Logistic Regression & convex optimization
Announcements:

This week we will release P3 and HW3
Recap on Naive Bayes

NB is a **generative model** which models $P(x, y)$

$$P(y \mid x) \propto P(y)P(x \mid y) = P(y) \prod_{i=1}^{d} P(x[i] \mid y)$$

Conditional independent assumption given label

$$\arg \max_{y} P(y \mid x)$$
Perceptron VS Gaussian Naive Bayes

\[
\frac{1}{8}
\]

\[P(x|y=-1)\]

\[P(x|y=+1)\]
Today

Logistic regression — a *discriminative learning* approach that directly models $P(y \mid x)$ for classification
Outline for today

1. Logistic Regression

2. Convex optimization

3. Gradient Descent
Logistic Regression

Setting: binary classification \( \mathcal{D} = \{x_i, y_i\}_{i=1}^n \), \( (x_i, y_i) \sim P \),

\( x_i \in \mathbb{R}^d, y_i \in \{-1, +1\} \)
Logistic Regression

Setting: binary classification $\mathcal{D} = \{x_i, y_i\}_{i=1}^n$, $(x_i, y_i) \sim P$, $x_i \in \mathbb{R}^d$, $y_i \in \{-1, +1\}$

(Note, we always assume $x$ contains a constant $1$)
Logistic Regression

Setting: binary classification $\mathcal{D} = \{x_i, y_i\}_{i=1}^{n}$, $(x_i, y_i) \sim P$,
$x_i \in \mathbb{R}^d, y_i \in \{-1, +1\}$

(Note, we always assume $x$ contains a constant 1)

Logistic regression directly models $P(y \mid x)$
Logistic Regression

Setting: binary classification \( \mathcal{D} = \{x_i, y_i\}_{i=1}^n \), \((x_i, y_i) \sim P\),
\( x_i \in \mathbb{R}^d, y_i \in \{-1, +1\} \)

(Note, we always assume \( x \) contains a constant 1)

Logistic regression **directly models** \( P(y \mid x) \)

\[
P(y \mid x) = \frac{1}{1 + \exp\left(-y(x^T w^*)\right)}
\]
Logistic regression assumes:

\[ P(y| x) = \frac{1}{1 + \exp(-y(x^T w^*))} \]

Draw the Sigmoid function \( \frac{1}{1 + \exp(-Z)} \)
Logistic regression assumes:

\[
P(y|x) = \frac{1}{1 + \exp\left(-y(x^Tw^*)\right)}
\]

The model assigns higher prob to

\[y = \text{sign}(x^Tw^*)\]
Logistic regression assumes:

\[ P(y | x) = \frac{1}{1 + \exp(-y(x^T w^*))} \]
Logistic regression assumes:

$$P(y \mid x) = \frac{1}{1 + \exp(-y(x^T w^*))}$$
Learn via MLE

Recall we have data $D = \{x_i, y_i\}_{i=1}^n$

$$Y = \{y_1, \ldots, y_n\}$$

$$X = \{x_1, \ldots, x_n\}$$

$$\arg \max_w P(D \mid w)$$

$$\Rightarrow P(D \mid w) = P(Y \mid X; w) \prod_{o=0}^n P(X; o)$$

$$= P(X)$$

$$= P(Y \mid X; w) P(X)$$
Learn via MLE

Recall we have data $\mathcal{D} = \{x_i, y_i\}_{i=1}^n$

$$\arg\max_w P(\mathcal{D} | w) = \arg\max_w P\left(\{y_i\}_{i=1}^n | \{x_i\}_{i=1}^n ; w\right)$$
Learn via MLE

Recall we have data $\mathcal{D} = \{x_i, y_i\}_{i=1}^n$

$$\arg \max_w P(\mathcal{D} \mid w) = \arg \max_w P(\{y_i\}_{i=1}^n \mid \{x_i\}_{i=1}^n; w)$$

$$= \arg \max_w \prod_{i=1}^n P(y_i \mid x_i; w)$$
Learn via MLE

Recall we have data $\mathcal{D} = \{x_i, y_i\}_{i=1}^{n}$

$$\arg \max_w P(\mathcal{D} \mid w) = \arg \max_w P \left( \{y_i\}_{i=1}^{n} \mid \{x_i\}_{i=1}^{n}; w \right)$$

$$= \arg \max_w \prod_{i=1}^{n} P \left( y_i \mid x_i; w \right)$$

Plug in logistic assumption and add log:

$$\arg \max_w \sum_{i=1}^{n} - \ln \left[ 1 + \exp \left( -y_i (w^\top x_i) \right) \right]$$
Learn via MLE

\[ \hat{w}_{mle} := \arg \max_w \sum_{i=1}^{n} \ln \left( \frac{1}{1 + \exp(-y_i(w^T x_i))} \right) \]

