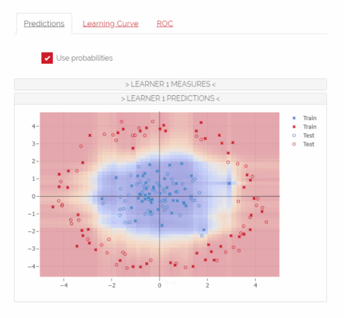
Introduction to Machine Learning

Gradient Boosting for Classification



Learning goals

- GB for binary classification simply uses Bernoulli or exponential loss
- For multiclass we fit g discriminant functions in parallel

BINARY CLASSIFICATION

For $\mathcal{Y} = \{0, 1\}$, we simply have to select an appropriate loss function, so let us use Bernoulli loss as in logistic regression:

$$L(y, f(\mathbf{x})) = -y \cdot f(\mathbf{x}) + \log(1 + \exp(f(\mathbf{x}))).$$

Then,

$$\begin{aligned}
\tilde{r}(f) &= -\frac{\partial L(y, f(\mathbf{x}))}{\partial f(\mathbf{x})} \\
&= y - \frac{\exp(f(\mathbf{x}))}{1 + \exp(f(\mathbf{x}))} \\
&= y - \frac{1}{1 + \exp(-f(\mathbf{x}))} = y - s(f(\mathbf{x})).
\end{aligned}$$

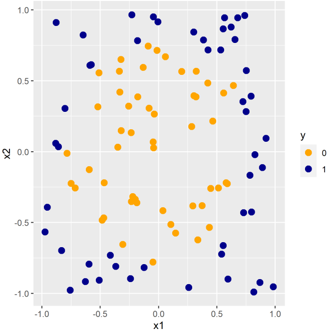
Here, $s(f(\mathbf{x}))$ is the logistic function, applied to a scoring model. Hence, effectively, the pseudo-residuals are $y - \pi(\mathbf{x})$.
Through $\pi(\mathbf{x}) = s(f(\mathbf{x}))$ we can also estimate posterior probabilities.

BINARY CLASSIFICATION

- Rest works as in regression.
- NB: We fit regression BLs against the PRs with $L2$ loss.
- Exponential loss works too. In practice there is no big difference, although Bernoulli loss makes a bit more sense from a theoretical (maximum likelihood) perspective.
- It can be shown GB with exp loss is basically equivalent to and generalizes AdaBoost.

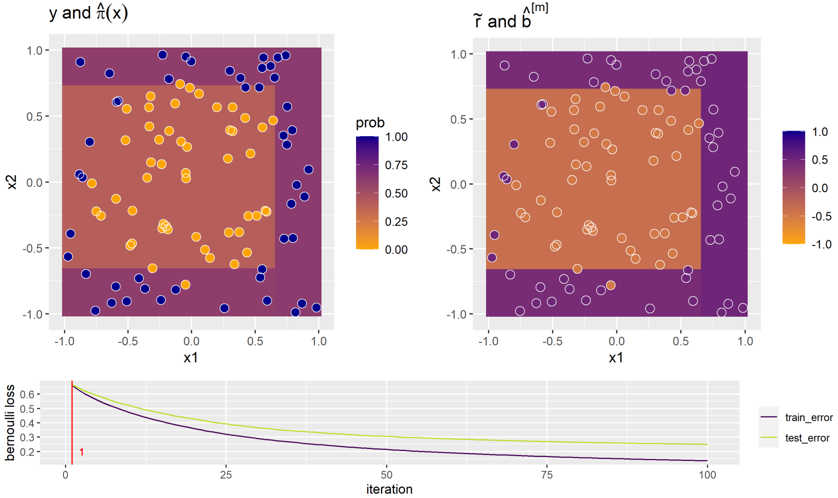
EXAMPLE: 2D CIRCLE DATA

- `mlbench` `circle` data with $n = 100$
- Bernoulli loss
- BL = shallow tree with max. depth of 3
- We initialized with $f^{[0]} = 0$.



