Ensemble Methods and Statistical Supervised Machine Learning

Exercise sheet: Ensemble methods and PAC-Bayesian Learning

Master MIASHS & MALIA

Guillaume Metzler
Institut de Communication (ICOM)
Université de Lyon, Université Lumière Lyon 2
Laboratoire ERIC UR 3083, Lyon, France
guillaume.metzler@univ-lyon2.fr

Abstract

Twenty exercises: ten on ensemble methods (bagging, random forests, AdaBoost, gradient boosting, XGBoost, stacking and voting), six computation exercises on these models, and four on PAC-Bayesian learning (Gibbs and Bayes risks, PAC-Bayesian bounds, Gibbs posterior, numerical certificates). They can be done with pen and paper (a calculator helps). Labels are in $\{-1, +1\}$ unless stated otherwise.

1 Ensemble methods

Exercise 1 – Bagging regression stumps by hand

Eight points: $x_i = i$, $y = (1,\ 2,\ 1,\ 2,\ 6,\ 5,\ 7,\ 6)$. A regression stump splits at a threshold $\theta \in \{1.5, 2.5, \dots, 7.5\}$ and predicts on each side the mean of the y of the *sample* points on that side; it is fitted by minimizing the sum of squared errors on the sample (if several thresholds give the same split of the sample, take the smallest). Three bootstrap samples were drawn (values of x, with repetitions):

$$S_1 = \{1, 1, 3, 4, 5, 7, 7, 8\}, \qquad S_2 = \{2, 2, 3, 4, 4, 6, 7, 8\}, \qquad S_3 = \{1, 2, 3, 5, 5, 6, 6, 8\}.$$

1. Fit a stump on each bootstrap sample (threshold and two leaf values).
2. Compute the bagged prediction (average of the three stumps) at every x_i. Compare with the stump fitted on the full data ($\theta = 4.5$, leaves 1.5 and 6). What does bagging do around $x = 4$ and $x = 5$?
3. For each point, list the stumps for which it is out-of-bag, and give its OOB prediction. Compute the OOB mean squared error on the points that have one. Why do points 3 and 8 have no OOB prediction, and why does this become rare with 100 stumps?
4. Compute the training MSE of the bagged predictor on the eight points and compare it with its OOB MSE. Which one would you report?

Exercise 2 – What bagging reduces, and what it does not

For a predictor f at a point x, the expected squared error decomposes as bias^2+variance+noise. Bagging averages T predictors with the same bias and variance v, and pairwise correlation ρ (first sheet, Exercise 1).

1. Show that the bias of the bagged predictor equals the bias of a single one, and give its variance.
2. A deep tree has bias$^2 = 0.01$, $v = 0.04$; trees on bootstrap samples have $\rho = 0.2$. A stump has bias$^2 = 0.09$, $v = 0.005$, $\rho = 0.6$. Compute bias^2+variance for one model and for $T = 100$ bagged models, in both cases (ignore the noise). Which base learner benefits from bagging? Which method would you use to improve the stump?
3. Why does bagging a linear regression (fitted by least squares) bring almost nothing?

Exercise 3 – Random forests: random features and decorrelation

A dataset has $d = 16$ features, of which only 2 are informative. A random forest draws m' candidate features at each split.

1. With $m' = 4$ ($\approx \sqrt{d}$), what is the probability that a given feature is among the candidates? That *none* of the two informative features is? Same question with $m' = 8$.
2. What happens to a split when no informative feature is available? Deduce the effect of m' on the strength of the trees and on their correlation.
3. Why does a random forest not overfit when the number of trees T increases, while boosting can? What is T then chosen for?

