Overview

Classification basics + Thresholding

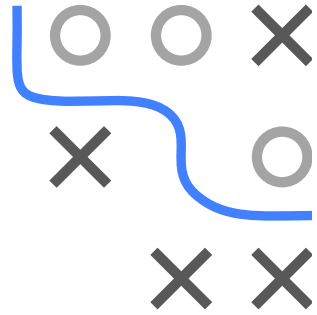
Logistic Regression

Sigmoid function
- What Are Odds?
- Why Use Log Odds?
- Modeling Log Odds with a Line
- Deriving the Sigmoid Function
- Conclusion

Classification basics

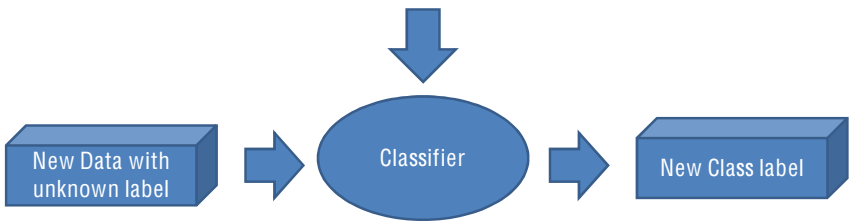
CLASSIFICATION

Learn functions that assign class labels to observation / feature vectors. Each observation belongs to exactly one class. The main difference to regression is that the target is categorical.



Our Data

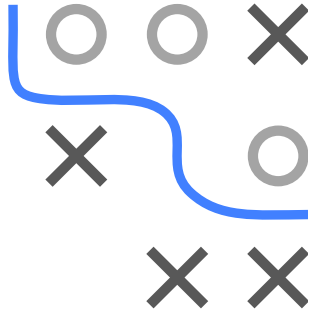
Sepal Length	Sepal Width	Petal Length	Petal Width	Species
5.1	3.5	1.4	0.2	setosa
5.9	3.0	5.1	1.8	virginica

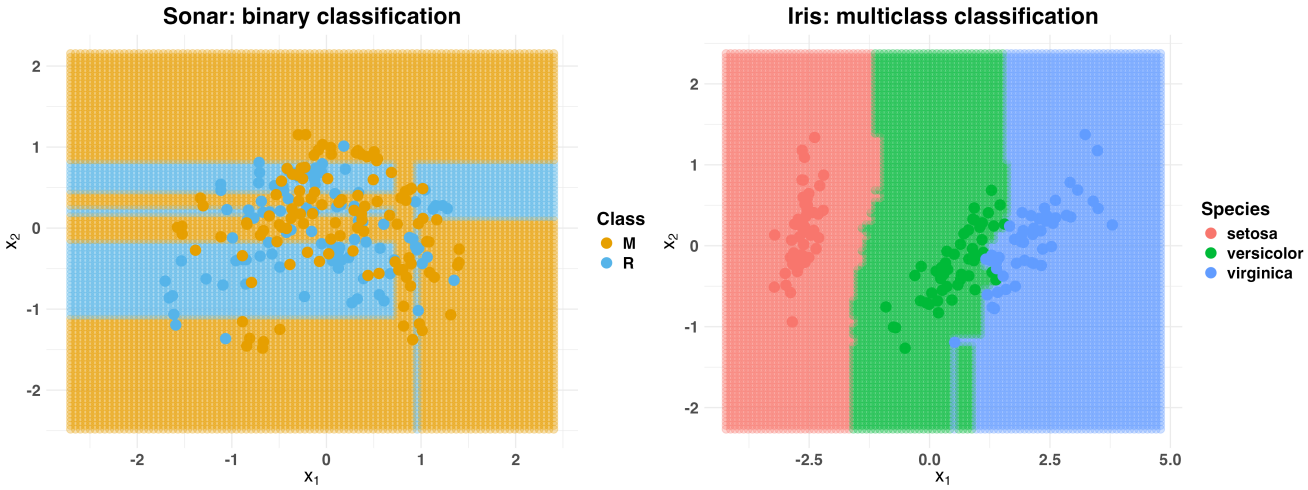


Sepal Length	Sepal Width	Petal Length	Petal Width	Species
5.4	3.3	3.2	1.1	???

BINARY AND MULTICLASS TASKS

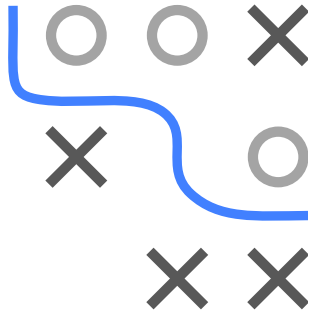
Tasks have a finite number of (unordered) classes.
They can be **binary** or **multiclass**.