Intuitively, \( \hat{w}_{mle} \) tries to explain the label:
Learn via MLE

\[ \hat{w}_{mle} := \arg \max_w \sum_{i=1}^n \ln \left[ \frac{1}{1 + \exp \left(-y_i(w^T x_i)\right)} \right] \]

Intuitively, \( \hat{w}_{mle} \) tries to explain the label:

Q: for \( y_i = +1 \), what we should expect from \( \hat{w}_{mle}^T x_i \)?

\( \hat{w}_{mle}^T x_i \geq 0 \)

\( (y_i)(\hat{w}_{mle}^T x_i) \gg 0 \)
Learn via MLE

\[ \hat{w}_{mle} := \arg \max_w \sum_{i=1}^{n} \ln \left( \frac{1}{1 + \exp(-y_i(w^T x_i))} \right) \]

Intuitively, \( \hat{w}_{mle} \) tries to explain the label:

Q: for \( y_i = +1 \), what we should expect from \( \hat{w}_{mle}^T x_i \)?

Q: for \( y_i = -1 \), what we should expect from \( \hat{w}_{mle}^T x_i \)?

\( < 0 \)
Learn via MLE

\[ \hat{w}_{mle} := \arg \max_w \sum_{i=1}^{n} \ln \left[ \frac{1}{1 + \exp (-y_i(w^T x_i))} \right] \]

Intuitively, \( \hat{w}_{mle} \) tries to explain the label:

Q: for \( y_i = +1 \), what we should expect from \( \hat{w}_{mle}^T x_i \)?

Q: for \( y_i = -1 \), what we should expect from \( \hat{w}_{mle}^T x_i \)?
Learn via MAP

\[ P(w \mid \mathcal{D}) \propto P(w)P(\mathcal{D} \mid w) \]
Learn via MAP

\[ P(w \mid \mathcal{D}) \propto P(w)P(\mathcal{D} \mid w) \]

We use Gaussian prior, i.e., \( P(w) = \mathcal{N}(0, \sigma^2 I) \).
Learn via MAP

\[ P(w \mid \mathcal{D}) \propto P(w)P(\mathcal{D} \mid w) \]

We use Gaussian prior, i.e., \( P(w) = \mathcal{N}(0, \sigma^2 I) \)

\[
\arg\max_w \ln \left( P(w) \prod_{i=1}^{n} P(y_i \mid x_i, w) \right) = \arg\max_w \ln P(w) + \sum_{i=1}^{n} \ln P(y_i \mid x_i, w)
\]
Learn via MAP

\[ P(w \mid \mathcal{D}) \propto P(w)P(\mathcal{D} \mid w) \]

We use Gaussian prior, i.e., \( P(w) = \mathcal{N}(0, \sigma^2 I) \)

\[
\arg \max_w \ln \left( P(w) \prod_{i=1}^{n} P(y_i \mid x_i, w) \right) = \arg \max_w \ln P(w) + \sum_{i=1}^{n} \ln P(y_i \mid x_i, w)
\]

\[
= \arg \min_w \left( \sum_{i=1}^{n} \ln (1 + \exp(-y_i(w^T x_i))) + \frac{\|w\|_2^2}{2\sigma^2} \right)
\]

\[\text{prior / regularization}\]
Comparison to Navie Bayes

1. Logistic regression does not model $P(x \mid y)$
Comparison to Naive Bayes

1. Logistic regression does not model $P(x \mid y)$

2. Gaussian NB leads a linear classifier in the form of

$$P(y \mid x) = \frac{1}{1 + \exp(w^T x)}$$
Comparison to Naïve Bayes

1. Logistic regression does not model \( P(x \mid y) \)

2. Gaussian NB leads a linear classifier in the form of
\[
P(y \mid x) = \frac{1}{1 + \exp(w^\top x)}
\]

Gaussian NB is a special case of logistic regression
Outline for today

1. Logistic Regression

2. Convex optimization

3. Gradient Descent
We need to solve the optimization problem

\[ \hat{w} := \arg \min_w \sum_{i=1}^n \ln \left[ 1 + \exp \left( -y_i (w^T x_i) \right) + \lambda \|w\|^2_2 \right] \]

\[ : = \ell(w) \]

\[ \forall \ell(w) = 0 \]