Exercise 4 – AdaBoost by hand on two features

Eight points in the plane, with labels:

point	1	2	3	4	5	6	7	8
x_1	1	2	3	4	5	6	7	8
x_2	2	1	4	6	7	8	3	5
y	-1	$+1$	-1	$+1$	$+1$	$+1$	-1	$+1$

The weak learners are stumps on one feature, $h(x) = s \cdot \mathrm{sign}(x_j - \theta)$, $j \in \{1, 2\}$, $s \in \{-1, +1\}$, with θ halfway between two consecutive values $(1.5, 2.5, \ldots, 7.5)$. Discrete AdaBoost: $w_i = \frac{1}{8}$ at the start; at each round choose the stump with the smallest weighted error ε_t, set $\alpha_t = \frac{1}{2} \ln \frac{1-\varepsilon_t}{\varepsilon_t}$, update $w_i \leftarrow w_i e^{-\alpha_t y_i h_t(x_i)}$ and renormalize.

1. **Round 1.** Show that the stump $h_1 = \mathrm{sign}(x_2 - 4.5)$ makes a single error and that no stump makes none. Compute ε_1, α_1 and the new weights. Check that the misclassified point now carries half of the total weight.
2. **Round 2.** With the new weights, the best stump is $h_2 = \mathrm{sign}(x_1 - 1.5)$. On which points does it err? Compute ε_2, α_2 and the new weights. Compute the scores $F_2 = \alpha_1 h_1 + \alpha_2 h_2$ and the training error of $\mathrm{sign}(F_2)$.
3. **Round 3.** The best stump is $h_3 = -\mathrm{sign}(x_2 - 1.5)$ (it predicts $+1$ only for $x_2 < 1.5$). On which points does it err? Compute ε_3, α_3, the scores F_3 and the training error of $\mathrm{sign}(F_3)$.
4. Compute the bound $\prod_t 2\sqrt{\varepsilon_t(1-\varepsilon_t)}$ on the training error after each round and compare with the actual errors.

5. **Label noise.** If point 2 had a wrong label and were misclassified at every round with $\varepsilon_t = 0.2$, by how much would its weight grow relative to a correctly classified point after 5 rounds? Why does this make AdaBoost sensitive to noise?

Exercise 5 – Gradient boosting with a robust loss

Regression data $y = (2, 3, 4, 5, 30)$ at $x = 1, \dots, 5$; the last point is an outlier.

1. Squared loss: compute the initial constant F_0 (the mean) and the pseudo-residuals.
2. Absolute loss $\ell(y, F) = |y - F|$: show that the best constant is the median and that the pseudo-residuals are $\text{sign}(y_i - F(x_i))$. Compute them.
3. Compare: what does the first tree try to fit in each case? Which loss would you choose with outliers? (The Huber loss combines both: quadratic for small residuals, linear for large ones.)

Exercise 6 – Gradient boosting for classification by hand

Labels $y \in \{0, 1\}$, logistic loss, score F and probability $p = \frac{1}{1+e^{-F}}$. Data $x = (1, 2, 3, 4)$, $y = (0, 1, 1, 1)$. Recall: pseudo-residual $r_i = y_i - p_i$; with a Newton step, the value of a leaf is $\frac{\sum r_i}{\sum p_i(1-p_i)}$ over its points.

1. Show that the best constant is $F_0 = \ln \frac{\bar{y}}{1-\bar{y}}$ (the log-odds). Compute F_0, p and the residuals.
2. Fit a stump on the residuals: the split is $\theta = 1.5$ (why?). Compute the two Newton leaf values.
3. Compute F_1, the new probabilities and the logistic loss before and after (with $\nu = 1$). Same with $\nu = 0.1$. Comment.

Exercise 7 – XGBoost by hand: leaf values and gains

Regression with the squared loss $\ell(y, F) = \frac{1}{2}(y - F)^2$, so $g_i = F(x_i) - y_i$ and $h_i = 1$. Data:

x	1	2	3	4	5	6
y	2	4	3	8	7	12

The initial prediction is the mean, $F_0 = 6$. Recall: leaf value $w^\star = -\frac{G}{H+\lambda}$ and gain $\frac{1}{2}\big[\frac{G_L^2}{H_L+\lambda} + \frac{G_R^2}{H_R+\lambda} - \frac{(G_L+G_R)^2}{H_L+H_R+\lambda}\big] - \gamma$.

