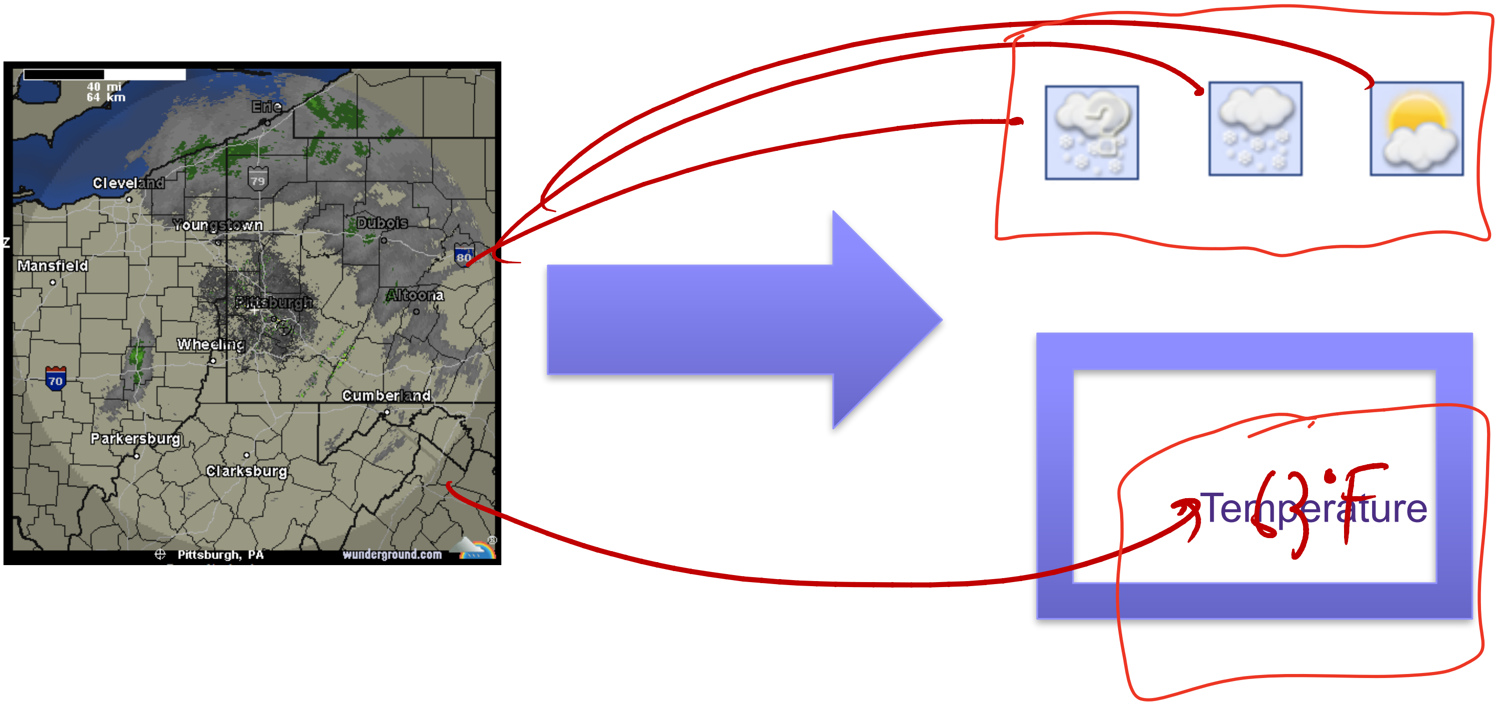
Classification
Logistic Regression

Thus far, regression:

predict a continuous value given some inputs

Weather prediction revisted



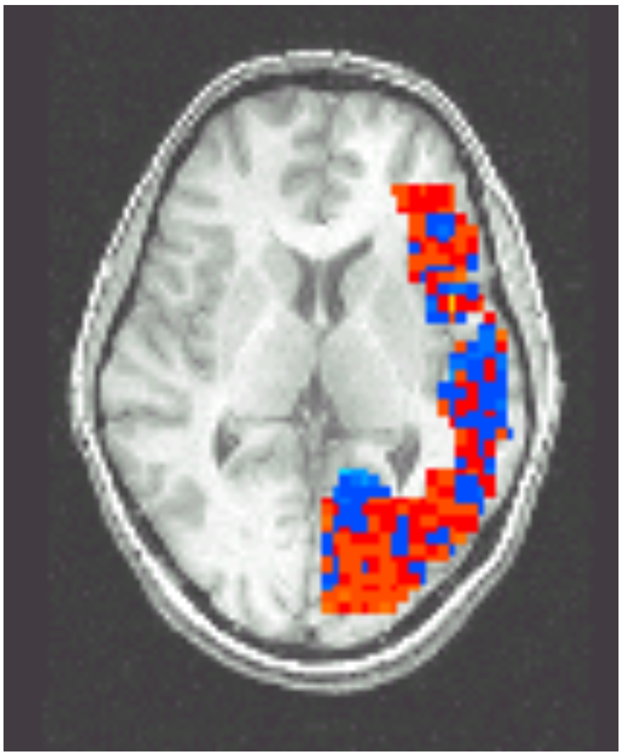
Reading Your Brain, Simple Example

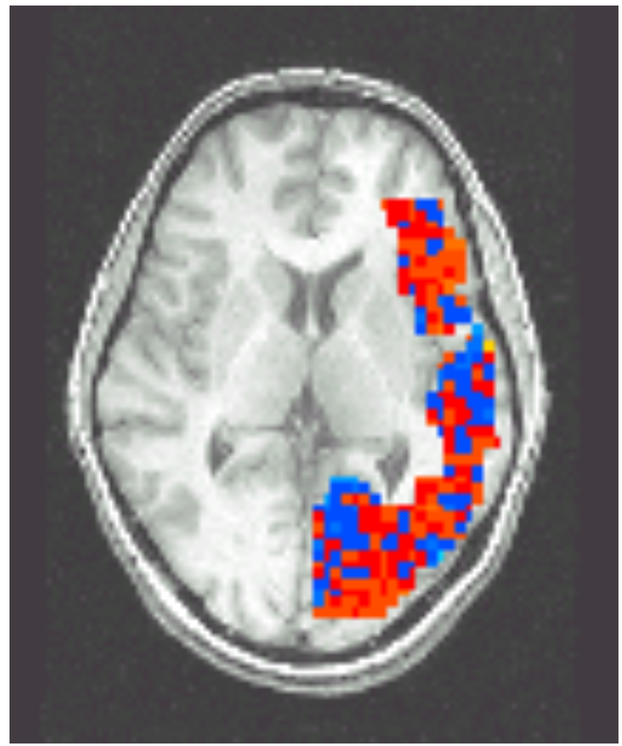
[Mitchell et al.]

Pairwise classification accuracy: 85%

Person

Animal





Classification

- Learn f: X -> Y
  - X - features
  - Y - target classes

$Y = \{1, \ldots, k\}$

- Loss Function
- Expected loss of f:

$$\ell(x, y) = \mathbb{1}[f(x) \neq y]$$

$$\mathbb{E}_{xy}[\ell(f)] = \mathbb{E}_{x}[\mathbb{E}_{Y|X}[\ell(x, y)]]$$

- Suppose you knew P(Y|X) exactly, how should you classify?
  - Bayes-Optimal classifier:

Classification

- Learn f: X -> Y
  - X - features
  - Y - target classes

- Loss Function

$$\ell(f(x), y) = \mathbf{1}\{f(x) \neq y\}$$

- Expected loss of f:

$$\mathbb{E}_{XY}[\mathbf{1}\{f(X) \neq Y\}] = \boxed{\mathbb{E}_X[\mathbb{E}_{Y|X}[\mathbf{1}\{f(x) \neq Y\}|X = x]]}$$

$$\mathbb{E}_{Y|X}[\mathbf{1}\{f(x) \neq Y\}|X = x] = \sum_{i=1} P(Y = i|X = x)\mathbf{1}\{f(x) \neq i\} = \sum_{i \neq f(x)} P(Y = i|X = x)$$

$$= 1 - P(Y = f(x)|X = x)$$

- Suppose you knew P(Y|X) exactly, how should you classify?
  - Bayes-Optimal classifier:

$$f(x) = \arg\max_y \boxed{\mathbb{P}(Y = y|X = x)}$$

Fixed x:

Binary Classification

- **Learn f: X -> Y**
  - **X - features**
  - **Y - target classes**

$$\boxed{Y \in \{0, 1\}}$$

- **Loss Function**

$$\ell(f(x), y) = \mathbf{1}\{f(x) \neq y\}$$

- **Expected loss of f:**

$$\mathbb{E}_{XY}[\mathbf{1}\{f(X) \neq Y\}] = \mathbb{E}_X[\mathbb{E}_{Y|X}[\mathbf{1}\{f(x) \neq Y\}|X = x]]$$

$$\mathbb{E}_{Y|X}[\mathbf{1}\{f(x) \neq Y\}|X = x] = \sum_{i=1}^{2} P(Y = i|X = x)\mathbf{1}\{f(x) \neq i\} = \sum_{i \neq f(x)} P(Y = i|X = x)$$

$$= 1 - P(Y = f(x)|X = x)$$

- **Suppose you knew P(Y|X) exactly, how should you classify?**
  - **Bayes-Optimal classifier:**

$$f(x) = \arg\max_y \mathbb{P}(Y = y|X = x)$$

Bayes Optimal Binary Classifier

$Y \in \{0, 1\}$

- Suppose you knew ~~P(Y|X) exactly~~, how should you classify?
  - Bayes-Optimal classifier:

$$\boxed{f(x) = \arg\max_y \mathbb{P}(Y = y | X = x)}$$

- Suppose we don't know P(Y|X), but have n iid examples

$$\boxed{\{(x_i, y_i)\}_{i=1}^n}$$

- What is a natural estimator for P(Y | X)?

Bayes Optimal Binary Classifier

- **Suppose we don't know P(Y|X), but have n iid examples**

$$\boxed{\{(x_i, y_i)\}_{i=1}^n}$$

$Y \in \{0, 1\}$

- **What is a natural estimator for P(Y | X)?**

Fix some $\tilde{x} \in X$

Suppose $\boxed{x_i = \tilde{x}}$ for $m \leq n$ samples

What is a natural estimator for $\boxed{\theta_* := \mathbb{P}(Y = 1 | X = \tilde{x})}$?

If k of the m labels are equal to $Y = 1$ then

?

$\boxed{k/m} = \theta^*_{MLE}$

Bayes Optimal Binary Classifier

- **Suppose we don't know P(Y|X), but have n iid examples**

$$\{(x_i, y_i)\}_{i=1}^n$$

$$Y \in \{0,1\}$$

- **What is a natural estimator for argmax_y P(Y = y | X)?**

If $X = \{0,1\}^d$, or is generally discrete

$$\hat{f}(x) = \arg\max_{y \in \{0,1\}} \boxed{\frac{\sum_{i=1}^n \mathbf{1[x_i = x, y_i = y]}}{\sum_{i=1}^n \mathbf{1[x_i = x]}}}$$

Issues?

What if I don't see x?

2^d many feature vectors

If $n < (\ll) \; 2^d$

Bayes Optimal Binary Classifier

- **What is a natural estimator for argmax_y P(Y = y | X)?**

If $X = \{0,1\}^d$, or is generally discrete $\quad Y \in \{0,1\}$

$$\hat{f}(x) = \arg\max_{y \in \{0,1\}} \frac{\sum_{i=1}^n \mathbf{1[x_i = x, y_i = y]}}{\sum_{i=1}^n \mathbf{1[x_i = x]}}$$

Issues?

