ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

inf | Informatik Computer Science

Introduction to Machine Learning

Acting under uncertainty: Bayesian decision theory

Prof. Andreas Krause
Learning and Adaptive Systems (las.ethz.ch)

Acting under uncertainty

- So far, have seen how we can interpret supervised learning as fitting probabilistic models of the data
- Next, we‘ll see how we can use the estimated models to make decisions

Acting under uncertainty

- Suppose we have estimated a logistic regression model (say, for spam filtering), and obtain *P(Y=spam|X)* $= p$
- Further suppose we have three actions: **Spam**, **NotSpam**, and **AskUser**
- Which should we pick?

cost

Actions	Y=spam	Y not spam
S	0	10
N	1	0
A	.5	.5

Expected cost

Action	$p = 0.2$	$p = 0.8$
S	$.2 \times 0 + .8 \cdot 10 = 8$	$.8 \times 0 + .2 \cdot 10 = 2$
N	$.2 \times 1 + .8 \cdot 0 = \boxed{.2}$	$.8 \times 1 + 0.2 \cdot 0 = .8$
A	$.5$	$\boxed{.5}$

Bayesian decision theory

- Given: e.g. $\{+1,-1\}, \{1,\dots,c\}, \mathbb{R}$
  - Conditional distribution over labels $P(y \mid \mathbf{x})$ $y \in \mathcal{Y}$
  - Set of actions $\mathcal{A}$ e.g. $A: \{S, N, A\}$ $\mathcal{A} \neq \mathcal{Y}$
  - Cost function $C : \mathcal{Y} \times \mathcal{A} \to \mathbb{R}$
- Bayesian Decision Theory recommends to pick the action that minimizes the expected cost

$$a^* = \arg\min_{a \in \mathcal{A}} \mathbb{E}_y[C(y,a) \mid \mathbf{x}]$$

- If we had access to the **true distribution** $P(y \mid \mathbf{x})$ this decision implements the Bayesian optimal decision
- In practice, can only **estimate** it, e.g., (logistic) regression

Recall: Logistic regression

- Learning:
  - Find optimal weights by minimizing logistic loss + regularizer

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \sum_{i=1}^{n} \log\Big(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i)\Big) + \lambda ||\mathbf{w}||_2^2$$

$$= \arg\max_{\mathbf{w}} P(\mathbf{w} \mid \mathbf{x}_1, \ldots, \mathbf{x}_n, y_1, \ldots, y_n)$$

- Classification:
  - Use conditional distribution

$$P(y \mid \mathbf{x}, \hat{\mathbf{w}}) = \frac{1}{1 + \exp(-y \hat{\mathbf{w}}^T \mathbf{x})}$$

Optimal decisions for logistic regression

- Est. cond. dist.: $\hat{P}(y \mid \mathbf{x}) = \mathrm{Ber}(y; \sigma(\hat{\mathbf{w}}^T \mathbf{x}))$
- Action set: $\mathcal{A} = \{+1, -1\}$
- Cost function: $C(y, a) = \underbrace{[y \neq a]}$ $= \begin{cases} 1 & \text{if } y \neq a \\ 0 & \text{o/w.} \end{cases}$

- Then the action that minimizes the expected cost

$$a^* = \arg\min_{a \in \mathcal{A}} \underbrace{\mathbb{E}_y [C(y, a) \mid \mathbf{x}]} = \sum_y P(y|x)[y \neq a]$$

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

is the most likely class:

$$a^* = \arg\max_y \hat{P}(y \mid \mathbf{x}) = \mathrm{sign}(\mathbf{w}^T \mathbf{x})$$

$$a^* = \arg\min_a \frac{1}{1 + \exp(a \cdot w^T x)}$$

$$= \arg\max_a 1 + \exp(a \cdot w^T x)$$

$$= \arg\max_{a \in \{-1, +1\}} a \cdot w^T x = \mathrm{sign}(w^T x)$$

Asymmetric costs

- Est. cond. dist.: $\hat{P}(y \mid \mathbf{x}) = \text{Ber}(y; \sigma(\hat{\mathbf{w}}^T \mathbf{x}))$
- Action set: $\mathcal{A} = \{+1, -1\}$
- Costs:

$$C(y,a) = \begin{cases} c_{FP} & \text{if } y = -1 \text{ and } a = +1 \\ c_{FN} & \text{if } y = +1 \text{ and } a = -1 \\ 0 & \text{otherwise} \end{cases}$$

