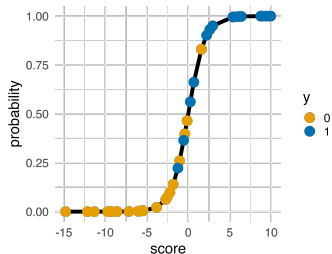
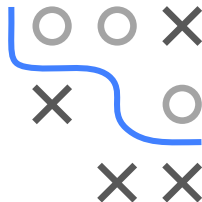


# Introduction to Machine Learning

## Classification

## Logistic Regression



### Learning goals

- Hypothesis space of LR
- Log-Loss derivation
- Intuition for loss
- LR as linear classifier

# MOTIVATION

- Let's build a **discriminant** approach, for binary classification, as a probabilistic classifier  $\pi(\mathbf{x} \mid \theta)$
- We encode  $y \in \{0, 1\}$  and use ERM:

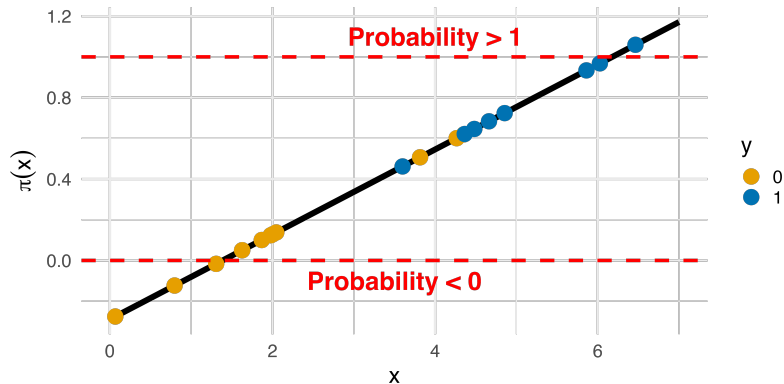
$$\arg \min_{\theta \in \Theta} \mathcal{R}_{\text{emp}}(\theta) = \arg \min_{\theta \in \Theta} \sum_{i=1}^n L\left(y^{(i)}, \pi\left(\mathbf{x}^{(i)} \mid \theta\right)\right)$$

- We want to “copy” over ideas from linear regression
- In the above, our model structure should be “mainly” linear and we need a loss function



# DIRECT LINEAR MODEL FOR PROBABILITIES

We could directly use an LM to model  $\pi(\mathbf{x} \mid \theta) = \theta^\top \mathbf{x}$ .  
And use L2 loss in ERM.

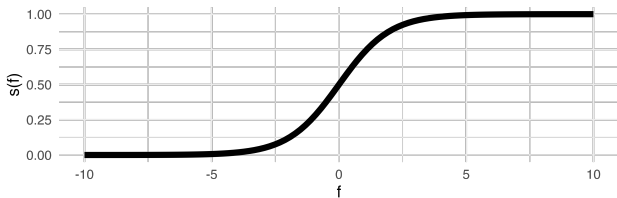


But: This obviously will result in predicted probabilities  $\pi(\mathbf{x} \mid \theta) \notin [0, 1]!$

# HYPOTHESIS SPACE OF LR

To avoid this, logistic regression “squashes” the estimated linear scores  $\theta^\top \mathbf{x}$  to  $[0, 1]$  through the **logistic function**  $s$ :

$$\pi(\mathbf{x} \mid \theta) = \frac{\exp(\theta^\top \mathbf{x})}{1 + \exp(\theta^\top \mathbf{x})} = \frac{1}{1 + \exp(-\theta^\top \mathbf{x})} = s(\theta^\top \mathbf{x}) = s(f(\mathbf{x}))$$