2^d possible inputs, for small d requires huge n

To make predictions for unseen inputs (xs),

need a **general** model for $\mathbb{P}(Y = 1 | X = x)$

Process

Decide on a **model**

Find the function which fits the data best

Choose a loss function

Pick the function which minimizes loss on data

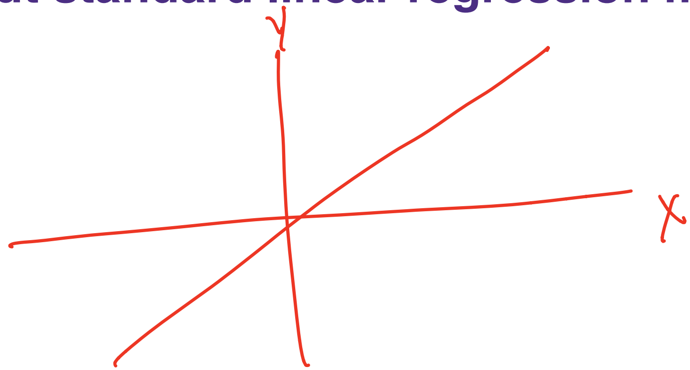
Use function to make prediction on new examples

Decide on a model, Binary Classification

To make predictions for unseen inputs (xs),

need a **general** model for $\boxed{\mathbb{P}(Y = 1|X = x)}$

- **What about standard linear regression model?**



Linear regression maps to $[-\infty, \infty]$

- **Need to map real values to [0,1]**
  - **We call such maps "link functions"**

Logistic Regression

[Binary]

Actually classification, not regression :)

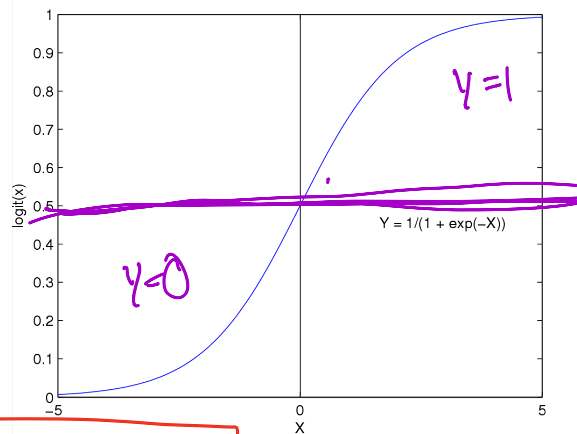
Learn $\mathbb{P}(Y = 1|X = x)$ using $\sigma(w^T x)$, for link function $\sigma = \boxed{\frac{1}{1 + exp(-z)}}$

Logistic function(or Sigmoid):

$$\boxed{\mathbb{P}[Y = 1|X = x, w]} = \sigma(w^T x) = \boxed{\frac{1}{1 + \exp(-w^T x)}}$$

$$\boxed{\mathbb{P}[Y = 0|X = x, w]} = \boxed{1 - \sigma(w^T x)} = \frac{\exp(-w^T x)}{1 + \exp(-w^T x)}$$

$$= \frac{1}{1 + \exp(w^T x)}$$

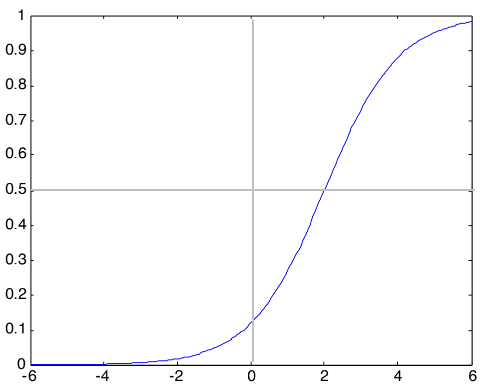


Features can be discrete or continuous!

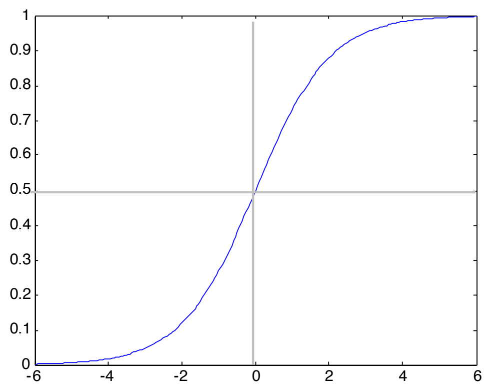
Understanding the sigmoid

$$\sigma(\boxed{w_0 + \sum_k w_k x_k}) = \frac{1}{1 + e^{w_0 + \sum_k w_k x_k}}$$