- Then the action that minimizes the expected cost is

$$C_+ = \mathbb{E}_y[C(y,+1) \mid x] = P(y=-1 \mid x) \cdot C_{FP} + \underbrace{P(y=+1 \mid x)}_{p} \cdot 0$$

$$= (1-p) \cdot C_{FP}$$

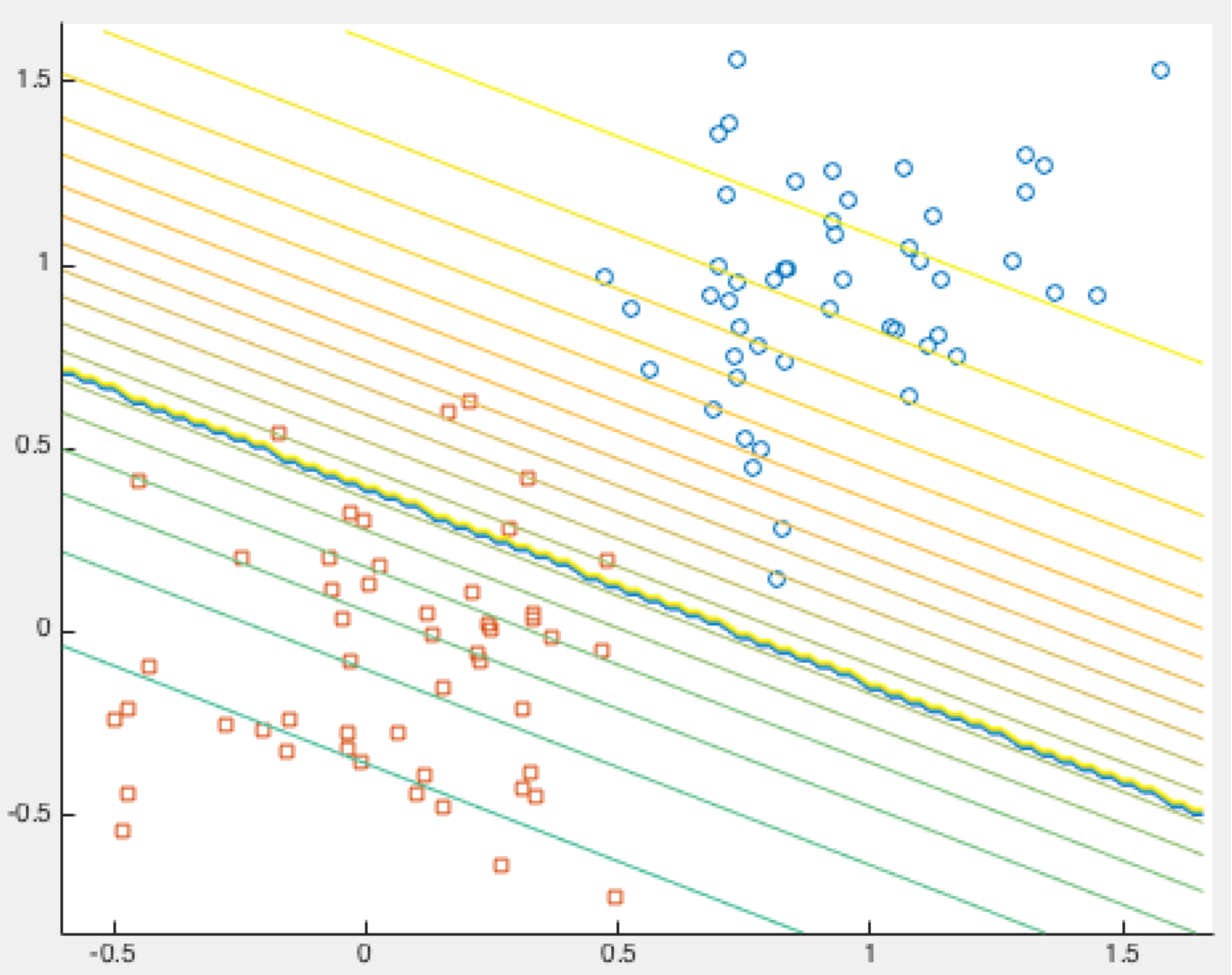
$$C_- = \mathbb{E}[C(y,-1) \mid x] = p \cdot C_{FN} + (1-p) \cdot 0 = p \cdot C_{FN}$$

predict $+1 \Leftrightarrow C_+ < C_-$

$\Leftrightarrow (1-p) C_{FP} < p C_{FN}$

$\Leftrightarrow p > \frac{C_{FP}}{C_{FP} + C_{FN}}$

Demo: Asymmetric costs



$$\frac{C_{FP}}{C_{FP}+C_{FN}}$$

„Doubtful“ logistic regression

- Est. cond. dist.: $\hat{P}(y \mid \mathbf{x}) = \text{Ber}(y; \sigma(\hat{\mathbf{w}}^T \mathbf{x}))$
- Action set: $\mathcal{A} = \{+1, -1, D\}$ "I don't know"
- Cost function: $$C(y,a) = \begin{cases} [y \neq a] & \text{if } a \in \{+1, -1\} \\ c & \text{if } a = D \end{cases}$$
- Then the action that minimizes the expected cost

