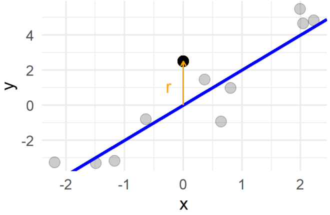
Introduction to Machine Learning

Pseudo-Residuals and Gradient Descent



Learning goals

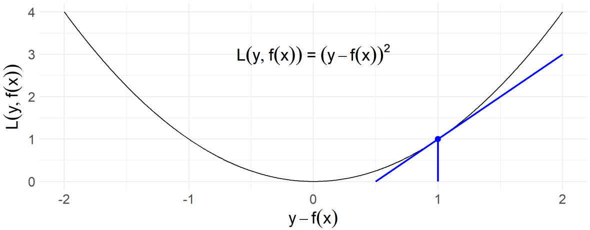
- Know the concept of pseudo-residuals
- Understand the relationship between pseudo-residuals and gradient descent

PSEUDO-RESIDUALS

- In regression, residuals are defined as $r := y - f(\mathbf{x})$.
- We further define **pseudo-residuals** as the negative first derivatives of loss functions w.r.t. $f(\mathbf{x})$

$$\tilde{r} \quad := \quad -\frac{\partial L(y, f(\mathbf{x}))}{\partial f(\mathbf{x})}.$$

- This definition also holds for score / probability based classifiers.
- Note that $\tilde{r}$ depends on y and $f(\mathbf{x})$ and L.

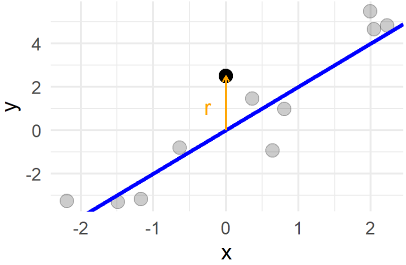


BEST POINT-WISE UPDATE

Assume we have (partially) fitted a model $f(\mathbf{x})$ to data $\mathcal{D}$.

Assume we could update $f(\mathbf{x})$ point-wise as we like. For a fixed $\mathbf{x} \in \mathcal{X}$, the best point-wise update is the direction of the residual $r = y - f(\mathbf{x})$

$$f(\mathbf{x}) \leftarrow f(\mathbf{x}) + r$$



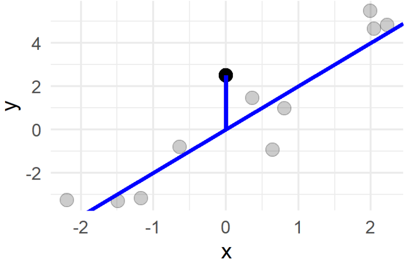
BEST POINT-WISE UPDATE

Assume we have (partially) fitted a model $f(\mathbf{x})$ to data $\mathcal{D}$.

Assume we could update $f(\mathbf{x})$ point-wise as we like. For a fixed $\mathbf{x} \in \mathcal{X}$, the best point-wise update is the direction of the residual $r = y - f(\mathbf{x})$

$$f(\mathbf{x}) \leftarrow f(\mathbf{x}) + r$$

The point-wise error at this specific $\mathbf{x}$ becomes 0.



APPROXIMATE BEST POINT-WISE UPDATE

When applying gradient descent to compute a point-wise update of $f(\mathbf{x})$, we would go a step into the direction of the negative gradient

$$f(\mathbf{x}) \leftarrow f(\mathbf{x}) - \frac{\partial L\left(y, f(\mathbf{x})\right)}{\partial f(\mathbf{x})}.$$

which is the direction of the pseudo-residual

$$f(\mathbf{x}) \quad \leftarrow \quad f(\mathbf{x}) + \tilde{r}$$

Iteratively stepping towards the direction of the pseudo-residuals is the underlying idea of gradient boosting, which is a learning algorithm that will be covered in a later chapter.

GD IN ML AND PSEUDO-RESIDUALS

- In GD, we move in the direction of the negative gradient by updating the parameters:

$$\boldsymbol{\theta}^{[t+1]} = \boldsymbol{\theta}^{[t]} - \alpha^{[t]} \cdot \nabla_{\boldsymbol{\theta}} \left.\mathcal{R}_{\text{emp}}(\boldsymbol{\theta})\right|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{[t]}}$$

- This can be seen as approximating the unexplained information (measured by the loss) through a model update.
- Using the chain rule:

$$\begin{aligned}
\nabla_{\boldsymbol{\theta}} \mathcal{R}_{\text{emp}}(\boldsymbol{\theta}) &= \sum_{i=1}^{n} \left.\frac{\partial L\left(y^{(i)}, f\right)}{\partial f}\right|_{f=f\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right)} \cdot \nabla_{\boldsymbol{\theta}} f\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right) \\
&= -\sum_{i=1}^{n} \tilde{r}^{(i)} \cdot \nabla_{\boldsymbol{\theta}} f\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right).
\end{aligned}$$

- Hence the update is determined by a loss-optimal directional change of the model output and a loss-independent derivate of f. This is a very flexible, nearly loss-independent formulation of GD.