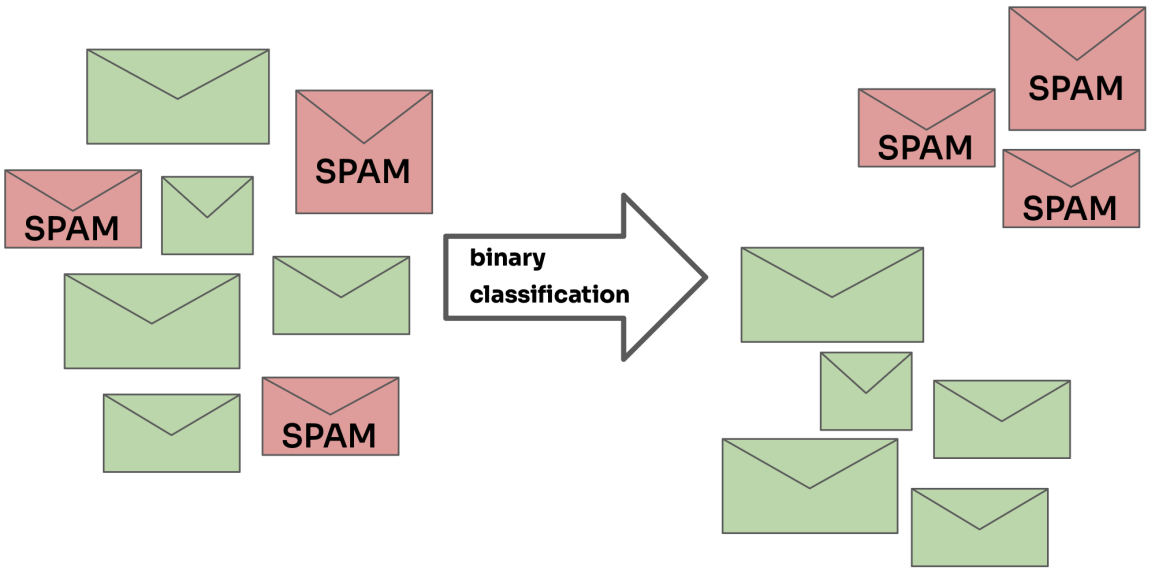




BINARY CLASSIFICATION TASK - EXAMPLES

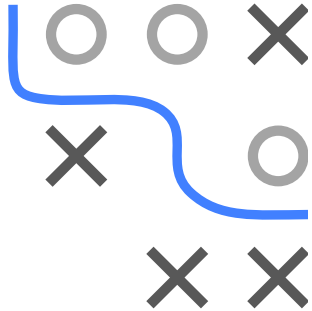
- Credit risk prediction, based on personal data and transactions
- Spam detection, based on textual features
- Churn prediction, based on customer behavior
- Predisposition for specific illness, based on genetic data





MULTICLASS TASK - IRIS

The iris dataset was introduced by the statistician Ronald Fisher and is one of the most frequent used data sets. Originally, it was designed for linear discriminant analysis.



Setosa

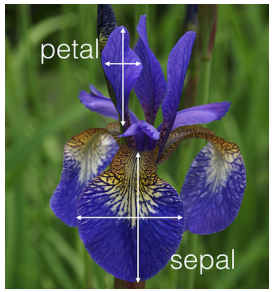
Versicolor

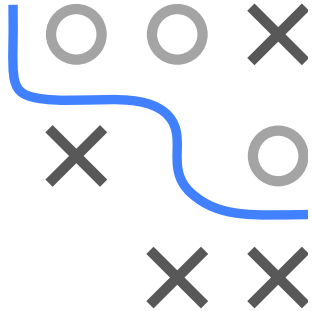
Virginica

Source:
`https://en.wikipedia.org/wiki/Iris_flower_data_set`

MULTICLASS TASK - IRIS

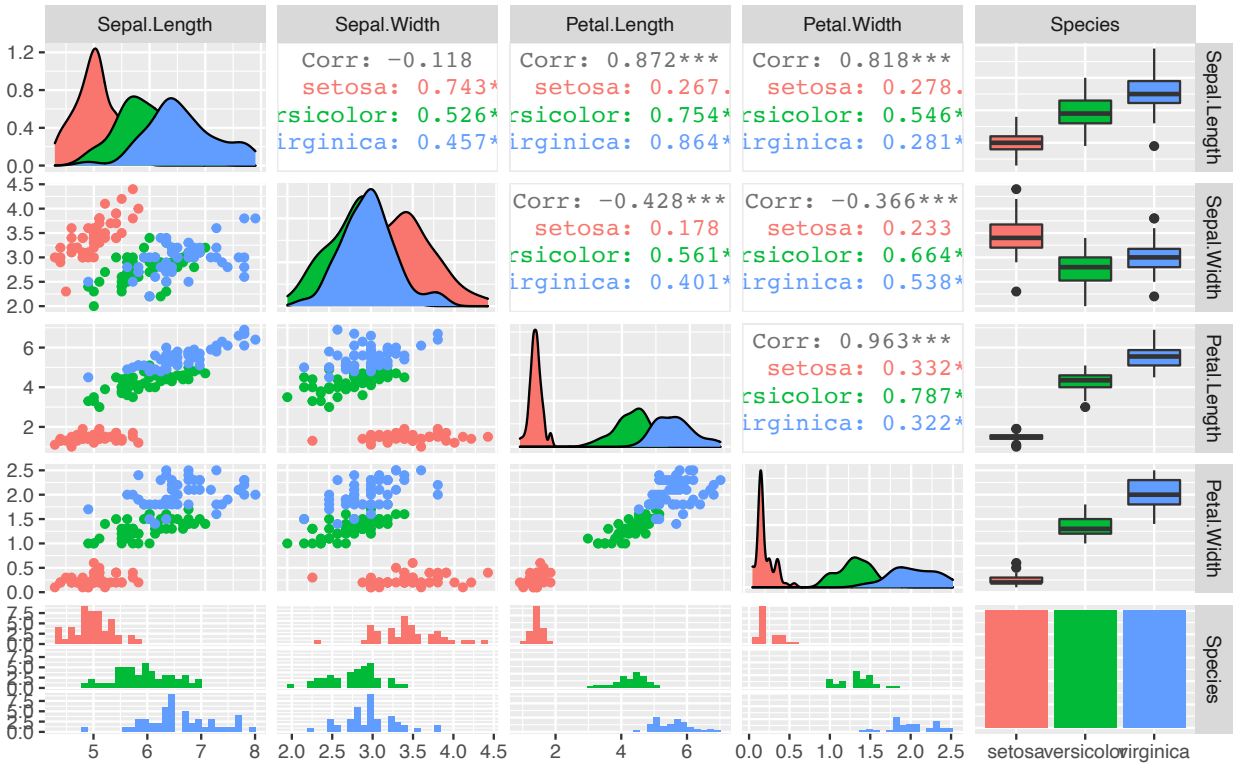
- 150 iris flowers
- Predict subspecies
- Based on sepal and petal length / width in [cm]