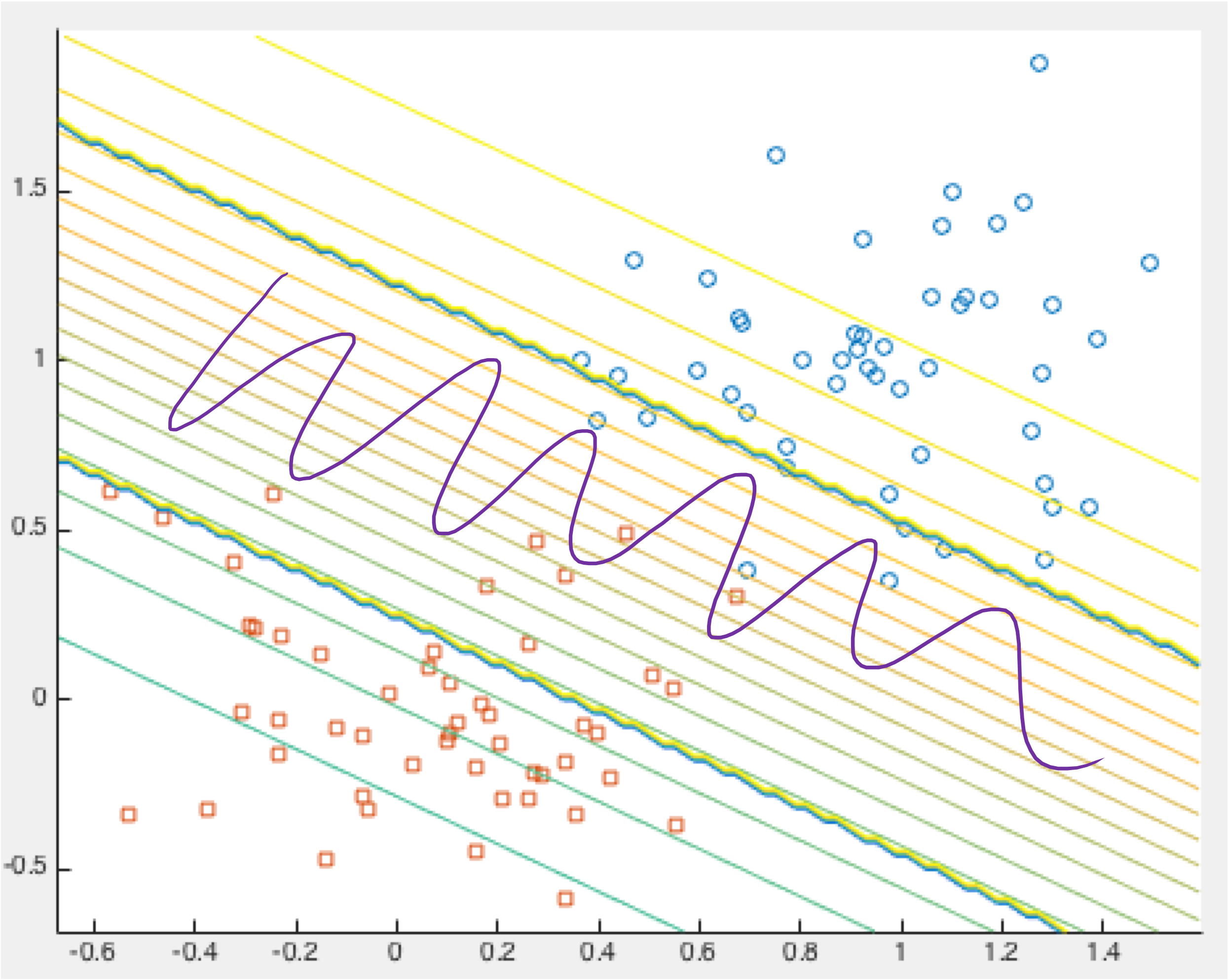
$$a^* = \arg\min_{a \in \mathcal{A}} \mathbb{E}_y[C(y,a) \mid \mathbf{x}]$$

is given by:

$$a^* = \begin{cases} y & \text{if } \hat{P}(y \mid \mathbf{x}) \geq 1 - c \\ D & \text{otherwise} \end{cases}$$

- I.e., pick most likely class only if confident enough!

Demo: Doubtful Logistic Regression



doubt

Optimal decisions for LS regression

- Est. cond. dist.: $\hat{P}(y \mid \mathbf{x}) = \mathcal{N}(y; \hat{\mathbf{w}}^T \mathbf{x}, \sigma^2)$
- Action set: $\mathcal{A} = \mathbb{R}$
- Cost function: $C(y, a) = (y - a)^2$

- Then the action that minimizes the expected cost

$$a^* = \arg\min_{a \in \mathcal{A}} \underbrace{\mathbb{E}_y[C(y, a) \mid \mathbf{x}]} = \mathbb{E}_y[(y-a)^2 \mid x]$$

$$\Rightarrow \frac{\partial}{\partial a} \mathbb{E}_y[\ -\text{''}-\ ] = 0$$

$$\mathbb{E}_y\left[\frac{\partial}{\partial a}(y-a)^2 \mid x\right]$$

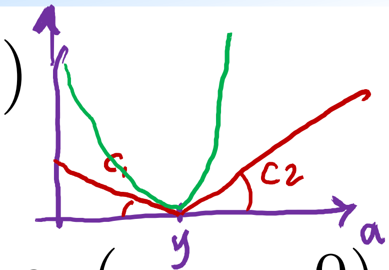
is the conditional mean:

$$a^* = \mathbb{E}_y[y \mid \mathbf{x}] = \int \hat{P}(y \mid \mathbf{x}) dy \quad \ldots \Rightarrow a^* = w^T x$$

$$= \hat{\mathbf{w}}^T \mathbf{x}$$

Example: Asymmetric cost for regression

- Est. cond. dist.: $\hat{P}(y \mid \mathbf{x}) = \mathcal{N}(y; \hat{\mathbf{w}}^T \mathbf{x}, \sigma^2)$
- Action set: $\mathcal{A} = \mathbb{R}$
- Cost : $C(y,a) = c_1 \underbrace{\max(y-a,0)}_{\text{underestimation}} + c_2 \underbrace{\max(a-y,0)}_{\text{overestimation}}$



