Summary

- Linearly separable classification problems.
- Logistic loss $\ell_{\text{log}}$ and (empirical) risk $\hat{R}_{\text{log}}$.
- Gradient descent.
For now, let’s consider binary classification: $\mathcal{Y} = \{-1, +1\}$.
A linear predictor $w \in \mathbb{R}^d$ classifies according to $\text{sign}(w^T x) \in \{-1, +1\}$.

Given $((x_i, y_i))_{i=1}^n$, a predictor $w \in \mathbb{R}^d$, we want $\text{sign}(w^T x_i)$ and $y_i$ to agree.
Let’s state our classification goal with a generic margin loss $\ell$:

$$\hat{R}(w) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i w^T x_i);$$

the key properties we want:

- $\ell$ is continuous;
- $\ell(z) \geq c \mathbb{1}[z \leq 0] = c\ell_{zo}(z)$ for some $c > 0$ and any $z \in \mathbb{R}$, which implies $\hat{R}_\ell(w) \geq c\hat{R}_{zo}(w)$.
- $\ell'(0) < 0$ (pushes stuff from wrong to right).
Let’s state our classification goal with a generic margin loss $\ell$:

$$\hat{R}(w) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i w^T x_i);$$

the key properties we want:

- $\ell$ is continuous;
- $\ell(z) \geq c \mathbb{1}[z \leq 0] = c \ell_{zo}(z)$ for some $c > 0$ and any $z \in \mathbb{R}$, which implies $\hat{R}_\ell(w) \geq c \hat{R}_{zo}(w)$.
- $\ell'(0) < 0$ (pushes stuff from wrong to right).

Examples.

- **Squared loss**, written in margin form: $\ell_{ls}(z) := (1 - z)^2$; note $\ell_{ls}(y\hat{y}) = (1 - y\hat{y})^2 = y^2(1 - y\hat{y})^2 = (y - \hat{y})^2$.
- **Logistic loss**: $\ell_{log}(z) = \ln(1 + \exp(-z))$. 
Logistic loss.

Squared loss.
(Slide from last time) Logistic loss 3

Logistic loss.

Squared loss.
Given a function $F : \mathbb{R}^d \rightarrow \mathbb{R}$, gradient descent is the iteration

$$w_{i+1} := w_i - \eta_i \nabla_w F(w_i),$$

where $w_0$ is given, and $\eta_i$ is a learning rate / step size.
Given a function $F : \mathbb{R}^d \to \mathbb{R}$, gradient descent is the iteration

$$w_{i+1} := w_i - \eta_i \nabla_w F(w_i),$$

where $w_0$ is given, and $\eta_i$ is a learning rate / step size.

Does this work for least squares?
Given a function \( F : \mathbb{R}^d \rightarrow \mathbb{R} \), gradient descent is the iteration

\[
\mathbf{w}_{i+1} := \mathbf{w}_i - \eta_i \nabla \mathbf{w} F(\mathbf{w}_i),
\]

where \( \mathbf{w}_0 \) is given, and \( \eta_i \) is a learning rate / step size.

Does this work for least squares? Later we’ll show it works for least squares and logistic regression due to convexity.
Gradient descent is the iteration: \( w_{i+1} := w_i - \eta_i \nabla_w \hat{R}_{\log}(w_i) \).

- Note \( \ell_{\log}'(z) = \frac{-1}{1 + \exp(z)} \), and use the chain rule (hw1!).
- Or use pytorch:

```python
def GD(X, y, loss, step = 0.1, n_iters = 10000):
    w = torch.zeros(X.shape[1], requires_grad = True)
    for i in range(n_iters):
        l = loss(X, y, w).mean()
        l.backward()
        with torch.no_grad():
            w -= step * w.grad
            w.grad.zero_()

    return w
```


Part 2 of logistic regression...
5. A maximum likelihood derivation
We’ve studied an ERM perspective on logistic regression:

- Form empirical logistic risk \( \hat{R}_{\log}(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^{n} \ln(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i)) \).
- Approximately solve \( \arg \min_{\mathbf{w} \in \mathbb{R}^d} \hat{R}_{\log}(\mathbf{w}) \) via gradient descent (or other convex optimization technique).

We only justified it with “popularity”!

Today we’ll derive \( \hat{R}_{\log} \) via Maximum Likelihood Estimation (MLE).