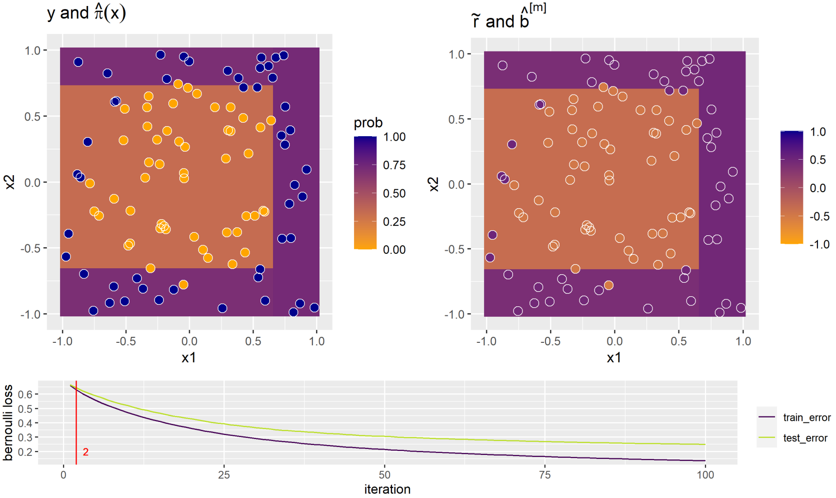
EXAMPLE: 2D CIRCLE DATA

BG color is predicted probs on LHS on RHS we show and preds of BL.



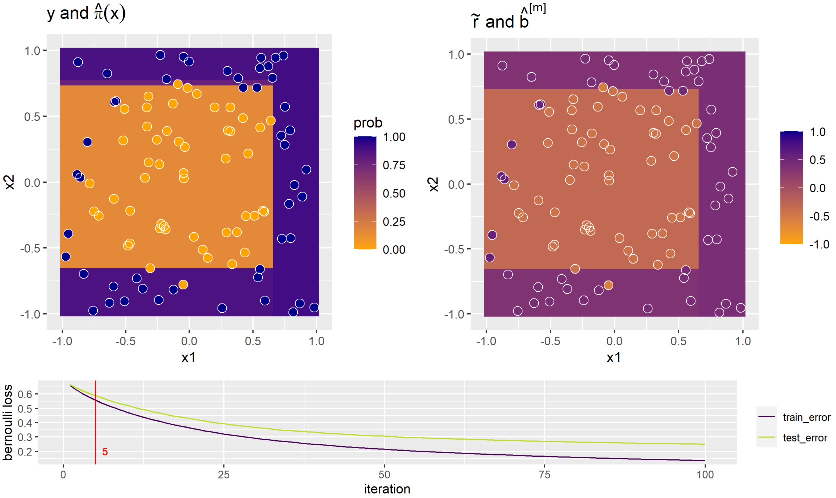
EXAMPLE: 2D CIRCLE DATA

BG color is predicted probs on LHS on RHS we show and preds of BL.



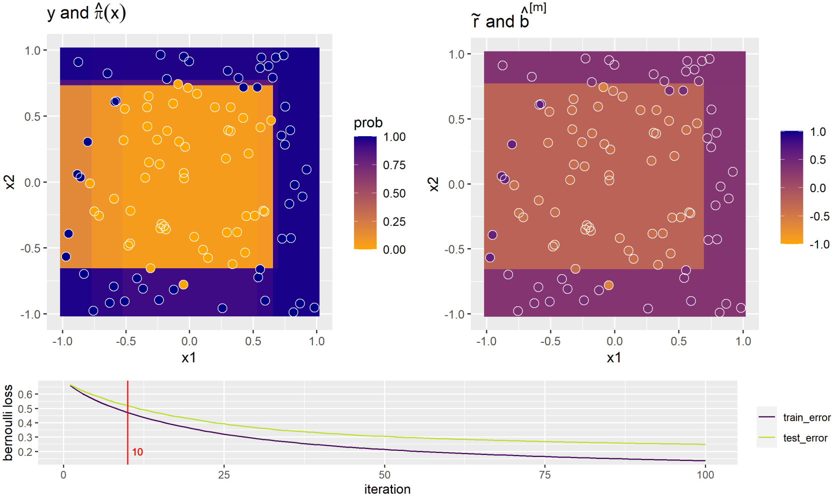
EXAMPLE: 2D CIRCLE DATA

BG color is predicted probs on LHS on RHS we show and preds of BL.