1. Compute the g_i, h_i, G and H. What is the value of a single leaf?
2. With $\lambda = 0$, $\gamma = 0$, compute the gain and the two leaf values of each split $\theta \in \{1.5, \dots, 5.5\}$. Which split is chosen? Check that with $\lambda = 0$ the leaf values are the means of the residuals $y_i - F_0$.
3. Same with $\lambda = 1$. Is the same split chosen? How do the leaf values change?
4. With $\lambda = 1$, the right leaf ($x \geq 4$) could be split again at $\theta = 4.5$ or $\theta = 5.5$. Compute both gains. With $\gamma = 1$, is the second split made?
5. Keep the depth-one tree ($\lambda = 1$). Compute the new predictions and the mean squared error with a learning rate $\eta = 1$ and $\eta = 0.3$ (initial MSE: 11.67).

Exercise 8 – Gradient boosting with stumps: the effect of the learning rate

Regression, squared loss, data:

x	1	2	3	4	5	6
y	1	3	2	8	7	9

Start from $F_0 = \bar{y}$. At each iteration we use **the same model**: a stump split at $\theta = 3.5$ whose two leaves are refitted to the mean of the current residuals $r_i = y_i - F_{t-1}(x_i)$ on each side; then $F_t = F_{t-1} + \nu\, g_t$.

1. Compute F_0, its MSE, and the residuals.
2. Run three iterations with $\nu = 1$, $\nu = 0.5$ and $\nu = 0.2$: give for each the leaf values, the predictions F_t on each side, and the MSE.
3. Show that on the left side $F_t = 5 - 3(1 - (1-\nu)^t)$ (and symmetrically on the right), and that the MSE is $\frac{2}{3} + 9(1-\nu)^{2t}$. What does the constant $\frac{2}{3}$ represent?
4. How many iterations are needed with $\nu = 0.5$ and $\nu = 0.2$ to cover 99% of the distance between F_0 and the limit? Why use a small ν anyway?
5. With $\nu = 1$, the residuals after one iteration are $(-1, 1, 0, 0, -1, 1)$. If the split were refitted at each iteration, would $\theta = 3.5$ still be chosen? What does this show?

Exercise 9 – Voting and stacking: hard, soft, learned

Three classifiers give the following probabilities of class 1 on three examples:

	ex. 1	ex. 2	ex. 3
A	0.90	0.40	0.20
B	0.45	0.45	0.90
C	0.45	0.60	0.35

1. Compute the hard vote (majority of the predicted classes, threshold 0.5) and the soft vote (average probability, threshold 0.5) on each example. Where do they differ, and why?
2. Stacking with $K = 5$ folds and these 3 base models: how many models are trained in total? Describe the data on which the meta-model is trained.
3. Give two situations where stacking is worth its cost, and one where a simple soft vote is preferable.

Exercise 10 – Random forest: training and out-of-bag performance

Ten points on a line, $x_i = i$, with labels $y = (-1, -1, +1, -1, -1, +1, +1, -1, +1, +1)$ (points 3 and 8 are noisy). A forest of three classification stumps $h(x) = \text{sign}(x - \theta)$ (predict $+1$ to the right of θ) is built on three bootstrap samples; each stump minimizes the number of errors on its sample. The samples (values of x, with repetitions) and the thresholds found are:

$$\begin{array}{lll} \text{tree 1} & S_1 = \{1,1,2,5,5,6,6,6,6,7\} & \theta_1 = 5.5 \\ \text{tree 2} & S_2 = \{2,3,3,3,4,5,6,7,8,9\} & \theta_2 = 2.5 \\ \text{tree 3} & S_3 = \{1,2,3,4,5,7,8,8,9,9\} & \theta_3 = 8.5 \end{array}$$

1. Check the threshold of tree 2: count its errors on S_2 for $\theta = 2.5$ and for $\theta = 5.5$.

2. Give the predictions of each tree on the ten points and the **training error** of each tree on the full data.
3. Compute the predictions of the forest (majority of the 3 trees) and its training error.
4. For each tree, list its out-of-bag points and compute its OOB error. Then compute the **OOB prediction** of the forest on each point (majority of the trees for which the point is out-of-bag) and the OOB error of the forest.
5. Compare the training error and the OOB error. Which one estimates the test error, and why is the other one optimistic?