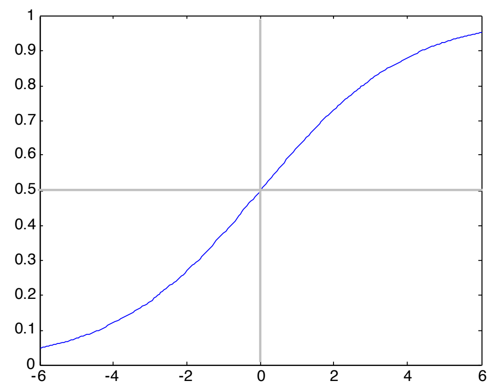
w_0=-2, w_1=-1



w_0=0, w_1=-1



w_0=0, w_1=-0.5



Sigmoid for binary classes

$$\mathbb{P}(Y = 0|w, X) = \frac{1}{1 + \exp(w_0 + \sum_k w_k X_k)}$$

$$\mathbb{P}(Y = 1|w, X) = 1 - \mathbb{P}(Y = 0|w, X) = \frac{\exp(w_0 + \sum_k w_k X_k)}{1 + \exp(w_0 + \sum_k w_k X_k)}$$

$$\frac{\mathbb{P}(Y = 1|w, X)}{\mathbb{P}(Y = 0|w, X)} = \frac{1}{\exp(-w^T x)} = \exp(w^T x) \underset{Y=0}{\overset{Y=1}{\gtrless}} 1$$

Sigmoid for binary classes

$$\mathbb{P}(Y=0|w,X) = \frac{1}{1+\exp(w_0+\sum_k w_k X_k)}$$

$$\mathbb{P}(Y=1|w,X) = 1-\mathbb{P}(Y=0|w,X) = \frac{\exp(w_0+\sum_k w_k X_k)}{1+\exp(w_0+\sum_k w_k X_k)}$$

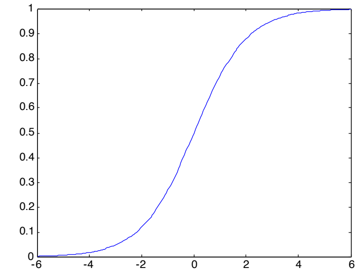
$$\frac{\mathbb{P}(Y=1|w,X)}{\mathbb{P}(Y=0|w,X)} = \exp(w_0+\sum_k w_k X_k)$$