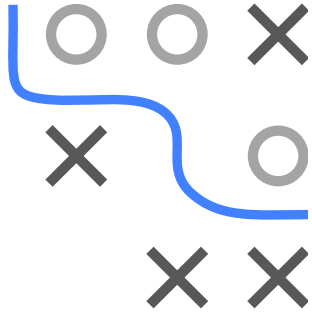




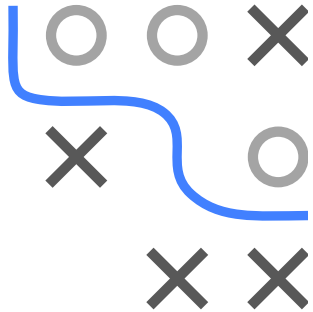
```
##      Sepal.Length Sepal.Width Petal.Length Petal.Width   Species
##   1:          5.1         3.5          1.4         0.2    setosa
##   2:          4.9         3.0          1.4         0.2    setosa
##   3:          4.7         3.2          1.3         0.2    setosa
##   4:          4.6         3.1          1.5         0.2    setosa
##   5:          5.0         3.6          1.4         0.2    setosa
##  ---
## 146:          6.7         3.0          5.2         2.3 virginica
## 147:          6.3         2.5          5.0         1.9 virginica
## 148:          6.5         3.0          5.2         2.0 virginica
## 149:          6.2         3.4          5.4         2.3 virginica
## 150:          5.9         3.0          5.1         1.8 virginica
```

MULTICLASS TASK - IRIS





PROBABILISTIC CLASSIFIERS



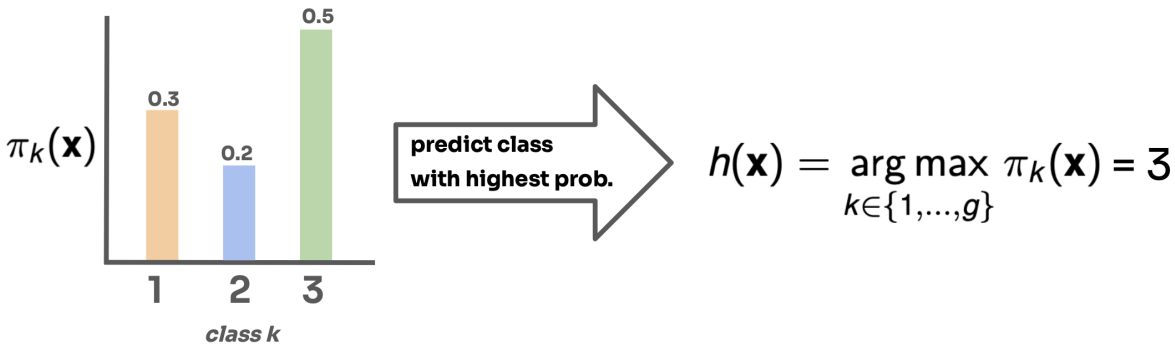
- Construct g **probability functions**

  $\pi_1, ..., \pi_g : \mathcal{X} \to [0, 1], \sum_{k=1}^{g} \pi_k(\mathbf{x}) = 1$

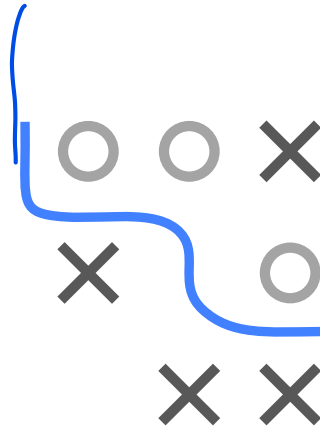
- Predicted class is usually the one with max probability

$$h(\mathbf{x}) = \underset{k \in \{1,...,g\}}{\arg\max}\ \pi_k(\mathbf{x})$$

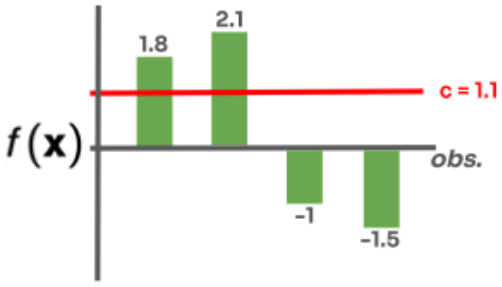
- For $g = 2$, single $\pi(\mathbf{x})$ is constructed, which models the predicted probability for the positive class (natural to encode $\mathcal{Y} = \{0, 1\}$)

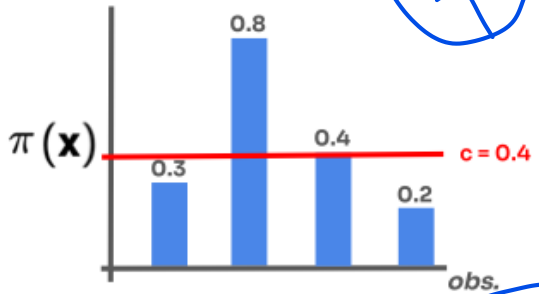


THRESHOLDING



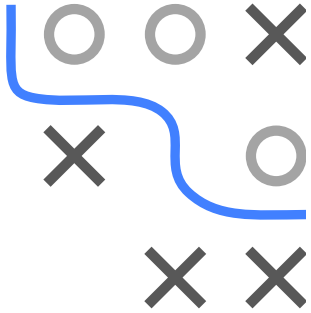
- For imbalanced cases or class with costs, we might want to deviate from the standard conversion of scores to classes
- Introduce basic concept (for binary case) and add details later
- Convert scores or probabilities to class outputs by thresholding: $h(\mathbf{x}) := [\pi(\mathbf{x}) \geq c]$ or $h(\mathbf{x}) := [f(\mathbf{x}) \geq c]$ for some threshold c
- Standard thresholds: $c = 0.5$ for probabilities, $c = 0$ for scores

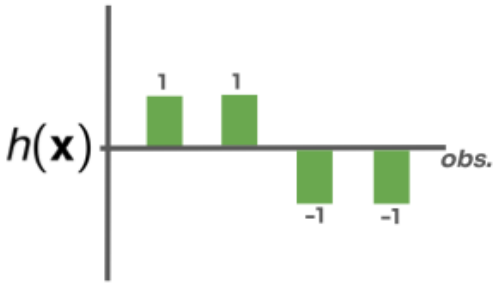


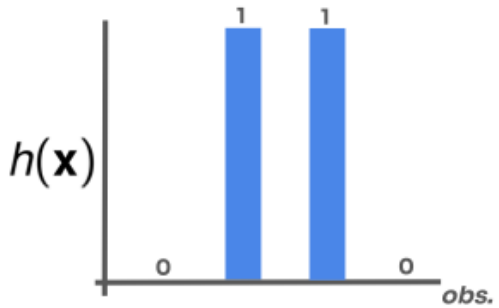


THRESHOLDING

- For imbalanced cases or class with costs, we might want to deviate from the standard conversion of scores to classes
- Introduce basic concept (for binary case) and add details later
- Convert scores or probabilities to class outputs by thresholding: $h(\mathbf{x}) := [\pi(\mathbf{x}) \geq c]$ or $h(\mathbf{x}) := [f(\mathbf{x}) \geq c]$ for some threshold c
- Standard thresholds: $c = 0.5$ for probabilities, $c = 0$ for scores