1. We form a model for \( \Pr[Y = 1 | X = x] \), parameterized by \( \mathbf{w} \).
2. We form a full data log-likelihood (equivalent to \( \hat{R}_{\log} \)).

Let’s first describe the distributions underlying the data.
Learning prediction functions

**IID model** for *supervised learning*:

$(X_1, Y_1), \ldots, (X_n, Y_n), (X, Y)$ are iid random *pairs* (i.e., *labeled examples*).

- $X$ takes values in $\mathcal{X}$. E.g., $\mathcal{X} = \mathbb{R}^d$.
- $Y$ takes values in $\mathcal{Y}$. E.g.,
  - *(regression problems)* $\mathcal{Y} = \mathbb{R}$;
  - *(classification problems)* $\mathcal{Y} = \{1, \ldots, K\}$ or $\mathcal{Y} = \{0, 1\}$ or $\mathcal{Y} = \{-1, +1\}$.

1. We observe $(X_1, Y_1), \ldots, (X_n, Y_n)$, and the choose a *prediction function* (i.e., *predictor*)

   $$\hat{f} : \mathcal{X} \to \mathcal{Y},$$

   This is called *“learning”* or *“training”*.

2. At prediction time, observe $X$, and form prediction $\hat{f}(X)$.

3. Outcome is $Y$, and

   - *squared loss* is $(\hat{f}(X) - Y)^2$ (regression problems).
   - *zero-one loss* is $\mathbb{1}\{\hat{f}(X) \neq Y\}$ (classification problems).

**Note**: expected zero-one loss is

$$\mathbb{E}[\mathbb{1}\{\hat{f}(X) \neq Y\}] = \mathbb{P}(\hat{f}(X) \neq Y),$$

which we also call *error rate*. 
Distributions over labeled examples

\(\mathcal{X}\): space of possible side-information (\textit{feature space}).
\(\mathcal{Y}\): space of possible outcomes (\textit{label space} or \textit{output space}).

Distribution \(P\) of random pair \((X, Y)\) taking values in \(\mathcal{X} \times \mathcal{Y}\) can be thought of in two parts:

1. \textit{Marginal distribution} \(P_X\) of \(X\):
   \[P_X\] is a probability distribution on \(\mathcal{X}\).

2. \textit{Conditional distribution} \(P_{Y|X=x}\) of \(Y\) given \(X = x\), for each \(x \in \mathcal{X}\):
   \[P_{Y|X=x}\] is a probability distribution on \(\mathcal{Y}\).
Optimal classifier

For binary classification, what function $f : \mathcal{X} \to \{0, 1\}$ has smallest risk (i.e., error rate) $\mathcal{R}(f) := \mathbb{P}(f(X) \neq Y)$?

- Conditional on $X = x$, the minimizer of conditional risk
  $$\hat{y} \mapsto \mathbb{P}(\hat{y} \neq Y \mid X = x)$$
  is
  $$\hat{y} := \begin{cases} 
1 & \text{if } \mathbb{P}(Y = 1 \mid X = x) > 1/2, \\
0 & \text{if } \mathbb{P}(Y = 1 \mid X = x) \leq 1/2.
\end{cases}$$

- Therefore, the function $f^* : \mathcal{X} \to \{0, 1\}$ where
  $$f^*(x) = \begin{cases} 
1 & \text{if } \mathbb{P}(Y = 1 \mid X = x) > 1/2, \\
0 & \text{if } \mathbb{P}(Y = 1 \mid X = x) \leq 1/2,
\end{cases} \quad x \in \mathcal{X},$$
  has the smallest risk.

- $f^*$ is called the **Bayes (optimal) classifier**.

For $\mathcal{Y} = \{1, \ldots, K\}$,
$$f^*(x) = \arg \max_{y \in \mathcal{Y}} \mathbb{P}(Y = y \mid X = x), \quad x \in \mathcal{X}.$$
Logistic regression

Suppose $\mathcal{X} = \mathbb{R}^d$ and $\mathcal{Y} = \{0, 1\}$. A \textit{logistic regression model} is a statistical model where the conditional probability function has a particular form:

$$Y \mid X = x \sim \text{Bern}(\eta_w(x)), \quad x \in \mathbb{R}^d,$$

with

$$\eta_w(x) := \text{logistic}(x^T w), \quad x \in \mathbb{R}^d$$

(with parameters $w \in \mathbb{R}^d$), and

$$\text{logistic}(z) := \frac{1}{1 + e^{-z}} = \frac{e^z}{1 + e^z}, \quad z \in \mathbb{R}.$$

- Conditional distribution of $Y$ given $X$ is Bernoulli; marginal distribution of $X$ not specified.
- With least squares, $Y \mid X = x$ was $N(w^T x, \sigma^2)$. 