Solve for \( w \)
We need to solve the optimization problem

\[
\hat{w} := \arg \min_w \sum_{i=1}^n \ln \left[ 1 + \exp \left( -y_i(w^T x_i) \right) + \lambda \|w\|_2^2 \right]
\]

There is no closed-form solution for the minimizer; luckily, \( \ell(w) \) is convex
We needs to solve the optimization problem

\[ \hat{w} := \arg \min_w \sum_{i=1}^n \ln \left[ 1 + \exp \left( -y_i (w^T x_i) \right) + \lambda \|w\|^2_2 \right] \]

\[ := \ell(w) \]

There is no closed-form solution for the minimizer; luckily, \( \ell(w) \) is convex

We will find an approximate minimizer via gradient descent
Setup for Optimization

We consider minimizing a (convex) function $\arg \min_w \ell(w)$.
Setup for Optimization

We consider minimizing a (convex) function $\arg \min_w \ell(w)$.

Def of convexity:

$\forall (x, x'), \alpha \in [0, 1], \ell(\alpha x + (1 - \alpha)x') \leq \alpha \ell(x) + (1 - \alpha)\ell(x')$
Setup for Optimization

We consider minimizing a (convex) function $\arg \min_w \ell(w)$

Def of convexity:

$\forall (x, x'), \alpha \in [0,1], \ell(\alpha x + (1-\alpha)x') \leq \alpha \ell(x) + (1-\alpha)\ell(x')$
Setup for Optimization

We consider minimizing a (convex) function $\arg\min_w \ell(w)$

Def of convexity:

$$\forall (x, x'), \alpha \in [0,1], \ell(\alpha x + (1 - \alpha)x') \leq \alpha \ell(x) + (1 - \alpha)\ell(x')$$
Global minimizer of a convex function

A convex function has global minimizer which has gradient equal to 0
Global minimizer of a convex function

A convex function has global minimizer which has gradient equal to 0
Examples of non-convex functions

Saddle point \((\ell(x, y) = x^2 - y^2)\)
Outline for today

1. Logistic Regression

2. Convex optimization

3. Gradient Descent
The Gradient Descent algorithm

Goal: minimize $\ell(w)$

Initialize $w^0 \in \mathbb{R}^d$

Iterate until convergence:
The Gradient Descent algorithm

Goal: minimize $\ell(w)$

Initialize $w^0 \in \mathbb{R}^d$

Iterate until convergence:

1. Compute gradient $g^t = \nabla \ell(w) \big|_{w=w_t}$
The Gradient Descent algorithm

Goal: minimize $\ell(w)$

Initialize $w^0 \in \mathbb{R}^d$

Iterate until convergence:

1. Compute gradient $g^t = \nabla \ell(w) |_{w=w_t}$

2. Update (GD): $w^{t+1} = w^t - \eta g^t$
The Gradient Descent algorithm

Goal: minimize $\ell(w)$

Initialize $w^0 \in \mathbb{R}^d$

Iterate until convergence:

1. Compute gradient $g^t = \nabla \ell(w) \big|_{w=w_t}$
2. Update (GD): $w^{t+1} = w^t - \eta g^t$

$\eta$: learning rate
The Gradient Descent demo

\[ \min_{x,y} (x^2 + y^2) \]
The Gradient Descent demo

\[ \min_{x,y}(x^2 + y^2) \]
Informal proof for GD convergence

First-order Taylor expansion: for infinitesimally small $\delta$ (i.e., $\delta \rightarrow 0$), we have
Informal proof for GD convergence