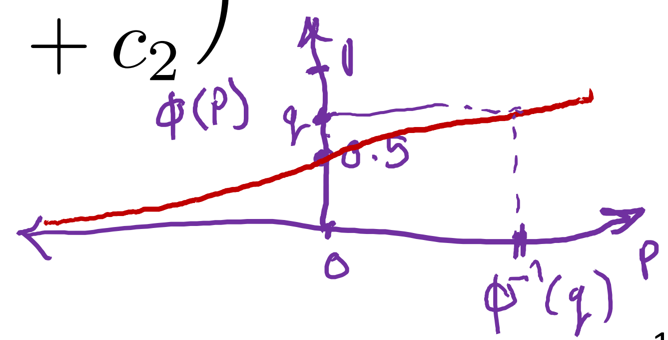
- Then the action that minimizes the expected cost

$$a^* = \arg\min_{a \in \mathcal{A}} \mathbb{E}_y[C(y,a) \mid \mathbf{x}]$$

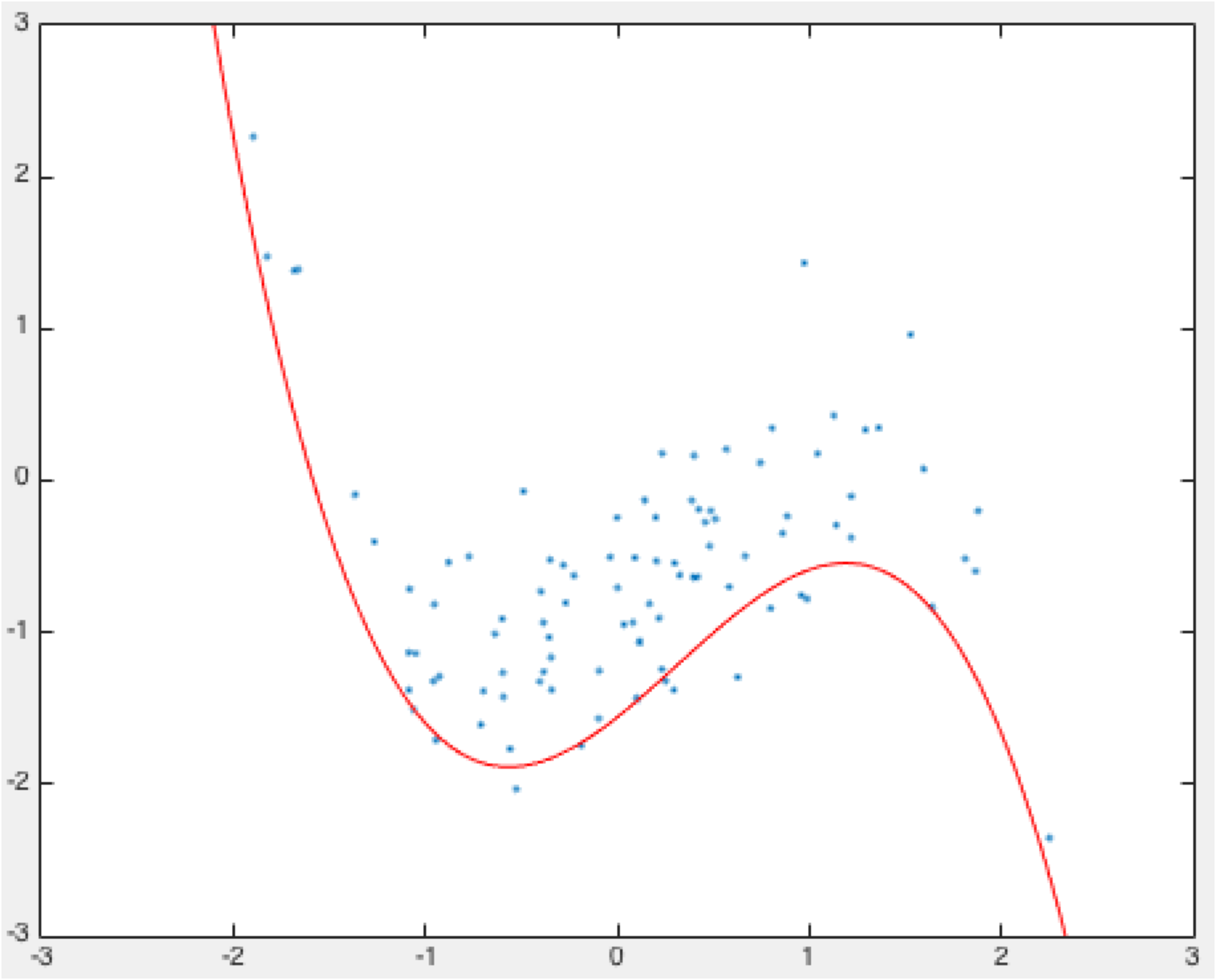
is:

$$a^* = \hat{\mathbf{w}}^T \mathbf{x} + \sigma \cdot \Phi^{-1}\left(\frac{c_1}{c_1 + c_2}\right)$$

inverse Gaussian CDF

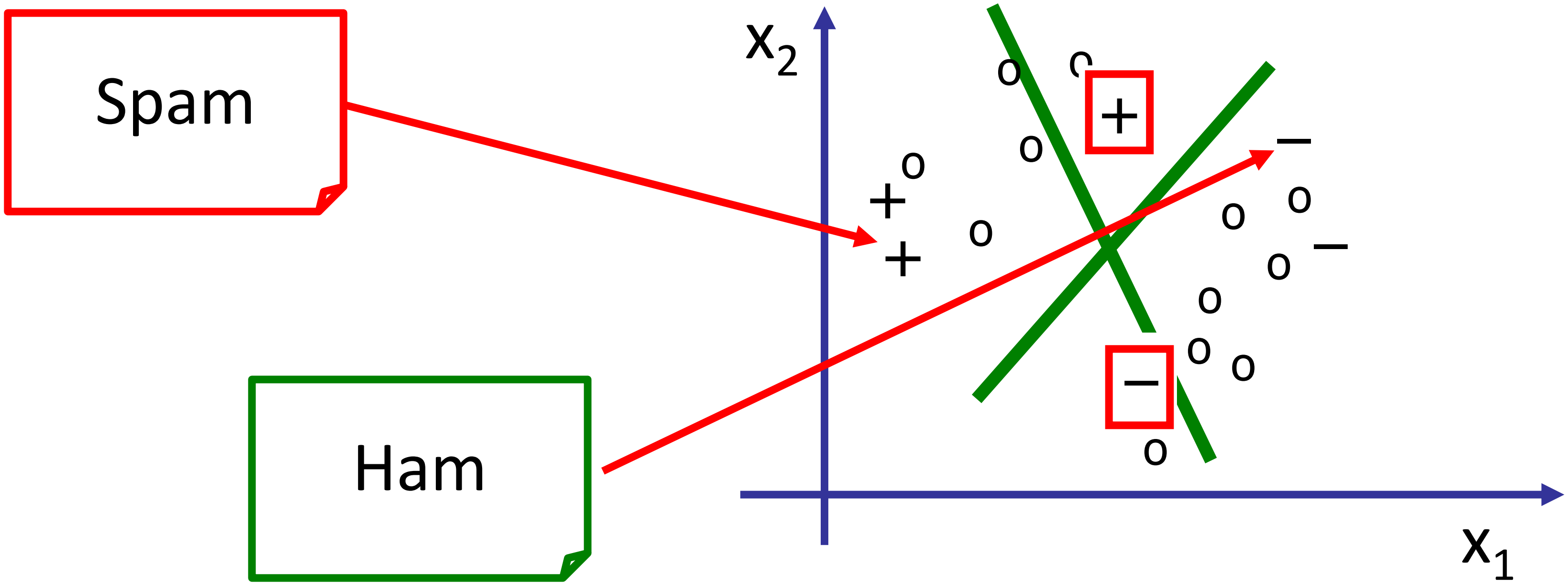


Demo: Asymmetric cost for regression



$$\frac{c_1}{c_1 + c_2}$$

Outlook: Active learning



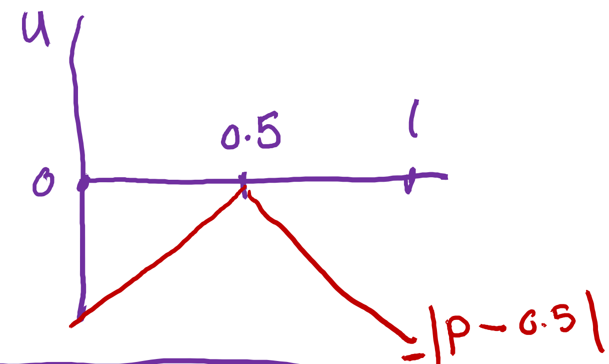
- Labels are expensive (need to ask expert)
- **Want to minimize the number of labels**

Uncertainty sampling

- Simple strategy: Always pick the example that we are most uncertain about

- Given $D = \{(x_i, y_i)\}_{i=1}^{n}$ estimate $\hat{P}(y|x)$
- For every unlabeled instance x_j