![Graph showing logistic function](image_url)
Log-likelihood of $w$ in iid logistic regression model, given data
$(X_i, Y_i) = (x_i, y_i)$ for $i = 1, \ldots, n$:

$$
\ln \prod_{i=1}^{n} \eta_w(x_i)^{y_i} (1 - \eta_w(x_i))^{1-y_i} \\
= \sum_{i=1}^{n} \left( y_i \ln \eta_w(x_i) + (1 - y_i) \ln(1 - \eta_w(x_i)) \right) \\
= - \sum_{i=1}^{n} \left( y_i \ln(1 + \exp(-w^T x_i)) + (1 - y_i) \ln(1 + \exp(w^T x_i)) \right) \\
= - \sum_{i=1}^{n} \ln(1 + \exp(-(2y_i - 1)w^T x_i)),
$$

and old form is recovered with labels $\tilde{y}_i := 2y_i - 1 \in \{-1, +1\}$. 
Equivalent way to characterize logistic regression model:
The **log-odds function**, given by

\[
\text{log-odds}_\beta(x) = \ln \frac{\eta_\beta(x)}{1 - \eta_\beta(x)} = \ln \left( \frac{e^{x^T \beta}}{1 + e^{x^T \beta}} \right) = x^T \beta,
\]

is a linear function\(^1\), parameterized by \(\beta \in \mathbb{R}^d\).

\(^1\)Some authors allow affine function; we can get this using affine expansion.
Equivalent way to characterize logistic regression model:

The log-odds function, given by

$$\text{log-odds}_\beta(x) = \ln \frac{\eta_\beta(x)}{1 - \eta_\beta(x)} = \ln \left( \frac{e^{x^T \beta}}{1 + e^{x^T \beta}} \right) = x^T \beta,$$

is a linear function\(^1\), parameterized by $\beta \in \mathbb{R}^d$.

Bayes optimal classifier $f_\beta : \mathbb{R}^d \rightarrow \{0, 1\}$ in logistic regression model:

$$f_\beta(x) = \begin{cases} 0 & \text{if } x^T \beta \leq 0, \\ 1 & \text{if } x^T \beta > 0. \end{cases}$$

\(^1\)Some authors allow affine function; we can get this using affine expansion.
Log-odds function and classifier

Equivalent way to characterize logistic regression model:
The log-odds function, given by

$$\text{log-odds}_\beta(x) = \ln \frac{\eta_\beta(x)}{1 - \eta_\beta(x)} = \ln \left( \frac{e^{x^T\beta}}{1 + e^{x^T\beta}} \right) = x^T\beta,$$

is a linear function\(^1\), parameterized by \(\beta \in \mathbb{R}^d\).

Bayes optimal classifier \(f_\beta : \mathbb{R}^d \rightarrow \{0, 1\}\) in logistic regression model:

$$f_\beta(x) = \begin{cases} 
0 & \text{if } x^T\beta \leq 0, \\
1 & \text{if } x^T\beta > 0.
\end{cases}$$

Such classifiers are called linear classifiers.

\(^1\)Some authors allow affine function; we can get this using affine expansion.
Where does the logistic regression model come from?

The following is one way the logistic regression model comes about (but not the only way).
Where does the logistic regression model come from?

The following is one way the logistic regression model comes about (but not the only way).

Consider the following generative model for \((X, Y)\) where

\[
Y \sim \text{Bern}(\pi),
\]

\[
X \mid Y = y \sim \text{N}(\mu_y, \Sigma).
\]
The following is one way the logistic regression model comes about (but not the only way).

Consider the following generative model for \((X, Y)\) where

\[
Y \sim \text{Bern}(\pi), \\
X \mid Y = y \sim \mathcal{N}(\mu_y, \Sigma).
\]

- Parameters: \(\pi \in [0, 1], \ \mu_0, \mu_1 \in \mathbb{R}^d, \ \Sigma \in \mathbb{R}^{d \times d} \ \text{sym. \& pos. def.}\)
Where does the logistic regression model come from?

The following is one way the logistic regression model comes about (but not the only way).

