculation: The majority vote predicts +1 when at least 2 out of 3 classifiers predict +1. See Table 5

Event	Probability
All three predict +1	$(0.25)(0.30)(0.35) = 0.02625$
h_1, h_2 predict +1, h_3 predicts -1	$(0.25)(0.30)(0.65) = 0.04875$
h_1, h_3 predict +1, h_2 predicts -1	$(0.25)(0.70)(0.35) = 0.06125$
h_2, h_3 predict +1, h_1 predicts -1	$(0.75)(0.30)(0.35) = 0.07875$
Total: $P(MV \text{ predicts } +1)$	$\mathbf{0.215}$

Table 5: Majority vote probability calculation

Result:

Classifier	Loss
h_1	0.75
h_2	0.70
h_3	0.65
Majority Vote	$L(MV) = 1 - 0.215 = \mathbf{0.785}$

Table 6: Comparison of losses

Conclusion:

From Table 6 since $L(\text{MV}) = 0.785 > 0.75 \geq \max\{L(h_1), L(h_2), L(h_3)\}$, we have $L(\text{MV}) > L(h)$ for all $h \in \mathcal{H}$.

This example demonstrates that the majority vote can perform worse than individual classifiers when all classifiers are systematically biased toward the wrong answer. The majority vote amplifies this systematic bias rather than correcting it.

Question 3

Let $\mathcal{H}$ be a hypothesis space, such that $|\mathcal{H}| = M$ and all $h \in \mathcal{H}$ have the same expected error, $L(h) = \frac{1}{2} - \varepsilon$ for $\varepsilon > 0$, and that the hypotheses in H make independent errors. Prove that $L(\text{MV}) \to 0$ as $|\mathcal{H}| \to \infty$.

Proof. We have M classifiers, each with error rate $p = \frac{1}{2} - \varepsilon$ where $\varepsilon > 0$, and their errors are independent. Therefore, each classifier predicts correctly with probability $1 - p = \frac{1}{2} + \varepsilon$.

Consider any example (X, Y). Let S be the number of classifiers that predict correctly out of the M total classifiers. Since the errors are independent:

$$S \sim \text{Binomial}(M, \frac{1}{2} + \varepsilon)$$

The majority vote is wrong if and only if $S < \frac{M}{2}$ (assuming M is odd for simplicity). Now the expected value is:

$$\mathbb{E}[S] = M\left(\frac{1}{2} + \varepsilon\right) = \frac{M}{2} + M\varepsilon$$

For the majority vote to be wrong, we need:

$$S < \frac{M}{2}$$

This is equivalent to:

$$S - \mathbb{E}[S] < \frac{M}{2} - \left(\frac{M}{2} + M\varepsilon\right) = -M\varepsilon$$

By Hoeffding's inequality, for independent random variables bounded in $[0, 1]$:

$$\mathbb{P}(S - \mathbb{E}[S] \leq -M\varepsilon) \leq \exp\left(-\frac{2(M\varepsilon)^2}{M}\right) = \exp(-2M\varepsilon^2)$$

Conclusion:

$$L(MV) = \mathbb{P}(\text{majority vote is wrong}) \leq \exp(-2M\varepsilon^2) \to 0 \text{ as } M \to \infty$$

The convergence is **exponentially fast** with rate $2\varepsilon^2$.

Even when individual classifiers are only slightly better than random (i.e., ε can be arbitrarily small but positive), the majority vote of many independent classifiers achieves perfect accuracy in the limit. This demonstrates the power of ensemble methods when the base classifiers are better than random and make independent errors. □

4 Occam's kl-razor vs PAC-Bayes-kl

Question 1

Theorem (Occam's kl-razor inequality for soft selection)**.** *Under the conditions of Theorem 3.38 [Sel25]:*

$$P\left(\exists\rho : \mathrm{kl}\left(\mathbb{E}_\rho[\hat{L}(h,S)]\|\mathbb{E}_\rho[L(h)]\right) \geq \frac{\mathbb{E}_\rho\left[\ln\frac{1}{\pi(h)}\right] + \ln\frac{1}{\delta}}{n}\right) \leq \delta$$

Proof. From Theorem 3.38 [Sel25], we know that with probability at least $1-\delta$, for all $h \in H$:

$$\mathrm{kl}(\hat{L}(h,S)\|L(h)) \leq \frac{\ln\frac{1}{\pi(h)\delta}}{n}$$

For any distribution ρ over H, we apply the convexity of the kl divergence (Corollary 2.19, [Sel25]):

$$\mathrm{kl}(\mathbb{E}_\rho[\hat{L}(h,S)]\|\mathbb{E}_\rho[L(h)]) \leq \mathbb{E}_\rho[\mathrm{kl}(\hat{L}(h,S)\|L(h))]$$