$$\hat{P}(y_j = +1 \mid x_j) = p$$



- Uncertainty score, u_j
- $j^* = \arg\max_j u_j$

Example: LogReg

$$u_j = -|w^T x_j|$$

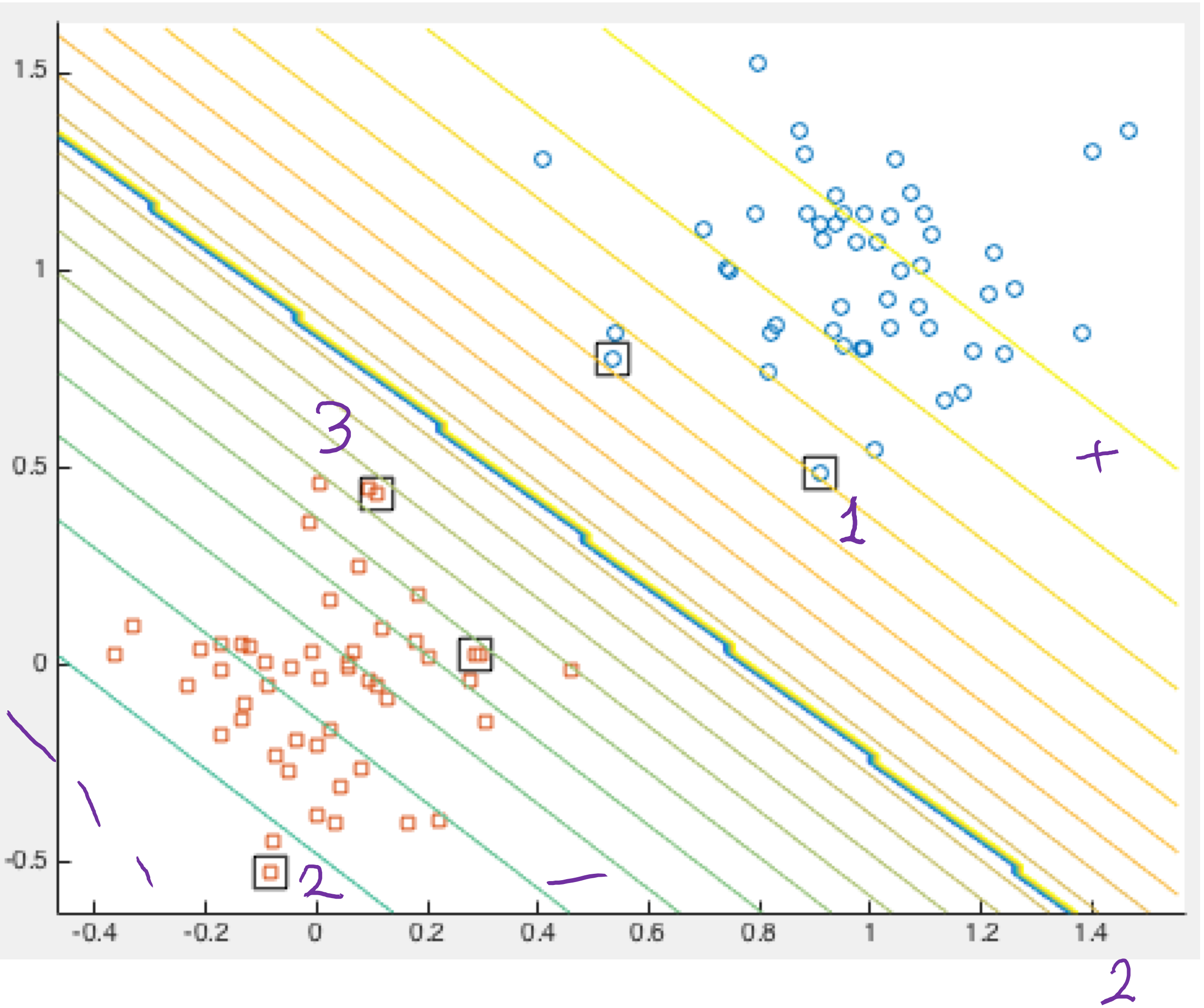
Uncertainty sampling

- **Given**: Pool of unlabeled examples $D_X = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$
- Also maintain labeled data set *D*, initially empty
- For *t=1,2,3,...*
  - Estimate $\hat{P}(Y_i \mid \mathbf{x}_i)$ given current data *D*
  - Pick unlabeled example that we are most uncertain about

$$i_t \in \arg\min_i \left|0.5 - \hat{P}(Y_i \mid \mathbf{x}_i)\right|$$

  - Query label y_{i_t} and set $D \leftarrow D \cup \{(\mathbf{x}_{i_t}, y_{i_t})\}$

Demo: Uncertainty sampling



Further comments

- Active learning violates i.i.d. assumption!
- Can get stuck with bad models
- More advanced selection criteria available
  - E.g.: query point that reduces uncertainty of other points as much as possible

Deriving decision rules

- Bayesian decision theory provides a principled way to derive decision rules from conditional distributions $P(Y|X)$
- Standard rules arise as special cases:
  - **Linear regression**: $\hat{\mathbf{w}}^T\mathbf{x}$
  - **Logistic regression**: $\text{sign}(\hat{\mathbf{w}}^T\mathbf{x})$
- Can accommodate more complex settings
  - „Doubt“ (i.e., requiring sufficient confidence)
  - Asymmetric losses
  - Active learning
  - ...

Summary: Learning through MAP inference

- Start with statistical assumptions on data:
  Data points modeled as iid (can be relaxed)
- Choose likelihood function
  - **Examples**: Gaussian, student-t, logistic, exponential, ...