First-order Taylor expansion: for infinitesimally small $\delta$ (i.e., $\delta \to 0$), we have

$$\ell(w - \delta) = \ell(w) - \nabla \ell(w)^T \delta + \frac{\delta^2}{2}$$

$$\delta = \nabla \ell(w)$$
Informal proof for GD convergence

First-order Taylor expansion: for infinitesimally small $\delta$ (i.e., $\delta \to 0$), we have

$$\ell(w - \delta) = \ell(w) - \nabla \ell(w)^T \delta$$

Substitute $\delta = \eta \nabla \ell(w)$, with $\eta \to 0^+$
Informal proof for GD convergence

First-order Taylor expansion: for infinitesimally small $\delta$ (i.e., $\delta \to 0$), we have

$$\ell(w - \delta) = \ell(w) - \nabla \ell(w)^T \delta$$

Substitute $\delta = \eta \nabla \ell(w)$, with $\eta \to 0^+$

$$\ell(w - \eta \nabla \ell(w)) = \ell(w) - \eta \nabla \ell(w)^T (\nabla \ell(w))$$
Informal proof for GD convergence

First-order Taylor expansion: for infinitesimally small $\delta$ (i.e., $\delta \to 0$), we have

$$\ell(w - \delta) = \ell(w) - \nabla \ell(w)^T \delta$$

Substitute $\delta = \eta \nabla \ell(w)$, with $\eta \to 0^+$

$$\ell(w - \eta \nabla \ell(w)) = \ell(w) - \eta \nabla \ell(w)^T (\nabla \ell(w))$$

$$(w - B \nabla \ell(w)) \leq \ell(w), \text{ if } \nabla \ell(w) \neq 0$$

$$\|\nabla \ell(w)\|_2^2 > 0$$
Informal proof for GD convergence

First-order Taylor expansion: for infinitesimally small $\delta$ (i.e., $\delta \to 0$), we have

$$\ell(w - \delta) = \ell(w) - \nabla \ell(w)^T \delta$$

Substitute $\delta = \eta \nabla \ell(w)$, with $\eta \to 0^+$

$$\ell(w - \eta \nabla \ell(w)) = \ell(w) - \eta \nabla \ell(w)^T (\nabla \ell(w))$$

$$\|\nabla \ell(w)\|^2_2 > 0$$

i.e., with sufficiently small $\eta$, GD decrease obj value if $\nabla \ell(w) \neq 0$!
How to set learning rate $\eta$ in practice?

Large $\eta$ typically is bad and can lead to diverge
How to set learning rate $\eta$ in practice?

Large $\eta$ typically is bad and can lead to diverge.
How to set learning rate $\eta$ in practice?

Large $\eta$ typically is bad and can lead to diverge.

In theory, for convex loss, $\eta = \frac{c}{\sqrt{k}}$ guarantees convergence.
How to set learning rate $\eta$ in practice?

Large $\eta$ typically is bad and can lead to diverge.

In theory, for convex loss, $\eta = c/\sqrt{k}$ guarantees convergence.
Let’s summarize by applying GD to logistic regression

Recall the objective for LR:

\[
\min_w \sum_{i=1}^{n} \ln \left[ 1 + \exp \left( -y_i (w^T x_i) \right) \right] + \lambda \|w\|_2^2
\]

Initialize \( w^0 \in \mathbb{R}^d \)

Iterate until convergence:
Let’s summarize by applying GD to logistic regression

Recall the objective for LR:

\[
\min_w \sum_{i=1}^n \ln \left[ 1 + \exp \left( -y_i(w^T x_i) \right) \right] + \lambda \|w\|^2
\]

Initialize \(w^0 \in \mathbb{R}^d\)

Iterate until convergence:

1. Compute gradient \(g^t = \sum_i \frac{\exp(-y_ix_i^Tw^t)(-y_ix_i)}{1 + \exp(-y_ix_i^Tw^t)} + 2\lambda w^t\)
Let’s summarize by applying GD to logistic regression

Recall the objective for LR:

\[
\min_w \sum_{i=1}^{n} \ln \left[ 1 + \exp \left( -y_i(w^T x_i) \right) \right] + \lambda \|w\|_2^2
\]

Initialize \( w^0 \in \mathbb{R}^d \)

Iterate until convergence:

1. Compute gradient \( g^t = \sum_i \frac{\exp(-y_i x_i^T w^t)(-y_i x_i)}{1 + \exp(-y_i x_i^T w^t)} + 2\lambda w^t \)

2. Update (GD): \( w^{t+1} = w^t - \eta g^t \)