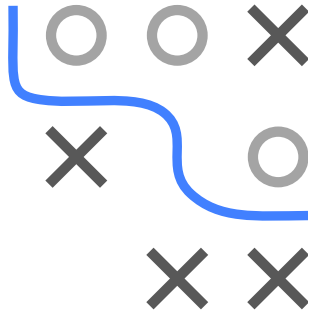




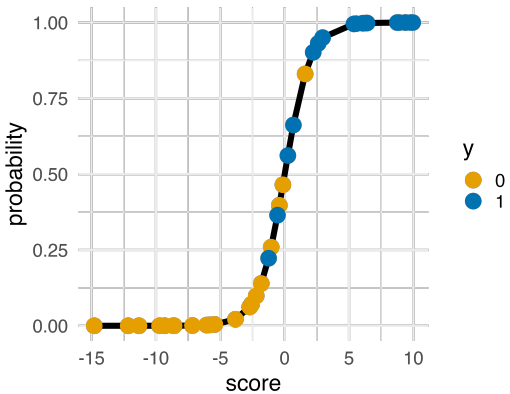


Logistic Regression

Introduction to Machine Learning



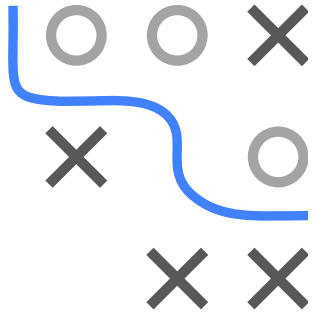
Classification
Logistic Regression



Learning goals

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

MOTIVATION



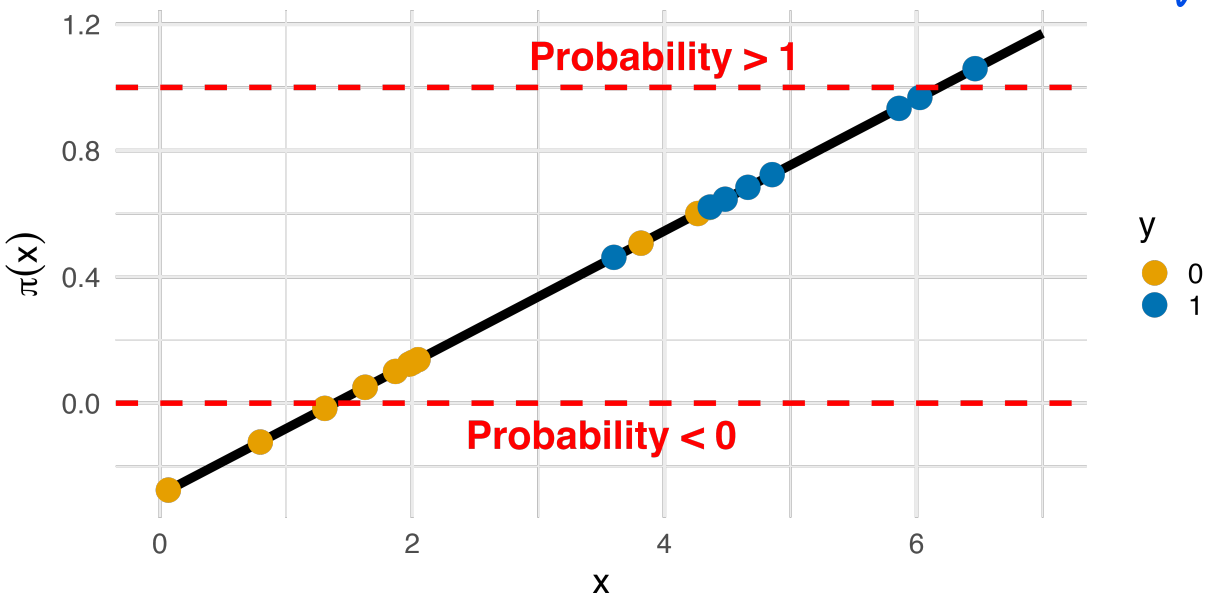
- Let's build a **discriminant** approach, for binary classification, as a probabilistic classifier $\pi(\mathbf{x} \mid \boldsymbol{\theta})$
- We encode $y \in \{0, 1\}$ and use ERM:

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

- 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 $\pi(\mathbf{x} \mid \boldsymbol{\theta}) = \boldsymbol{\theta}^\top \mathbf{x}$.
And use L2 loss in ERM.