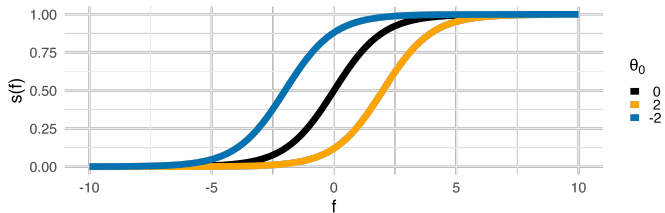


⇒ **Hypothesis space of LR:**

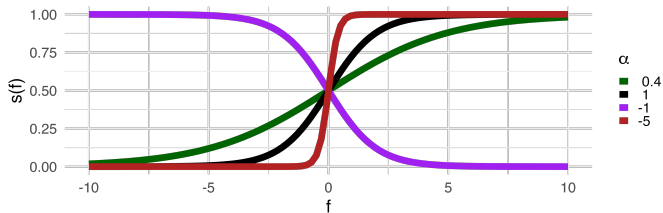
$$\mathcal{H} = \left\{ \pi : \mathcal{X} \rightarrow [0, 1] \mid \pi(\mathbf{x} \mid \theta) = s(\theta^\top \mathbf{x}) \mid \theta \in \mathbb{R}^{p+1} \right\}$$

# LOGISTIC FUNCTION

Intercept  $\theta_0$  shifts  $\pi = s(\theta_0 + f) = \frac{\exp(\theta_0 + f)}{1 + \exp(\theta_0 + f)}$  horizontally

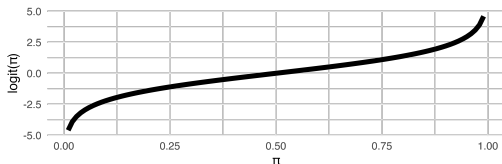


Scaling  $f$  like  $s(\alpha f) = \frac{\exp(\alpha f)}{1 + \exp(\alpha f)}$  controls slope and direction



# THE LOGIT

The inverse  $s^{-1}(\pi) = \log\left(\frac{\pi}{1-\pi}\right)$  where  $\pi$  is a probability is called **logit** (also called **log odds** since it is equal to the logarithm of the odds  $\frac{\pi}{1-\pi}$ )



- Positive logits indicate probabilities  $> 0.5$  and vice versa
- E.g.: if  $p = 0.75$ , odds are 3 : 1 and logit is  $\log(3) \approx 1.1$
- Features  $\mathbf{x}$  act linearly on logits, controlled by coefficients  $\theta$ :

$$s^{-1}(\pi(\mathbf{x})) = \log\left(\frac{\pi(\mathbf{x})}{1 - \pi(\mathbf{x})}\right) = \theta^T \mathbf{x}$$

# DERIVING LOG-LOSS

We need to find a suitable loss function for **ERM**. We look at likelihood which multiplies up  $\pi(\mathbf{x}^{(i)} | \theta)$  for positive examples, and  $1 - \pi(\mathbf{x}^{(i)} | \theta)$  for negative.

$$\mathcal{L}(\theta) = \prod_{i \text{ with } y^{(i)}=1} \pi(\mathbf{x}^{(i)} | \theta) \prod_{i \text{ with } y^{(i)}=0} (1 - \pi(\mathbf{x}^{(i)} | \theta))$$

We can now cleverly combine the 2 cases by using exponents (note that only one of the 2 factors is not 1 and “active”):

$$\mathcal{L}(\theta) = \prod_{i=1}^n \pi(\mathbf{x}^{(i)} | \theta)^{y^{(i)}} (1 - \pi(\mathbf{x}^{(i)} | \theta))^{1-y^{(i)}}$$



## DERIVING LOG-LOSS / 2

Taking the log to convert products into sums:

$$\begin{aligned}\ell(\boldsymbol{\theta}) &= \log \mathcal{L}(\boldsymbol{\theta}) = \sum_{i=1}^n \log \left( \pi \left( \mathbf{x}^{(i)} \mid \boldsymbol{\theta} \right)^{y^{(i)}} \left( 1 - \pi \left( \mathbf{x}^{(i)} \mid \boldsymbol{\theta} \right) \right)^{1-y^{(i)}} \right) \\ &= \sum_{i=1}^n y^{(i)} \log \left( \pi \left( \mathbf{x}^{(i)} \mid \boldsymbol{\theta} \right) \right) + \left( 1 - y^{(i)} \right) \log \left( 1 - \pi \left( \mathbf{x}^{(i)} \mid \boldsymbol{\theta} \right) \right)\end{aligned}$$