    ➔ loss function
- Choose prior
  - **Examples**: Gaussian, Laplace, exponential, ...

    ➔ regularizer
- Optimize for MAP parameters
- Choose hyperparameters (i.e., variance, etc.) through cross-validation
- Make predictions via Bayesian Decision Theory

What you should be able to do

- Understand and apply logistic regression and its variants
- Relate logistic regression and Perceptron/SVM
- Derive MAP estimation problems for different priors and likelihood functions
- Solve resulting optimization problems by applying gradient descent
- Derive decision rules from cost functions via Bayesian decision theory
- Apply uncertainty sampling for binary classification

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

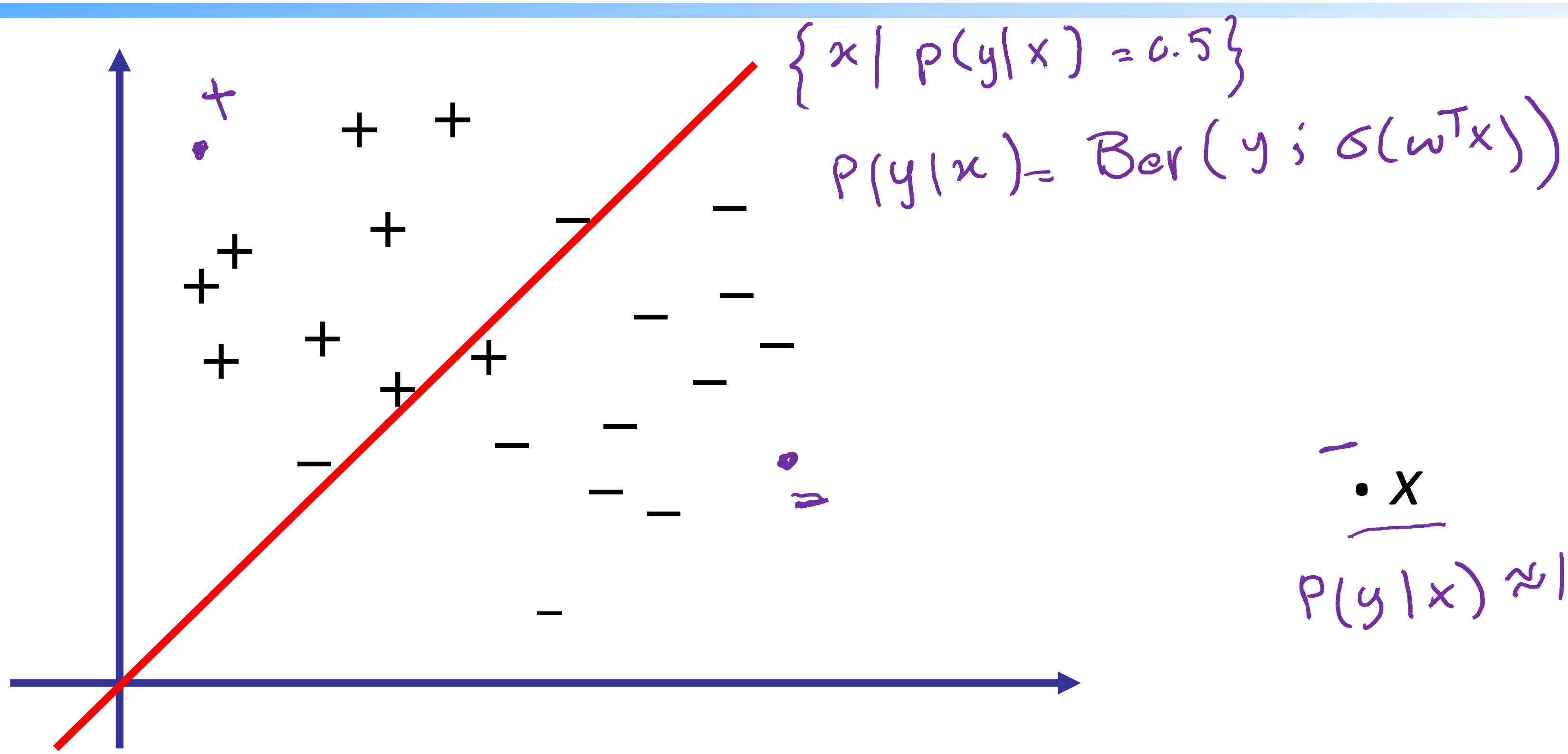
inf | Informatik Computer Science

Introduction to Machine Learning

Discriminative vs. Generative Modeling

Prof. Andreas Krause
Learning and Adaptive Systems (las.ethz.ch)

Motivating example: Logistic regression



- What will logistic regression predict for data point *x*?
- Logistic regression can be overconfident about labels for outliers

Discriminative modeling

- So far, we have considered learning methods that estimate conditional distributions

$$P(y \mid \mathbf{x})$$

- **Examples**: Linear regression, logistic regression, etc.