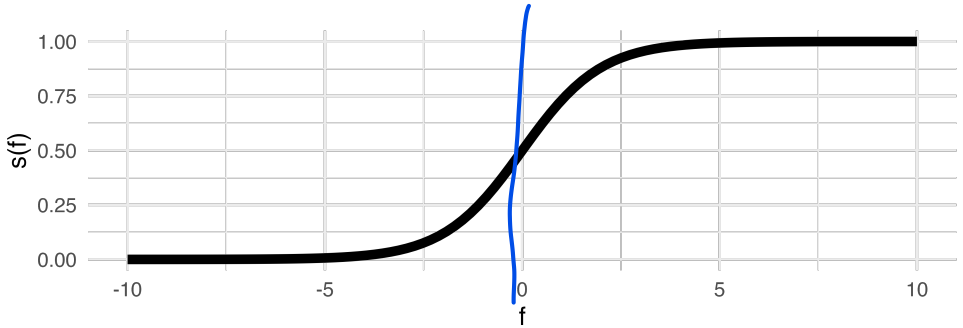


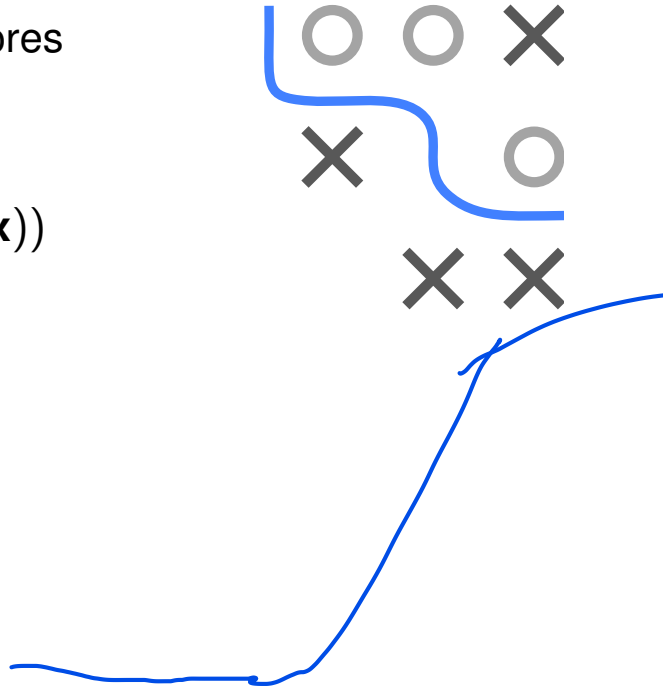
But: This obviously will result in predicted probabilities $\pi(\mathbf{x} \mid \boldsymbol{\theta}) \notin [0, 1]$!

HYPOTHESIS SPACE OF LR

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

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



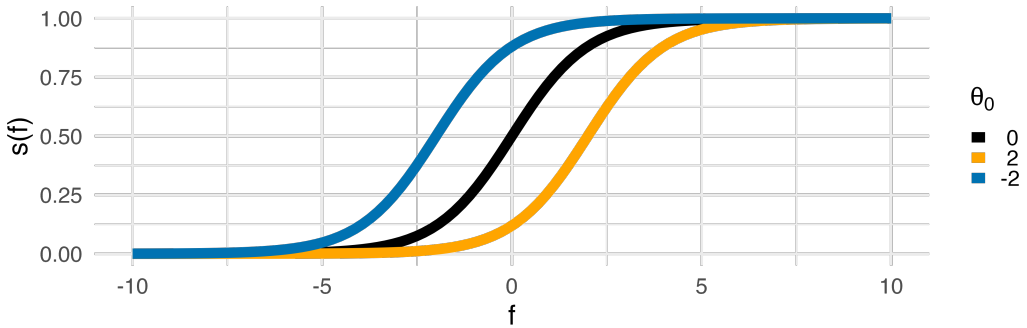


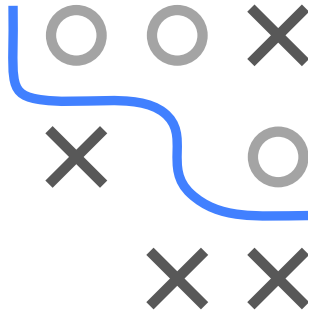
$\Rightarrow$ **Hypothesis space** of LR:

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

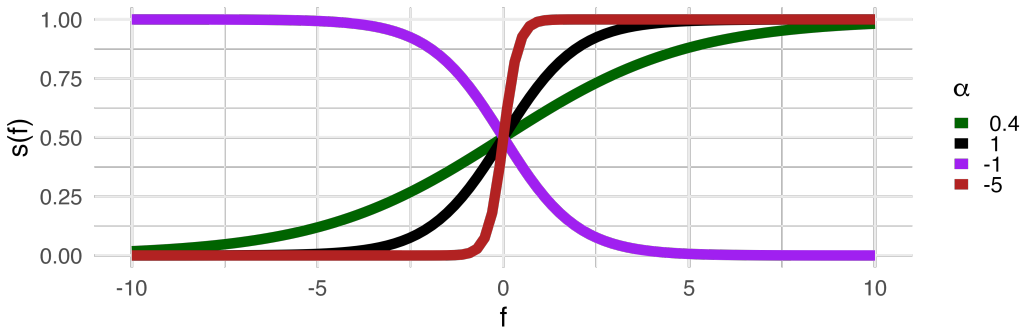
LOGISTIC FUNCTION

Intercept θ_0 shifts $\pi = s(\theta_0 + f) = \frac{\exp(\theta_0+f)}{1+\exp(\theta_0+f)}$ horizontally



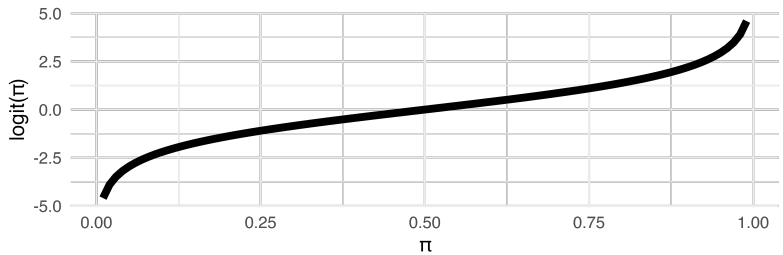


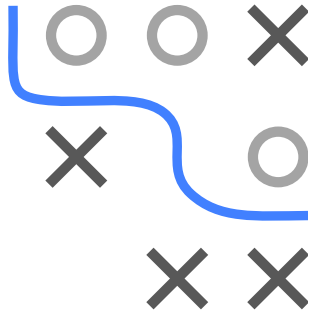
Scaling f like $s(\alpha f) = \frac{\exp(\alpha f)}{1+\exp(\alpha f)}$ controls slope and direction