Since we want to minimize the risk, we work with the negative  $\ell(\boldsymbol{\theta})$ :

$$-\ell(\boldsymbol{\theta}) = \sum_{i=1}^n -y^{(i)} \log \left( \pi \left( \mathbf{x}^{(i)} \mid \boldsymbol{\theta} \right) \right) - \left( 1 - y^{(i)} \right) \log \left( 1 - \pi \left( \mathbf{x}^{(i)} \mid \boldsymbol{\theta} \right) \right)$$



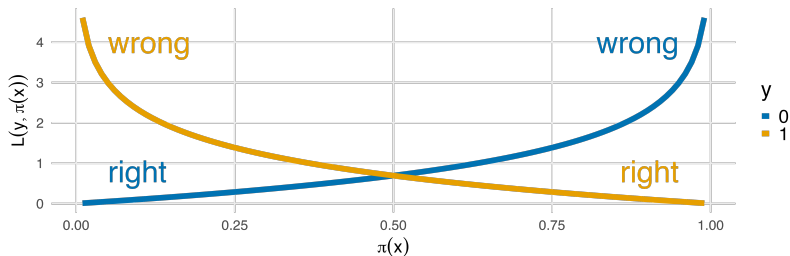


# BERNOULLI / LOG LOSS

The resulting loss

$$L(y, \pi) = -y \log(\pi) - (1 - y) \log(1 - \pi)$$

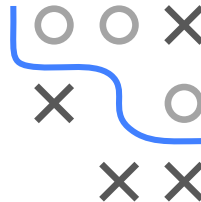
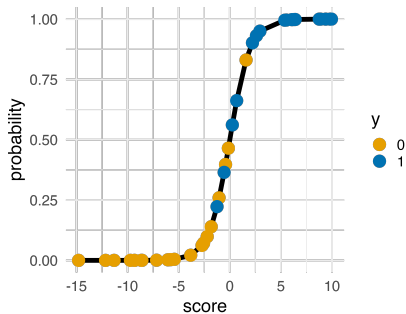
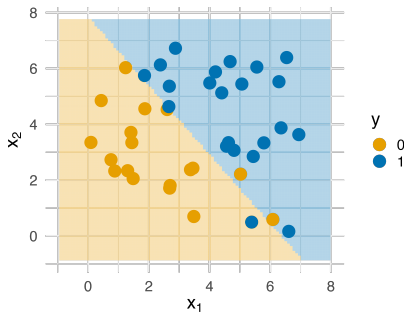
is called **Bernoulli**, **binomial**, **log** or **cross-entropy** loss



- Penalizes confidently wrong predictions heavily
- Is used for many other classifiers, e.g., in NNs or boosting

# LOGISTIC REGRESSION IN 2D

LR is a linear classifier, as  $\pi(\mathbf{x} \mid \theta) = s(\theta^\top \mathbf{x})$  and  $s$  is isotonic.



# OPTIMIZATION

- Log-Loss is convex, under regularity conditions LR has a unique solution (because of its linear structure), but not an analytical one
- To fit LR we use numerical optimization, e.g., Newton-Raphson
- If data is linearly separable, the optimization problem is unbounded and we would not find a solution; way out is regularization
- Why not use least squares on  $\pi(\mathbf{x}) = s(f(\mathbf{x}))$ ?  
Answer: ERM problem is not convex anymore :(
- We can also write the ERM as

$$\arg \min_{\theta \in \Theta} \mathcal{R}_{\text{emp}}(\theta) = \arg \min_{\theta \in \Theta} \sum_{i=1}^n L\left(y^{(i)}, f\left(\mathbf{x}^{(i)} \mid \theta\right)\right)$$

With  $f(\mathbf{x} \mid \theta) = \theta^T \mathbf{x}$  and  $L(y, f) = -yf + \log(1 + \exp(f))$

This combines the sigmoid with the loss and shows a convex loss directly on a linear function