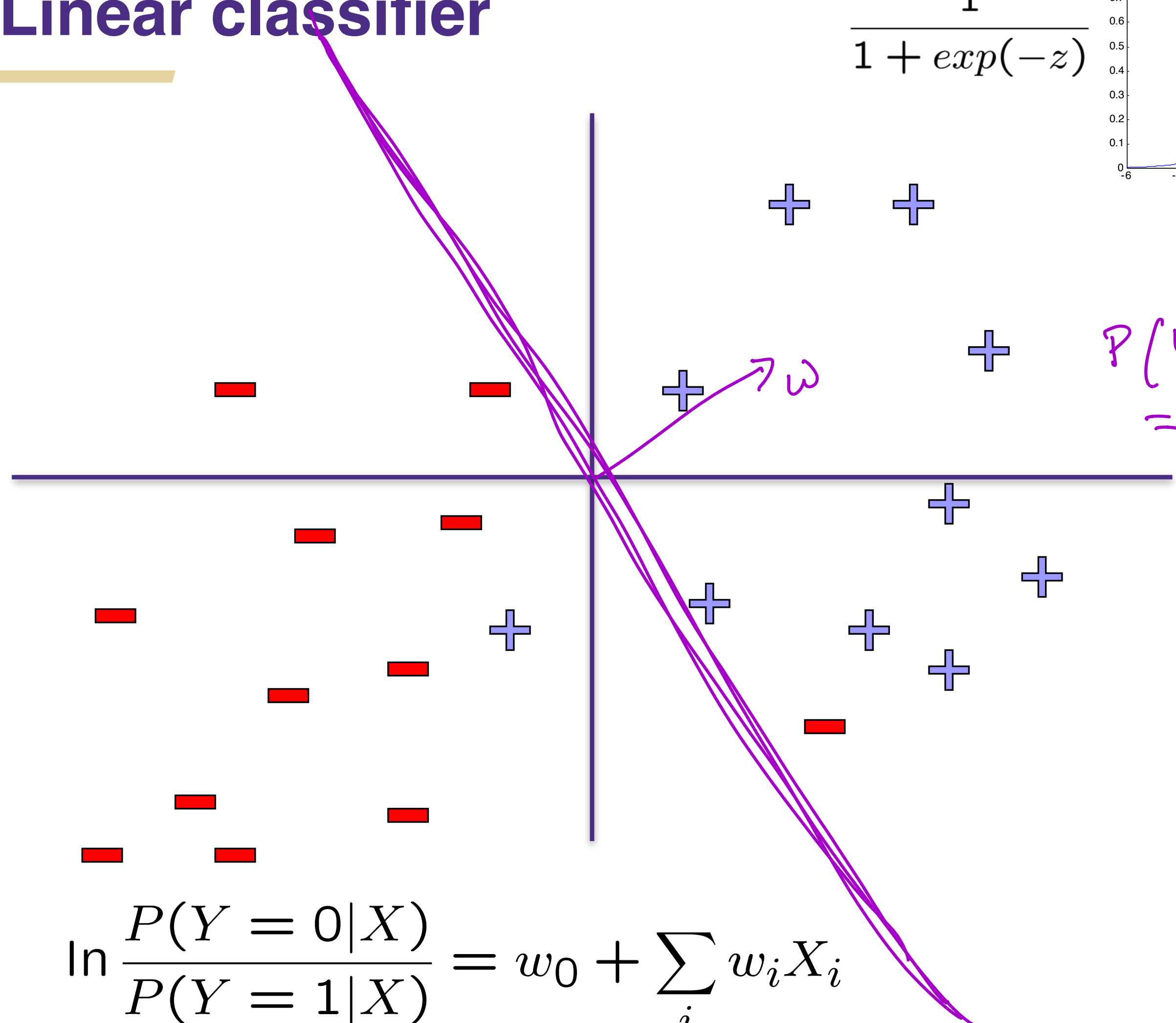
Linear Decision Rule!

$$\log\frac{\mathbb{P}(Y=1|w,X)}{\mathbb{P}(Y=0|w,X)} = w_0+\sum_k w_k X_k$$

Logistic Regression – a Linear classifier

$$\frac{1}{1+exp(-z)}$$





$P(Y=1|X=x,w)$
$= \sigma(w^T x)$

$$\ln \frac{P(Y=0|X)}{P(Y=1|X)} = w_0 + \sum_i w_i X_i$$

Process

Decide on a **model**

Find the function which fits the data best

Choose a loss function

Pick the function which minimizes loss on data

Use function to make prediction on new examples

Loss function: Conditional Likelihood

- **Have a bunch of iid data:** $\{(x_i, y_i)\}_{i=1}^n \quad x_i \in \mathbb{R}^d, \ \boxed{y_i \in \{-1, 1\}}$

$$P(Y=0|x,w) := P(Y = -1|x, w) = \frac{1}{1 + \exp(w^T x)}$$

$$P(Y = 1|x, w) = \frac{\exp(w^T x)}{1 + \exp(w^T x)}$$

- **This is equivalent to:**

$$P(Y = y|x, w) = \frac{1}{1 + \exp(-y\, w^T x)}$$

- **So we can compute the maximum likelihood estimator:**

$$\boxed{\widehat{w}_{MLE} = \arg\max_w \prod_{i=1}^n P(y_i|x_i, w)}$$

Loss function: Conditional Likelihood

- **Have a bunch of iid data:** $\{(x_i, y_i)\}_{i=1}^n \quad x_i \in \mathbb{R}^d, \quad y_i \in \{-1, 1\}$

$$\widehat{w}_{MLE} = \arg\max_w \prod_{i=1}^n P(y_i | x_i, w) \qquad P(Y = y | x, w) = \frac{1}{1 + \exp(-y\, w^T x)}$$

$$= \arg\min_w \sum_{i=1}^n \log(1 + \exp(-y_i\, x_i^T w))$$

Loss function: Conditional Likelihood

- **Have a bunch of iid data:** $\{(x_i, y_i)\}_{i=1}^n \quad x_i \in \mathbb{R}^d, \quad y_i \in \{-1, 1\}$

$$P(Y = y|x, w) = \frac{1}{1 + \exp(-y\, w^T x)}$$

$$\begin{aligned}\widehat{w}_{MLE} &= \arg\max_w \prod_{i=1}^n P(y_i|x_i, w) \\ &= \arg\min_w \sum_{i=1}^n \log(1 + \exp(-y_i\, x_i^T w))\end{aligned}$$