THE LOGIT

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



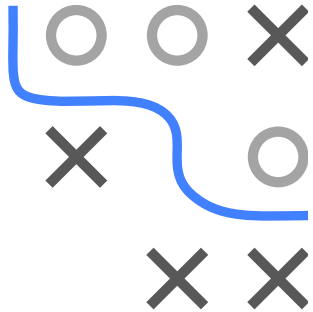


- Positive logits indicate probabilities > 0.5 and vice versa
- E.g.: if $p = 0.75$, odds are $3:1$ and logit is $log(3) \approx 1.1$
- Features $\mathbf{x}$ act linearly on logits, controlled by coefficients $\boldsymbol{\theta}$:

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

$$\text{Brier} = \left(y - \frac{1}{1+e^{-\theta^T x}}\right)^2$$

DERIVING LOG-LOSS



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

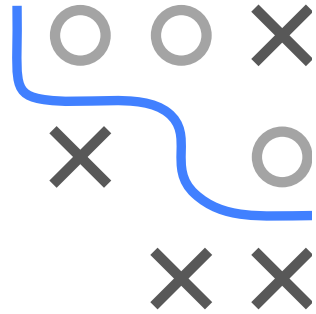
$$\mathcal{L}(\boldsymbol{\theta}) = \prod_{i \text{ with } y^{(i)}=1} \pi\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right) \prod_{i \text{ with } y^{(i)}=0} \left(1 - \pi\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right)\right)$$

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

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

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 $\ell(\boldsymbol{\theta})$:

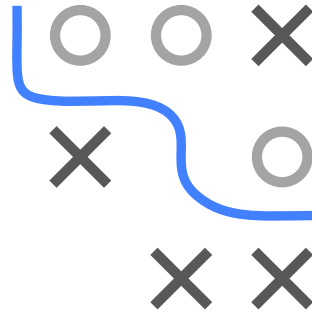
$$-\ell(\boldsymbol{\theta}) = \sum_{i=1}^{n} -y^{(i)} \log \left(\pi \left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta} \right) \right) - \left(1 - y^{(i)} \right) \log \left(1 - \pi \left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta} \right) \right)$$

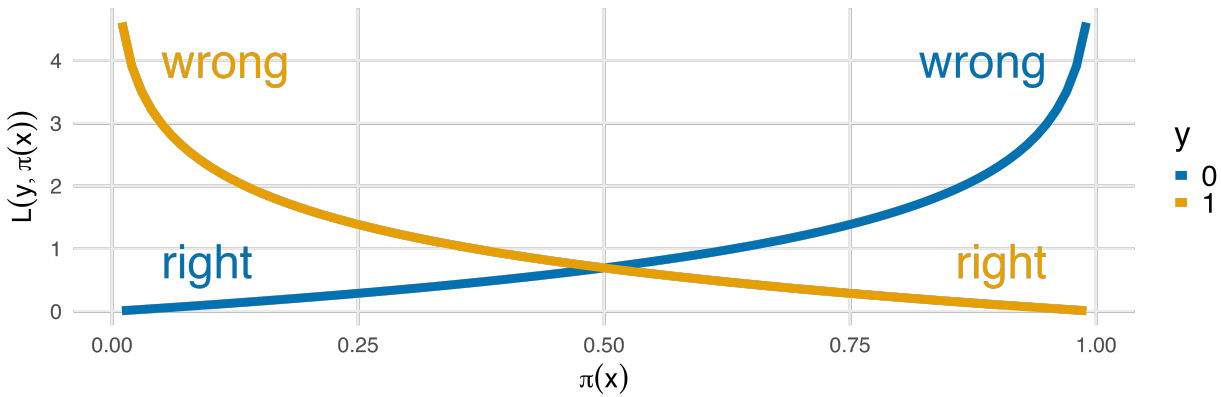
BERNOULLI / LOG LOSS

The resulting loss

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

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

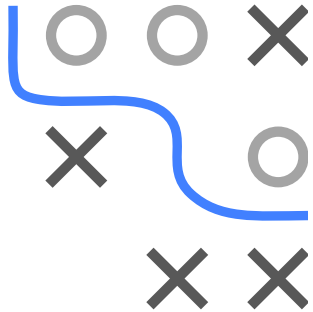


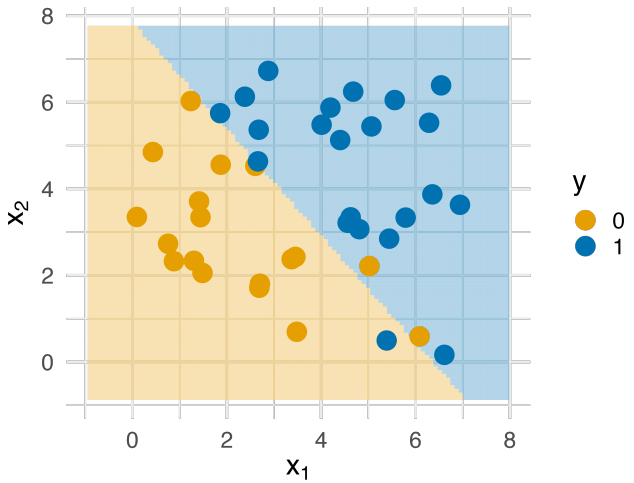


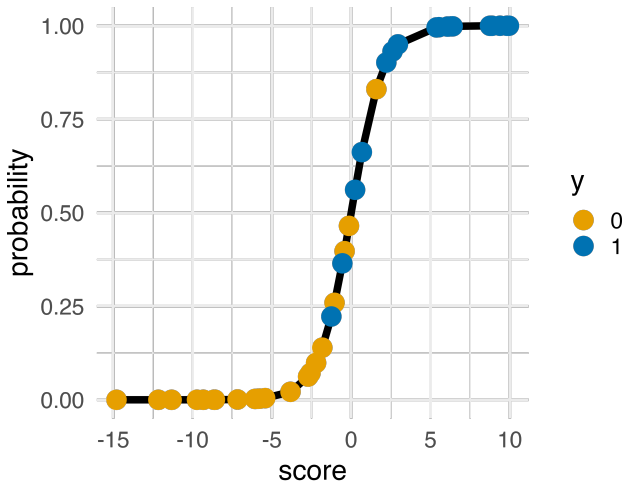
- Penalizes confidently wrong predictions heavily
- Is used for many other classifiers, e.g., in NNs or boosting

LOGISTIC REGRESSION IN 2D

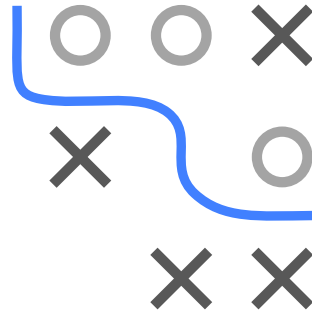
LR is a linear classifier, as $\pi(\mathbf{x} \mid \theta) = s\left(\theta^\top \mathbf{x}\right)$ and s is isotonic.