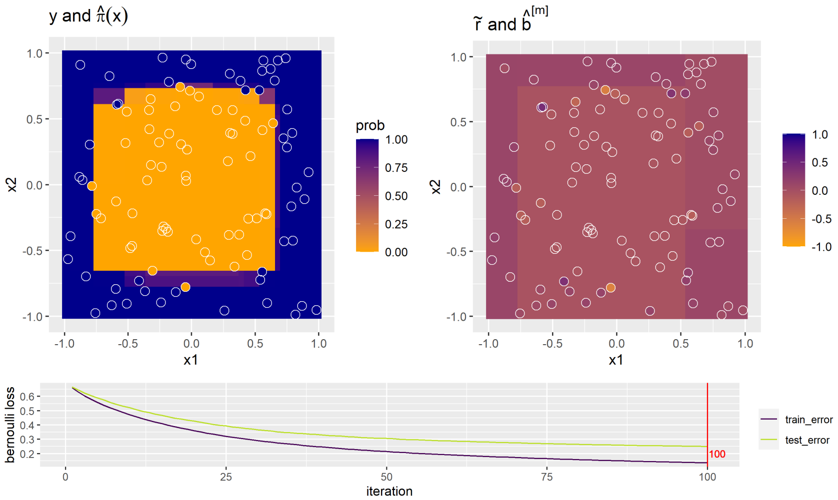
EXAMPLE: 2D CIRCLE DATA

BG color is predicted probs on LHS on RHS we show and preds of BL.



EXAMPLE: 2D CIRCLE DATA

BG color is predicted probs on LHS on RHS we show and preds of BL.



MULTICLASS PROBLEMS

We proceed as in softmax regression and model a categorical distribution with multinomial / log loss. For $\mathcal{Y} = \{1, \dots, g\}$, we create g discriminant functions $f_k(\mathbf{x})$, one for each class and each one being an **additive** model of base learners.

We define the $\pi_k(\mathbf{x})$ through the softmax function:

$$\pi_k(\mathbf{x}) = s_k(f_1(\mathbf{x}), \dots, f_g(\mathbf{x})) = \exp(f_k(\mathbf{x})) / \sum_{j=1}^{g} \exp(f_j(\mathbf{x})).$$

Multinomial loss L:

$$L(y, f_1(\mathbf{x}), \dots f_g(\mathbf{x})) = -\sum_{k=1}^{g} \mathbb{1}_{\{y=k\}} \ln \pi_k(\mathbf{x}).$$

Pseudo-residuals:

$$-\frac{\partial L(y, f_1(\mathbf{x}), \dots, f_g(\mathbf{x}))}{\partial f_k(\mathbf{x})} = \mathbb{1}_{\{y=k\}} - \pi_k(\mathbf{x}).$$

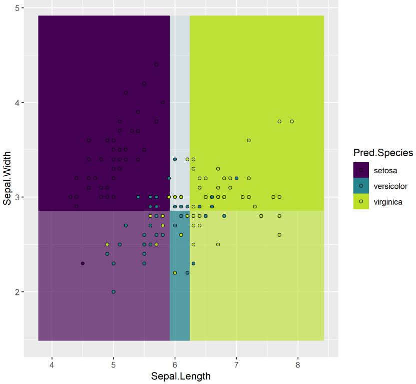
MULTICLASS PROBLEMS

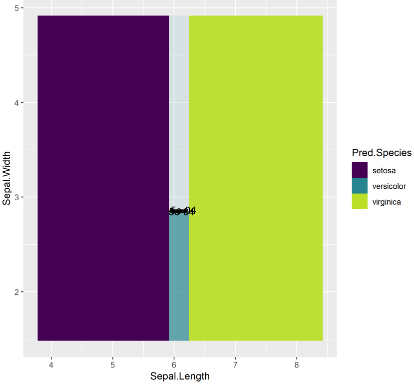
Algorithm 1 GB for Multiclass

Initialize $f_k^{[0]}(\mathbf{x}) = 0,\ k = 1, \dots, g$
for $m = 1 \to M$ **do**
    Set $\pi_k^{[m]}(\mathbf{x}) = \frac{\exp(f_k^{[m]}(\mathbf{x}))}{\sum_j \exp(f_j^{[m]}(\mathbf{x}))}, k = 1, \dots, g$
    for $k = 1 \to g$ **do**
        For all i: Compute $\tilde{r}_k^{[m](i)} = \mathbb{1}_{\{y^{(i)}=k\}} - \pi_k^{[m]}(\mathbf{x}^{(i)})$
        Fit a regression base learner $\hat{b}_k^{[m]}$ to the pseudo-residuals $\tilde{r}_k^{[m](i)}$.
        Update $\hat{f}_k^{[m]} = \hat{f}_k^{[m-1]} + \alpha \hat{b}_k^{[m]}$
    end for
end for
Output $\hat{f}_1^{[M]}, \dots, \hat{f}_g^{[M]}$

EXAMPLE: 2D IRIS

LHS: BG color is predicted probs and point col is true label; RHS: Contour lines of discriminant functions.