2 Computations on ensemble models

Exercise 11 – Counting in a bootstrap sample

A bootstrap sample draws $m = 10$ examples with replacement from 10.

1. The number of copies of a given example follows a binomial distribution. Which one? Compute the probabilities that it appears 0, 1, 2 times, and at least twice.
2. Deduce the expected number of *distinct* examples in the sample. What fraction of the data does a bootstrap sample contain when $m = 1000$?

Exercise 12 – Gini impurity and feature importance in a tree

A classification tree of a random forest was grown on 100 examples (50 positive, 50 negative):

node	split	examples (pos, neg)	children
root	$x_1 \le a$	100 $(50, 50)$	A, B
A	$x_2 \le b$	60 $(45, 15)$	A_1: 40 $(38, 2)$, A_2: 20 $(7, 13)$
B	$x_2 \le c$	40 $(5, 35)$	B_1: 10 $(5, 5)$, B_2: 30 $(0, 30)$

The Gini impurity of a node with class proportions $(p, 1-p)$ is $G = 1 - p^2 - (1-p)^2 = 2p(1-p)$.

1. Compute the Gini impurity of every node.
2. The decrease of a split is $\frac{n_{\text{node}}}{n} G_{\text{node}} - \frac{n_{\text{left}}}{n} G_{\text{left}} - \frac{n_{\text{right}}}{n} G_{\text{right}}$ with $n = 100$. Compute it for the three splits.
3. The (mean decrease in impurity) importance of a feature is the sum of the decreases of the splits that use it. Compute the importances of x_1 and x_2, then normalize them to sum to 1. Which feature is the most important in this tree?

Exercise 13 – Out-of-bag error and permutation importance

A forest of 5 trees, 6 training examples with labels $y = (+1, +1, -1, -1, +1, -1)$. The table gives the prediction of each tree on each example; a star marks the examples that are out-of-bag for that tree.

	x_1	x_2	x_3	x_4	x_5	x_6
tree 1	$+1$	$+1^\star$	-1	$+1^\star$	$+1$	-1
tree 2	$+1^\star$	-1	-1	$-1^\star$	$+1$	$+1^\star$
tree 3	-1	$+1$	$-1^\star$	$-1^\star$	$-1^\star$	-1
tree 4	$+1^\star$	$+1$	$+1$	-1	$+1$	$-1^\star$
tree 5	$+1^\star$	$+1^\star$	$-1^\star$	-1	$-1^\star$	$-1^\star$

1. Compute the prediction of the whole forest (majority of the 5 trees) on each example and its training error.
2. Compute the OOB prediction of each example (majority of the trees for which it is out-of-bag) and the OOB error. Compare with the training error.
3. On a larger dataset the OOB error of a forest is 0.10. After randomly permuting the values of x_1 (resp. x_2, x_3) in the OOB examples, it becomes 0.30 (resp. 0.12, 0.10). Give the permutation importance of each feature and interpret.

Exercise 14 – Multiclass AdaBoost (SAMME)

With K classes, SAMME uses the weight $\alpha_t = \ln \frac{1-\varepsilon_t}{\varepsilon_t} + \ln(K-1)$ and keeps a weak learner as long as $\alpha_t > 0$.