Logistic Loss: $\ell_i(w) = \log(1 + \exp(-y_i\, x_i^T w))$

Squared error Loss: $\ell_i(w) = (y_i - x_i^T w)^2$ (MLE for Gaussian noise)

Process

Decide on a **model**

Find the function which fits the data best

Choose a loss function

Pick the function which minimizes loss on data

Use function to make prediction on new examples

Loss function: Conditional Likelihood

- **Have a bunch of iid data:** $\{(x_i, y_i)\}_{i=1}^n \quad x_i \in \mathbb{R}^d, \quad y_i \in \{-1, 1\}$

$$P(Y = y|x, w) = \frac{1}{1 + \exp(-y\, w^T x)}$$

$$\widehat{w}_{MLE} = \arg\max_w \prod_{i=1}^n P(y_i|x_i, w)$$

$$= \arg\min_w \sum_{i=1}^n \log(1 + \exp(-y_i\, x_i^T w)) = J(w)$$

What does $J(w)$ look like? Is it convex?

Loss function: Conditional Likelihood

$$\{(x_i, y_i)\}_{i=1}^n \quad x_i \in \mathbb{R}^d, \quad y_i \in \{-1, 1\}$$

$$\widehat{w}_{MLE} = \arg\max_w \prod_{i=1}^n P(y_i | x_i, w) \qquad P(Y = y | x, w) = \frac{1}{1 + \exp(-y\, w^T x)}$$

$$= \arg\min_w \sum_{i=1}^n \log(1 + \exp(-y_i\, x_i^T w)) = J(w)$$

Good news: $J(\mathbf{w})$ is convex function of **w**, no local optima problems

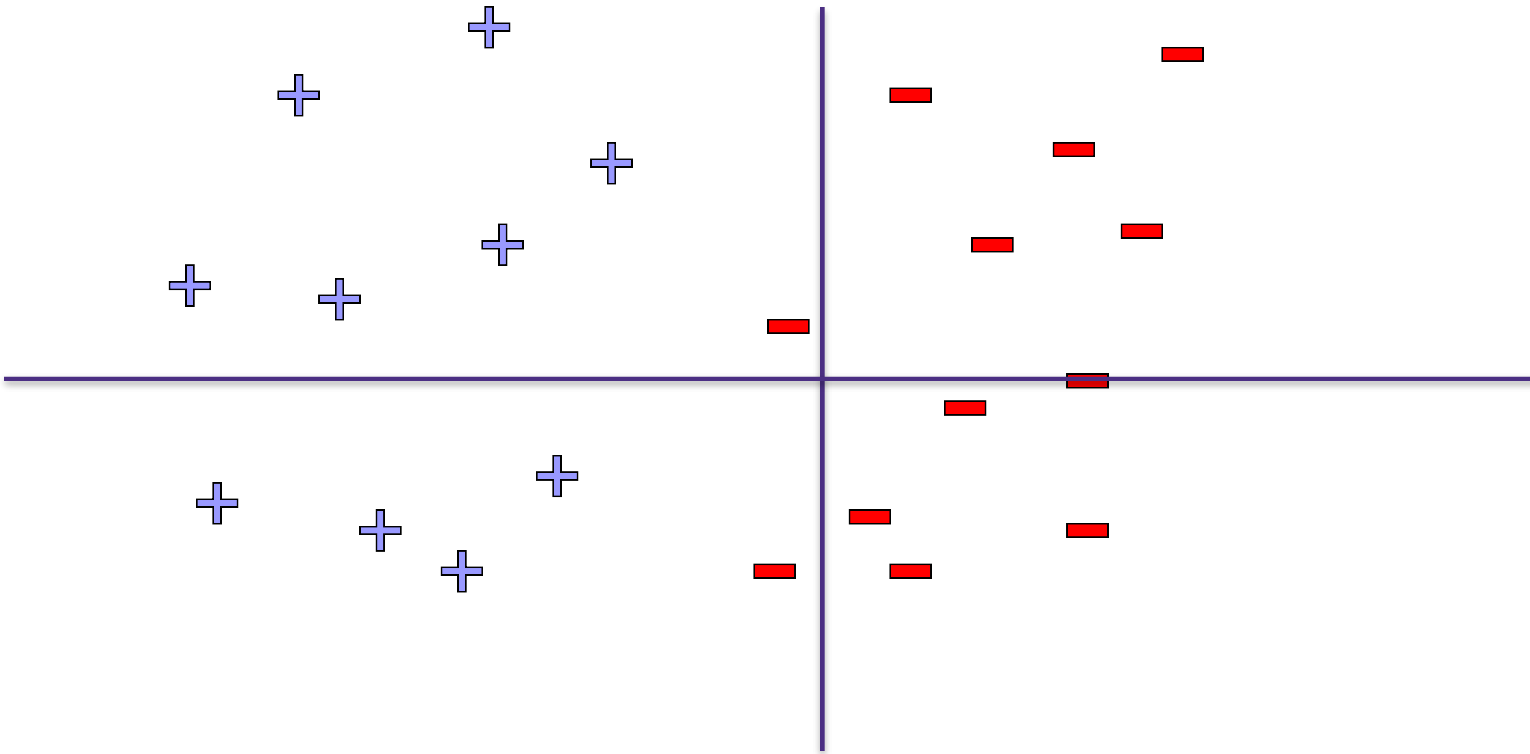
Bad news: no closed-form solution to maximize $J(\mathbf{w})$