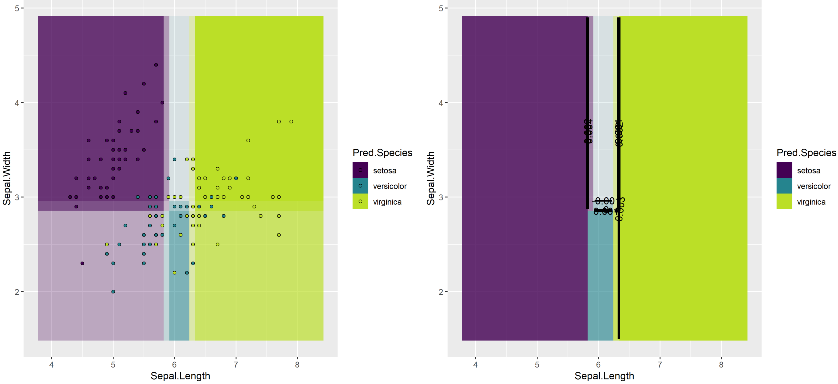




Iteration=1

EXAMPLE: 2D IRIS

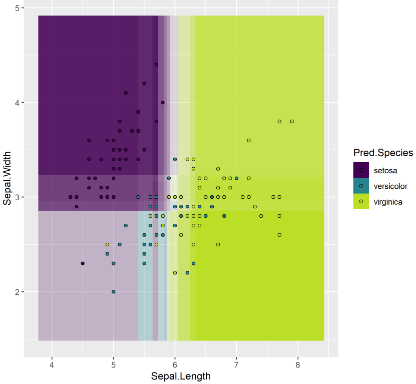
LHS: BG color is predicted probs and point col is true label; RHS: Contour lines of discriminant functions.

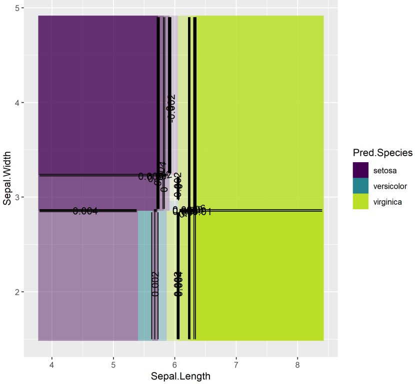


Iteration=2

EXAMPLE: 2D IRIS

LHS: BG color is predicted probs and point col is true label; RHS: Contour lines of discriminant functions.

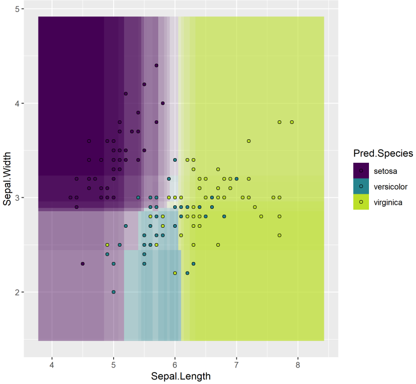


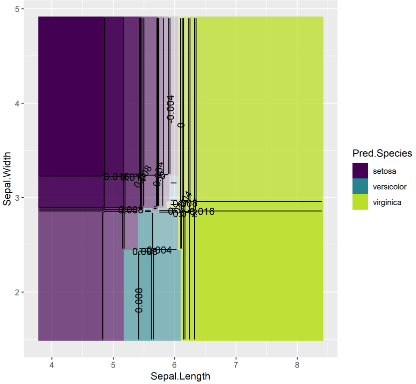


Iteration=5

EXAMPLE: 2D IRIS

LHS: BG color is predicted probs and point col is true label; RHS: Contour lines of discriminant functions.

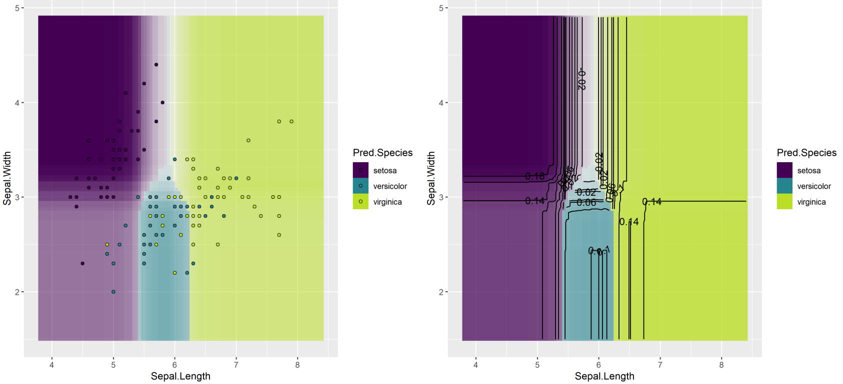




Iteration=10

EXAMPLE: 2D IRIS

LHS: BG color is predicted probs and point col is true label; RHS: Contour lines of discriminant functions.



Iteration=100