1. Compute α for $(K, \varepsilon) = (2, 0.3)$, $(3, 0.3)$, $(3, 0.5)$, $(3, \frac{2}{3})$, $(10, 0.8)$. Compare $(2, 0.3)$ with the binary AdaBoost value $\frac{1}{2} \ln \frac{1-\varepsilon}{\varepsilon}$.
2. Show that $\alpha_t > 0 \iff \varepsilon_t < 1 - \frac{1}{K}$. Interpret this condition (what is the error of a random guess?).

Exercise 15 – One round of XGBoost with two features

Six examples, labels $y \in \{0, 1\}$, logistic loss, initial score $F = 0$ (so $p_i = 0.5$), $\lambda = 1$, $\gamma = 0$:

x_1	1	2	3	4	5	6
x_2	3	1	2	6	5	4
y	1	0	0	1	1	1

1. Compute $g_i = p_i - y_i$, $h_i = p_i(1 - p_i)$, G and H.
2. Compute the gain $\frac{1}{2}\left[\frac{G_L^2}{H_L+\lambda} + \frac{G_R^2}{H_R+\lambda} - \frac{G^2}{H+\lambda}\right]$ of the splits $x_1 < \theta$ and $x_2 < \theta$ for $\theta \in \{1.5, \dots, 5.5\}$. Which split is chosen?
3. Compute the leaf values $w^\star = -\frac{G}{H+\lambda}$, then the new scores, probabilities and log-loss with learning rate $\eta = 0.3$ and $\eta = 1$ (initial log-loss $\ln 2 = 0.693$).

Exercise 16 – Pseudo-residuals of the Huber loss

The Huber loss with threshold δ is $\ell(y, F) = \frac{1}{2}(y - F)^2$ if $|y - F| \le \delta$ and $\delta|y - F| - \frac{1}{2}\delta^2$ otherwise.

1. Show that its pseudo-residual is $r = y - F$ if $|y - F| \le \delta$ and $\delta \operatorname{sign}(y - F)$ otherwise.
2. Data $y = (1, 2, 2.5, 3, 10)$, current prediction $F = 2.5$ everywhere (the median), $\delta = 1$. Compute the residuals, the pseudo-residuals of the squared loss and of the Huber loss.
3. Compare the influence of the last point on the next tree in both cases.

3 PAC-Bayesian learning

Exercise 17 – Gibbs risk, Bayes risk and the KL to the prior

Five equally likely examples and four voters; a 1 indicates an error:

	x_1	x_2	x_3	x_4	x_5
h_1	1	0	0	0	0
h_2	1	1	0	0	0
h_3	0	0	1	0	0
h_4	0	0	0	1	0

1. Give the risk of each voter. For Q uniform, compute $W_Q(x_j)$ on each example, the Gibbs risk and the Bayes risk (ties count as errors). Check $R_{\mathcal{D}}(B_Q) \leq 2R_{\mathcal{D}}(G_Q)$.
2. Same with $Q = (0.4, 0.1, 0.25, 0.25)$. Which posterior is better for the Gibbs classifier? For the vote?
3. With the uniform prior P, compute $\mathrm{KL}(Q\|P)$ for both posteriors. Which one "costs" more in a PAC-Bayesian bound?
4. Explain why the vote is better than the Gibbs classifier here, using the distribution of W_Q.

Exercise 18 – Computing PAC-Bayesian certificates

A uniform prior P on $n = 100$ voters, $m = 1000$ examples, $\delta = 0.05$, so $\ln \frac{2\sqrt{m}}{\delta} \simeq 7.14$. Three posteriors are considered: uniform on all the voters (empirical Gibbs risk 0.25), uniform on the 10 best (0.15), a Dirac on the best voter (0.12).