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

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

With $f(\mathbf{x} \mid \boldsymbol{\theta}) = \boldsymbol{\theta}^T \mathbf{x}$ 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

Sigmoid function

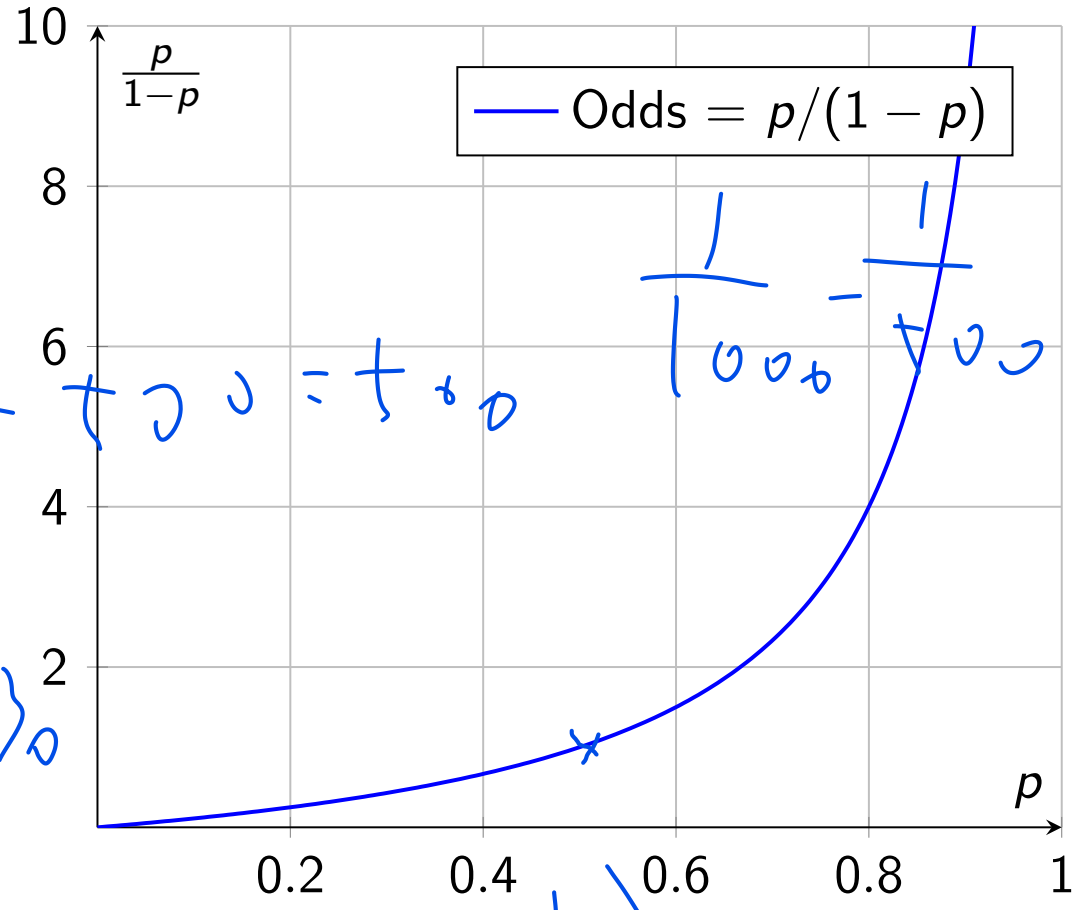
Odds: Concept and Definition

- Consider a binary outcome: success (probability p) vs. failure (probability $1 - p$).
- **Odds** of success are defined as:

$$\text{odds} = \frac{p}{1-p}.$$

- If $p = 0.5$, odds $= 1$ (success and failure equally likely).
- If $p > 0.5$, odds > 1 (success more likely).
- If $p < 0.5$, odds < 1 (failure more likely).

Plot: Odds vs. Probability



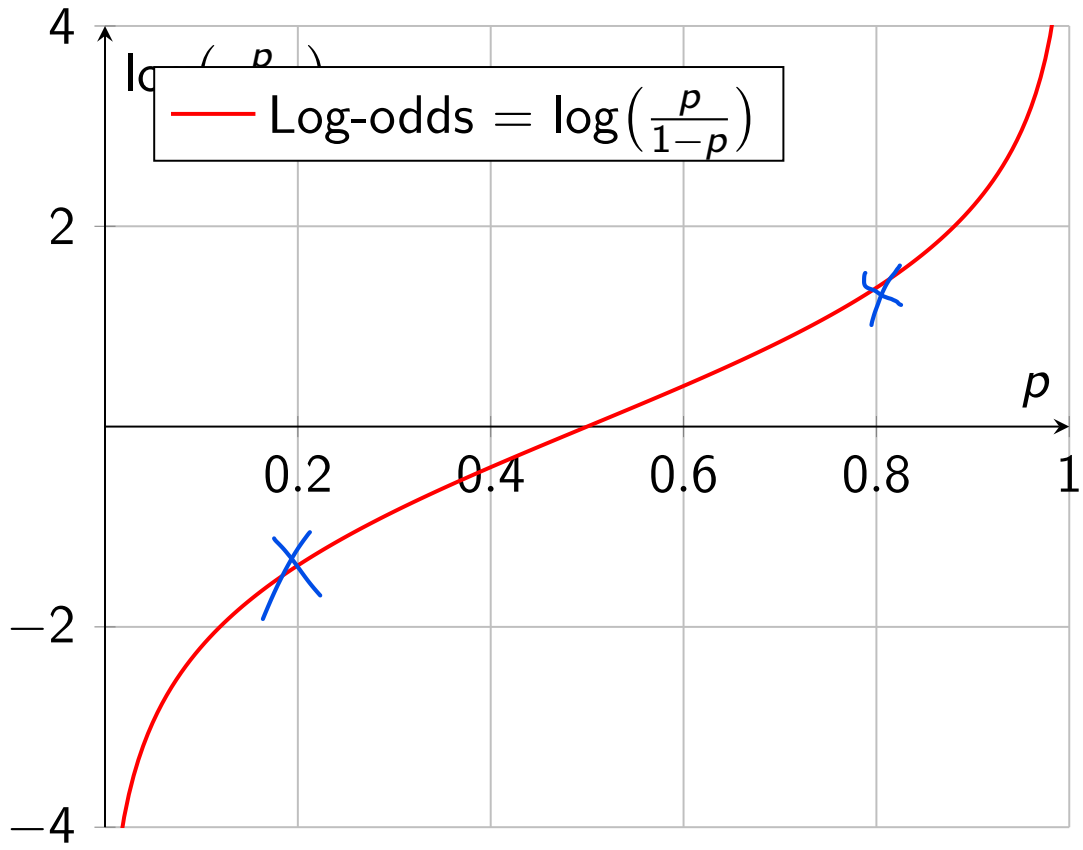
Log-Odds (Logit)