- Such models *do not* attempt to model $P(\mathbf{x})$
- Thus, they will not be able to detect outliers (i.e., „unusual" points for which *P(**x**)* is very small)

Discriminative vs. Generative models

- Discriminative models aim to estimate

$$P(y \mid \mathbf{x})$$

- Generative models aim to estimate joint distribution

$$P(y, \mathbf{x})$$

- Can derive conditional from joint distribution, but not vice versa!

$$P(y, x) \rightsquigarrow P(y|x) = \frac{P(x,y)}{P(x)}$$

$$P(x) \to \sum_{y'} P(x, y')$$

Typical approach to generative modeling

$P(x, y) = P(x|y)P(y)$ (chain rule)

$= P(y|x)P(x)$

1. Estimate prior on labels $P(y)$

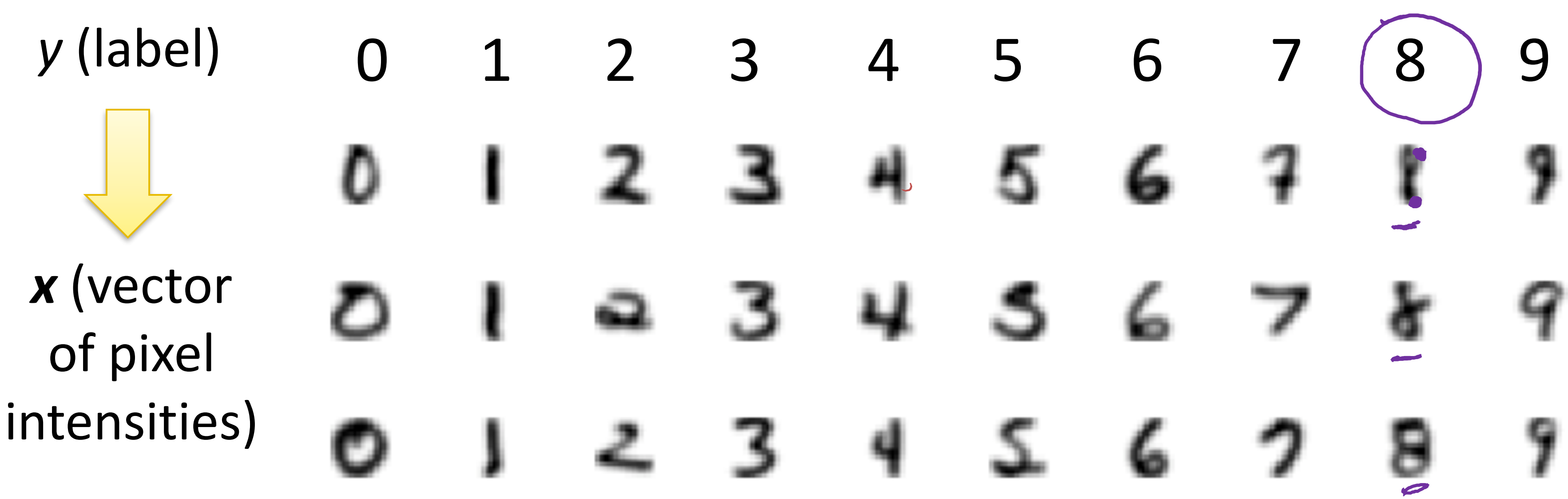
2. Estimate conditional distribution for each class *y* $P(\mathbf{x} \mid y)$

3. Obtain predictive distribution using Bayes' rule:

$$P(y \mid \mathbf{x}) = \frac{1}{\underset{P(x)}{Z}} \underbrace{P(y)P(\mathbf{x} \mid y)}_{P(y,x)}$$

A note on generative modeling

- Generative modeling attempts to infer the process, according to which examples are generated $P(x,y)$
- First generate class label $P(y)$
- Then, generate features given class $P(\mathbf{x} \mid y)$



Example: Naive Bayes Model

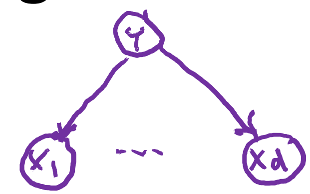
- Model class label as generated from categorical variable

$$P(Y = y) = p_y \qquad\qquad y \in \mathcal{Y} = \{1, \ldots, c\}$$

$$\forall y = 1, \ldots, c \quad p_y \geq 0 \,, \quad \sum_{y=1}^{c} p_y = 1$$

- Model features as conditionally independent given Y

$$P(X_1, \ldots, X_d \mid Y) = \prod_{i=1}^{d} P(X_i \mid Y)$$



$$P(X_1 = x_1, \ldots, X_d = x_d \mid Y = y) = \prod_{i=1}^{d} P(X_i = x_i \mid Y = y)$$

- I.e., given class label, each feature is „generated“ independently of the other features.
- Need to still specify feature distributions $P(X_i \mid Y)$

Example: Gaussian *Naive* Bayes classifiers

- Model class label as generated from categorical variable

$$P(Y = y) = p_y \qquad\qquad y \in \mathcal{Y} = \{1, \ldots, c\}$$

- Model features by (conditionally) independent Gaussians

$$P(x_i \mid y) = \mathcal{N}(x_i \mid \mu_{y,i}, \sigma^2_{y,i})$$

depend on class y and feature i

$i \in \{1, \ldots, d\}$

- How do we estimate the parameters?