Since the bound from Theorem 3.38 [Sel25] holds for all $h \in H$ with probability at least $1-\delta$, we have:

$$\begin{aligned}\mathbb{E}_\rho[\mathrm{kl}(\hat{L}(h,S)\|L(h))] &\leq \mathbb{E}_\rho\left[\frac{\ln\frac{1}{\pi(h)\delta}}{n}\right] \\ &= \mathbb{E}_\rho\left[\frac{\ln\frac{1}{\pi(h)} + \ln\frac{1}{\delta}}{n}\right] \\ &= \frac{\mathbb{E}_\rho\left[\ln\frac{1}{\pi(h)}\right] + \ln\frac{1}{\delta}}{n}\end{aligned}$$

Therefore:

$$\mathrm{kl}(\mathbb{E}_\rho[\hat{L}(h,S)]\|\mathbb{E}_\rho[L(h)]) \leq \frac{\mathbb{E}_\rho\left[\ln\frac{1}{\pi(h)}\right] + \ln\frac{1}{\delta}}{n}$$

with probability at least $1-\delta$. This completes the proof. □

Question 2

Comparison of Occam's kl-razor and PAC-Bayes-kl

We have **Occam's kl-razor (Theorem 3.40):**

$$\mathrm{kl}(\mathbb{E}_\rho[\hat{L}(h,S)]\|\mathbb{E}_\rho[L(h)]) \leq \frac{\mathbb{E}_\rho\left[\ln\frac{1}{\pi(h)}\right] + \ln\frac{1}{\delta}}{n}$$

and **PAC-Bayes-kl (Theorem 3.26):**

$$\mathrm{kl}(\mathbb{E}_\rho[\hat{L}(h,S)]\|\mathbb{E}_\rho[L(h)]) \leq \frac{\mathrm{KL}(\rho\|\pi) + \ln\frac{2\sqrt{n}}{\delta}}{n}$$

The difference lies in the complexity terms:

- **Occam's:** $\mathbb{E}_\rho \left[\ln \frac{1}{\pi(h)}\right]$
- **PAC-Bayes:** $\mathrm{KL}(\rho\|\pi)$

We can decompose the KL divergence as:

$$\mathrm{KL}(\rho\|\pi) = \mathbb{E}_\rho \left[\ln \frac{\rho(h)}{\pi(h)}\right] = \mathbb{E}_\rho \left[\ln \frac{1}{\pi(h)}\right] - H(\rho)$$

where $H(\rho) = -\mathbb{E}_\rho[\ln \rho(h)]$ is the entropy of ρ.

Advantages of PAC-Bayes-kl

1. **Entropy Bonus:** PAC-Bayes-kl includes a $-H(\rho)$ term, providing a "bonus" for randomization. When ρ spreads probability mass over multiple hypotheses (high entropy), the complexity penalty decreases.
2. **Flexible Complexity Control:** The complexity is measured by the actual amount of selection $\mathrm{KL}(\rho\|\pi)$ rather than just the average complexity $E_\rho \left[\ln \frac{1}{\pi(h)}\right]$.

Disadvantages of PAC-Bayes-kl

1. **Worse Constants:** The constant term $\ln \frac{2\sqrt{n}}{\delta}$ is worse than $\ln \frac{1}{\delta}$ by a factor of approximately $\ln(2\sqrt{n}) \approx \frac{\ln n}{2}$.
2. **More Complex Analysis:** The proof requires the sophisticated change-of-measure inequality and PAC-Bayes lemma, while Occam's kl-razor follows more directly from convexity and union bounds.

Appendix

Code for GridSearch

```
param_grid = {
    "estimator__max_depth": [1, 2, 3],
    "n_estimators": [50, 100, 200],
}

# Create base classifier (Decision Tree with Gini impurity)
base_classifier = DecisionTreeClassifier(criterion="gini",
    random_state=42)

# Create AdaBoost classifier with SAMME algorithm
ada_boost = AdaBoostClassifier(estimator=base_classifier,
    algorithm="SAMME", random_state=42)

# Set up 2-fold cross validation
cv = StratifiedKFold(n_splits=2, shuffle=True, random_state=42)
```

```
# Perform grid search
print("Performing grid search with 2-fold cross validation...")
grid_search = GridSearchCV(estimator=ada_boost, param_grid=
    param_grid, cv=cv, scoring="accuracy", n_jobs=-1, verbose=1)

# Fit grid search
grid_search.fit(X_train_central, y_train_flat)
```

Listing 1: Grid search

References

[RZH06] Saharon Rosset, Hui Zou, and Trevor Hastie. "Multi-class AdaBoost". In: *Statistics and its interface* 2 (Feb. 2006). DOI: 10.4310/SII.2009.v2.n3.a8.

[SB14] Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014. ISBN: 1107057132.

[Sel25] Yevgeny Seldin. *Machine Learning The Science of Selection under Uncertainty*. Apr. 22, 2025. URL: https://sites.google.com/site/yevgenyseldin/teaching.