Good news: convex functions easy to optimize

Linear Separability

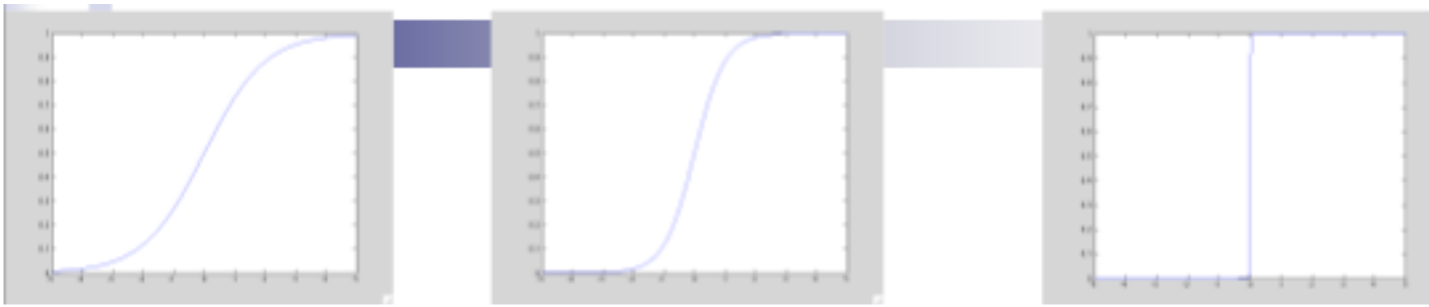
$$\arg\min_{w} \sum_{i=1}^{n} \log(1 + \exp(-y_i\, x_i^T w))$$

When is this loss small?



Large parameters → Overfitting

When data is linearly separable, weights $\Rightarrow \infty$



$$\frac{1}{1+e^{-x}} \qquad \frac{1}{1+e^{-2x}} \qquad \frac{1}{1+e^{-100x}}$$

Overfitting

Penalize high weights to prevent overfitting?

Regularized Conditional Log Likelihood

Add a penalty to avoid high weights/overfitting?:

$$\arg\min_{w,b} \sum_{i=1}^{n} \log\left(1 + \exp(-y_i\left(x_i^T w + b\right))\right) + \lambda ||w||_2^2$$

Be sure to not regularize the offset b!