Consider the following generative model for \((X, Y)\) where

\[
Y \sim \text{Bern}(\pi), \\
X \mid Y = y \sim \text{N}(\mu_y, \Sigma).
\]

- **Parameters:** \(\pi \in [0, 1], \mu_0, \mu_1 \in \mathbb{R}^d, \Sigma \in \mathbb{R}^{d \times d} \text{ sym. & pos. def.}\)

Figure shows (unconditional) probability density function for \(X\).
Suppose we are given the following.

- \( p_Y \): probability mass function for \( Y \).
- \( p_{X|Y=y} \): conditional probability density function for \( X \) given \( Y = y \).
Statistical model for conditional distribution

Suppose we are given the following.

- \( p_Y \): probability mass function for \( Y \).
- \( p_{X|Y=y} \): conditional probability density function for \( X \) given \( Y = y \).

What is the conditional distribution of \( Y \) given \( X \)?
Suppose we are given the following.

- \( p_Y \): probability mass function for \( Y \).
- \( p_{X|Y=y} \): conditional probability density function for \( X \) given \( Y = y \).

What is the conditional distribution of \( Y \) given \( X \)?

By Bayes' rule: for any \( x \in \mathbb{R}^d \),

\[
P(Y = y \mid X = x) = \frac{p_Y(y) \cdot p_{X|Y=y}(x)}{p_X(x)}
\]

(where \( p_X \) is unconditional density for \( X \)).
Suppose we are given the following.

- \( p_Y \): probability mass function for \( Y \).
- \( p_{X|Y=y} \): conditional probability density function for \( X \) given \( Y = y \).

What is the conditional distribution of \( Y \) given \( X \)?

By Bayes’ rule: for any \( x \in \mathbb{R}^d \),

\[
P(Y = y \mid X = x) = \frac{p_Y(y) \cdot p_{X|Y=y}(x)}{p_X(x)}
\]

(where \( p_X \) is unconditional density for \( X \)).

Therefore, log-odds function is

\[
\text{log-odds}(x) = \ln \left( \frac{p_Y(1)}{p_Y(0)} \cdot \frac{p_{X|Y=1}(x)}{p_{X|Y=0}(x)} \right).
\]
Log-odds function for our toy model

Log-odds function:

\[ \text{log-odds}(\mathbf{x}) = \ln \left( \frac{p_Y(1)}{p_Y(0)} \right) + \ln \left( \frac{p_{X|Y=1}(\mathbf{x})}{p_{X|Y=0}(\mathbf{x})} \right). \]
Log-odds function for our toy model

Log-odds function:

$$ \text{log-odds}(\mathbf{x}) = \ln \left( \frac{p_{Y(1)}}{p_{Y(0)}} \right) + \ln \left( \frac{p_{X|Y=1}(\mathbf{x})}{p_{X|Y=0}(\mathbf{x})} \right). $$

In our toy model, we have $Y \sim \text{Bern}(\pi)$ and $X \mid Y = y \sim \mathcal{N}(\mu_y, \mathbf{A}\mathbf{A}^T)$, so:

$$ \text{log-odds}(\mathbf{x}) = \ln \frac{\pi}{1 - \pi} + \ln \frac{e^{-\frac{1}{2} \| \mathbf{A}^{-1}(\mathbf{x} - \mu_1) \|^2_2}}{e^{-\frac{1}{2} \| \mathbf{A}^{-1}(\mathbf{x} - \mu_0) \|^2_2}} $$

$$ = \ln \frac{\pi}{1 - \pi} - \frac{1}{2} \| \mathbf{A}^{-1}(\mathbf{x} - \mu_1) \|^2_2 + \frac{1}{2} \| \mathbf{A}^{-1}(\mathbf{x} - \mu_0) \|^2_2 $$

$$ = \ln \frac{\pi}{1 - \pi} - \frac{1}{2} (\| \mathbf{A}^{-1}_1 \mu_1 \|_2^2 - \| \mathbf{A}^{-1}_0 \mu_0 \|_2^2) + (\mu_1 - \mu_0)^T (\mathbf{A}\mathbf{A}^T)^{-1} \mathbf{x}. $$

constant—doesn’t depend on $\mathbf{x}$

linear function of $\mathbf{x}$
Log-odds function for our toy model

Log-odds function:

$\text{log-odds}(x) = \ln \left( \frac{p_Y(1)}{p_Y(0)} \right) + \ln \left( \frac{p_{X|Y=1}(x)}{p_{X|Y=0}(x)} \right).$