Log-odds or **logit** is:

$$\log\left(\frac{p}{1-p}\right).$$

- Maps odds $(0, \infty)$ to the entire real line $(-\infty, +\infty)$.
- Mathematical convenience: Addition in log-space corresponds to multiplication in probability space. - An event being 5 times bigger than another (odds ratio $= 5$) translates to $\log(5)$, and an event being 5 times smaller (odds ratio $= 1/5$) translates to $\log(1/5) = -\log(5)$. Hence, these two situations have the **same absolute effect** in log-scale, but opposite signs.
- Central to logistic regression: we assume a *linear* relationship in the log-odds.

Plot: Log-Odds vs. Probability



Linear Model for Log-Odds

- In **logistic regression**, we assume:

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n.$$

- Let $z = \beta_0 + \sum_{i=1}^{n} \beta_i x_i$.
- Then: $\log\left(\frac{p}{1-p}\right) = z$.

From Log-Odds to Probability

Starting with:

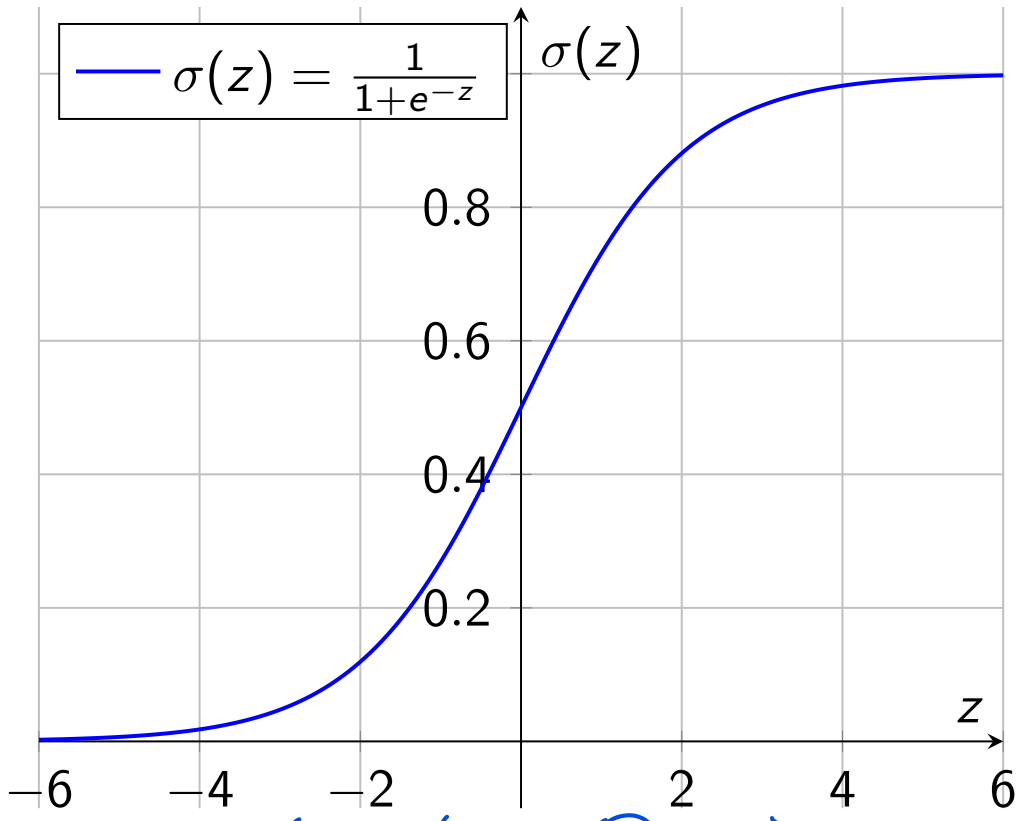
$$\log\left(\frac{p}{1-p}\right) = z \quad \Longrightarrow \quad \frac{p}{1-p} = e^z.$$

Then solving for p:

$$p = \frac{e^z}{1+e^z} = \frac{1}{1+e^{-z}}$$

- This function $\sigma(z) = \frac{1}{1+e^{-z}}$ is known as the **sigmoid** (or logistic) function.
- Maps $z \in (-\infty, +\infty)$ to $p \in (0, 1)$.

Plot: Sigmoid Function



$\theta_0 + \theta_1 x$

$x_1 = 4$ $\theta_0 = 3$ $\theta_1 = 8$

$\frac{1}{1+e^{-(3+4.8)}}$

Key Takeaways

1. **Odds**: $\frac{p}{1-p}$.
2. **Log-odds (logit)**: $\log\left(\frac{p}{1-p}\right)$ transforms $(0, \infty)$ to $(-\infty, +\infty)$.
3. **Linear modeling**: logistic regression assumes log-odds is a linear combination of features.
4. **Sigmoid function**: arises by solving for p from the log-odds. Maps real numbers to probabilities (0 to 1).