1. Compute $\mathrm{KL}(Q\|P)$ for each posterior.
2. Compute McAllester's bound $\widehat{R}_S(G_Q) + \sqrt{\frac{\mathrm{KL}+7.14}{2m}}$ for each. Which posterior has the best certificate?
3. Seeger's bound is $\overline{\mathrm{kl}}^{-1}(\widehat{R}_S(G_Q), \varepsilon)$ with $\varepsilon = \frac{\mathrm{KL}+7.14}{m}$. For the Dirac, run four steps of the bisection on $[0.12, 1]$ (compute $\mathrm{kl}(0.12\|p)$ at each midpoint; the values are 0.425, 0.128, 0.039, 0.0115 at $p = 0.56, 0.34, 0.23, 0.175$). Give the interval obtained. (The exact value is 0.176.)
4. Why is the bound valid for the posterior we chose *after* looking at the data? What would make it invalid?

Exercise 19 – The Gibbs posterior minimizes the linear bound

The linear PAC-Bayesian bound states that, for $\lambda > 0$ fixed in advance, with probability $\geq 1 - \delta$, for all Q:

$$R_{\mathcal{D}}(G_Q) \leq \widehat{R}_S(G_Q) + \frac{\mathrm{KL}(Q\|P) + \ln \frac{1}{\delta}}{\lambda} + \frac{\lambda}{8m}.$$

1. Show that $\lambda \widehat{R}_S(G_Q) + \mathrm{KL}(Q\|P) = \mathrm{KL}(Q\|Q_\lambda) - \ln Z_\lambda$, where $Q_\lambda(h) = P(h)e^{-\lambda \widehat{R}_S(h)}/Z_\lambda$. Deduce that Q_λ minimizes the bound.
2. Three voters with empirical risks $(0.10, 0.15, 0.30)$, uniform prior. Compute Q_λ for $\lambda = 20$ and $\lambda = 100$, and the limits $\lambda \to 0$ and $\lambda \to \infty$. Interpret λ.

3. For $\lambda = 20$, compute $\widehat{R}_S(G_{Q_\lambda})$ and $\mathrm{KL}(Q_\lambda \| P)$ (answers: 0.116 and 0.454). With $m = 500$, $\delta = 0.05$ and the bound evaluated at $\lambda = 100$, compare the bound for the posteriors Q_{20}, Q_{40}, Q_{100} (their values are 0.175, 0.168, 0.166). Why is Q_{100} the best here?
4. Why can λ not be chosen freely after seeing the data in this bound, and how is this solved in practice?

Exercise 20 – Numerical PAC-Bayes: sample sizes, Gaussian posteriors and forests

$\delta = 0.05$ throughout.

1. **Sample size.** A posterior has $\mathrm{KL}(Q\|P) = 5$. How many examples are needed for the complexity term of McAllester's bound, $\sqrt{\frac{\mathrm{KL} + \ln(2\sqrt{m}/\delta)}{2m}}$, to be at most 0.05? (Evaluate it for $m = 1000, 2000, 3000$, then refine.)
2. **Gaussian posterior on linear classifiers.** $Q = \mathcal{N}(\mathbf{w}, I)$ with $\mathbf{w} = (2, -1, 2)$; the prior is $\mathcal{N}(\mathbf{w}_0, I)$ with $\mathbf{w}_0 = (1, -1, 1)$, learned on a separate part of the data. On $m = 2000$ examples the empirical Gibbs risk (probit loss) is 0.08; $\ln \frac{2\sqrt{2000}}{0.05} \simeq 7.49$. Compute $\mathrm{KL}(Q\|P)$, McAllester's bound on the Gibbs risk, and a bound on the risk of the linear classifier $\mathrm{sign}\langle \mathbf{w}, x \rangle$ (Seeger's bound gives 0.107). Same McAllester bound with the prior $\mathcal{N}(\mathbf{0}, I)$.
3. **A forest without a test set.** The uniform posterior on the trees of a forest ($\mathrm{KL} = 0$) has an out-of-bag Gibbs risk 0.18, each tree being evaluated on at least $m = 600$ out-of-bag examples ($\ln \frac{2\sqrt{600}}{0.05} \simeq 6.89$). Seeger's bound on the Gibbs risk is 0.243. Deduce a bound on the risk of the forest. Is it informative, knowing that the forest has a test error around 0.05? Why?