In our toy model, we have $Y \sim \text{Bern}(\pi)$ and $X \mid Y = y \sim \text{N}(\mu_y, AA^T)$, so:

$\text{log-odds}(x) = \ln \frac{\pi}{1 - \pi} + \ln \frac{e^{-\frac{1}{2} \|A^{-1}(x - \mu_1)\|_2^2}}{e^{-\frac{1}{2} \|A^{-1}(x - \mu_0)\|_2^2}}$

$= \ln \frac{\pi}{1 - \pi} - \frac{1}{2} \|A^{-1}(x - \mu_1)\|_2^2 + \frac{1}{2} \|A^{-1}(x - \mu_0)\|_2^2$

$= \ln \frac{\pi}{1 - \pi} - \frac{1}{2} \left( \|A^{-1} \mu_1\|_2^2 - \|A^{-1} \mu_0\|_2^2 \right) + (\mu_1 - \mu_0)^T (AA^T)^{-1} x.$

- ▶ This is an affine function of $x$. 

constant—doesn’t depend on $x$

linear function of $x$
Log-odds function for our toy model

Log-odds function:

\[
\log\text{-odds}(\mathbf{x}) = \ln \left( \frac{p_Y(1)}{p_Y(0)} \right) + \ln \left( \frac{p_{X|Y=1}(\mathbf{x})}{p_{X|Y=0}(\mathbf{x})} \right).
\]

In our toy model, we have \( Y \sim \text{Bern}(\pi) \) and \( \mathbf{X} \mid Y = y \sim \mathcal{N}(\mu_y, \mathbf{A} \mathbf{A}^\top) \), so:

\[
\begin{align*}
\log\text{-odds}(\mathbf{x}) &= \ln \frac{\pi}{1 - \pi} + \ln \frac{e^{-\frac{1}{2} \| \mathbf{A}^{-1}(\mathbf{x} - \mu_1) \|^2_2}}{e^{-\frac{1}{2} \| \mathbf{A}^{-1}(\mathbf{x} - \mu_0) \|^2_2}} \\
&= \ln \frac{\pi}{1 - \pi} - \frac{1}{2} \| \mathbf{A}^{-1}(\mathbf{x} - \mu_1) \|^2_2 + \frac{1}{2} \| \mathbf{A}^{-1}(\mathbf{x} - \mu_0) \|^2_2 \\
&= \ln \frac{\pi}{1 - \pi} - \frac{1}{2} (\| \mathbf{A}^{-1}\mu_1 \|^2_2 - \| \mathbf{A}^{-1}\mu_0 \|^2_2) + (\mu_1 - \mu_0)^\top (\mathbf{A} \mathbf{A}^\top)^{-1} \mathbf{x}.
\end{align*}
\]

- This is an affine function of \( \mathbf{x} \).
- Hence, the statistical model for \( Y \mid \mathbf{X} \) is a logistic regression model (with affine feature expansion).
Log-odds function for our toy model

Log-odds function:

$$\text{log-odds}(x) = \ln \left( \frac{p_Y(1)}{p_Y(0)} \right) + \ln \left( \frac{p_{X|Y=1}(x)}{p_{X|Y=0}(x)} \right).$$

In our toy model, we have $Y \sim \text{Bern}(\pi)$ and $X | Y = y \sim \mathcal{N}(\mu_y, AA^T)$, so:

$$\text{log-odds}(x) = \ln \frac{\pi}{1 - \pi} \left[ 1 + \frac{e^{-\frac{1}{2} \| A^{-1}(x - \mu_1) \|^2_2}}{e^{-\frac{1}{2} \| A^{-1}(x - \mu_0) \|^2_2}} \right]$$

$$= \ln \frac{\pi}{1 - \pi} - \frac{1}{2} \| A^{-1}(x - \mu_1) \|^2_2 + \frac{1}{2} \| A^{-1}(x - \mu_0) \|^2_2$$

$$= \ln \frac{\pi}{1 - \pi} - \frac{1}{2} \left( \| A^{-1} \mu_1 \|^2_2 - \| A^{-1} \mu_0 \|^2_2 \right) + \frac{(\mu_1 - \mu_0)^T (AA^T)^{-1} x}{\| A^{-1} \mu_1 \|^2_2 - \| A^{-1} \mu_0 \|^2_2}$$

constant—doesn’t depend on $x$

linear function of $x$

- This is an affine function of $x$.
- Hence, the statistical model for $Y | X$ is a logistic regression model (with affine feature expansion).
- Important: Logistic regression model forgets about $p_{X|Y=y}$!
6. Multiclass classification and cross-entropy
All our methods so far handle multiclass:

- **$k$-nn and decision tree**: plurality label.

- **Least squares**: 
  \[ \min_{W \in \mathbb{R}^{d \times k}} \| AW - B \|_F^2 \]  
  with \( B \in \mathbb{R}^{n \times k} \); 
  \( W \in \mathbb{R}^{d \times k} \) is $k$ separate linear regressors in $\mathbb{R}^d$.

How about **linear classifiers**?

- At prediction time, \( x \mapsto \arg \max_y \hat{f}(x)_y \).

- As in binary case: interpretation \( f(x)_y = \Pr[Y = y | X = x] \).

What is a good loss function?
Cross-entropy

Given two probability vectors $p, q \in \Delta_k = \{ p \in \mathbb{R}^k_{\geq 0} : \sum_i p_i = 1 \}$,

\[
H(p, q) = -\sum_{i=1}^k p_i \ln q_i \quad \text{(cross-entropy)}.
\]

- If $p = q$, then $H(p, q) = H(p)$ (entropy); indeed

\[
H(p, q) = -\sum_{i=1}^k p_i \ln \left( p_i \frac{q_i}{p_i} \right) = H(p) + \text{KL}(p, q).
\]

Since $\text{KL} \geq 0$ and moreover 0 iff $p = q$, this is the cost/entropy of $p$ plus a penalty for differing.

- Choose encoding $\tilde{y}_i = e_y$ for $y \in \{1, \ldots, k\}$, and $\hat{y} \propto \exp(f(x))$ with $f : \mathbb{R}^d \to \mathbb{R}^k$;

\[
\ell_{ce}(\tilde{y}, f(x)) = H(\tilde{y}, \hat{y}) = -\sum_{i=1}^k \tilde{y}_i \ln \left( \frac{\exp(f(x)_i)}{\sum_{j=1}^k \exp(f(x)_j)} \right)
\]

\[
= -\ln \left( \frac{\exp(f(x)_y)}{\sum_{j=1}^k \exp(f(x)_j)} \right) = -f(x)_y + \ln \sum_{j=1}^k \exp(f(x)_j).
\]

(In pytorch, use `torch.nn.CrossEntropyLoss()(f(x), y).`)
Cross-entropy, classification, and margins

The zero-one loss for classification is

$$\ell_{zo}(y_i, f(\mathbf{x})) = 1 \left[ y_i \neq \arg \max_j f(\mathbf{x})_j \right].$$

In the multiclass case, can define margin as

$$f(\mathbf{x})_y - \max_{j \neq y} f(\mathbf{x})_j,$$

interpreted as “the distance by which $f$ is correct”. (Can be negative!)

Since $\ln \sum_j z_j \approx \max_j z_j$, cross-entropy satisfies

$$\ell_{ce}(\tilde{y}_i, f(\mathbf{x})) = -f(\mathbf{x})_y + \ln \sum_j \exp (f(\mathbf{x})_j)$$

$$\approx -f(\mathbf{x})_y + \max_j f(\mathbf{x})_j,$$

thus minimizing cross-entropy maximizes margins.
Cross-entropy and logistic loss

With a linear model \( f(x) = W^T x \) for \( W \in \mathbb{R}^{d \times k} \), with two labels \( \{1, 2\} \),

\[
\ell_{ce}(e_1, f(x)) = -\ln \left( \frac{\exp(f(x)_1)}{\exp(f(x)_1) + \exp(f(x)_2)} \right) = \ln \left( 1 + \exp(f(x)_2 - f(x)_1) \right)
\]

\[
\ell_{ce}(e_2, f(x)) = \ln \left( 1 + \exp(f(x)_1 - f(x)_2) \right).
\]

Thus if we write \( \tilde{y} := 2y - 3 \) and \( v := W_{:2} - W_{:1} \),

\[
\ln(1 + \exp(-\tilde{y}v^T x)) = \ell_{ce}(e_y, W^T x).
\]
7. Summary
Summary

Part 1.
- Linearly separable classification problems.
- Logistic loss $\ell_{\log}$ and (empirical) risk $\hat{R}_{\log}$.
- Gradient descent.

Part 2.
- MLE perspective on logistic regression.
- Cross-entropy.