Machine Learning – Lecture 1

Introduction

09.10.2019

Bastian Leibe
RWTH Aachen
http://www.vision.rwth-aachen.de/

leibe@vision.rwth-aachen.de

Organization

- Lecturer
  - Prof. Bastian Leibe (leibe@vision.rwth-aachen.de)
- Assistants
  - Ali Athar (athar@vision.rwth-aachen.de)
  - Sabarinath Mahadevan (mahadevan@vision.rwth-aachen.de)
- Course webpage
  - http://www.vision.rwth-aachen.de/courses/
  - Slides will be made available on the webpage and in moodle
  - Lecture recordings as screencasts will be available via moodle
- Please subscribe to the lecture in rwth online!
  - Important to get email announcements and moodle access!

Language

- Official course language will be English
  - If at least one English-speaking student is present.
  - If not… you can choose.
- However…
  - Please tell me when I'm talking too fast or when I should repeat something in German for better understanding!
  - You may at any time ask questions in German!
  - You may turn in your exercises in German.
  - You may answer exam questions in German.

Organization

- Structure: 3V (lecture) + 1Ü (exercises)
  - 6 EECS credits
  - Part of the area "Applied Computer Science"
- Place & Time
  - Lecture/Exercises: Wed 08:30 – 10:00 room HG Aula
  - Lecture/Exercises: Thu 14:30 – 16:00 room TEMP2
- Exam
  - Written exam
  - 1st Try TBD TBD
  - 2nd Try TBD TBD

Exercises and Supplementary Material

- Exercises
  - Typically 1 exercise sheet every 2 weeks.
  - Pen & paper and programming exercises
    - Python for first exercise slots
    - TensorFlow for Deep Learning part
  - Hands-on experience with the algorithms from the lecture.
  - Send your solutions the night before the exercise class.
  - ~~Need to reach ≥ 50% of the points to qualify for the exam!~~
- Teams are encouraged!
  - You can form teams of up to 4 people for the exercises.
  - Each team should only turn in one solution via L2P.
  - But list the names of all team members in the submission.

Course Webpage

Course Schedule

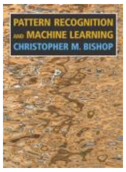
Date	Title	Content	Material
Thu, 2017-10-12	Introduction	Introduction, Probability Theory, Bayes Decision Theory, Minimizing Expected Loss	
Mon, 2017-10-16	Prob. Density Estimation I	Parametric Methods, Gaussian Distribution, Maximum Likelihood	
Thu, 2017-10-19	Prob. Density Estimation II	Bayesian Learning, Nonparametric Methods, Histograms, Kernel Density Estimation	
Mon, 2017-10-23	Prob. Density Estimation III	Mixture of Gaussians, k-Means Clustering, EM-Clustering, EM Algorithm	
Thu, 2017-10-26	Linear Discriminant Functions I	Linear Discriminant Functions, Least-squares Classification, Generalized Linear Models	
Mon, 2017-10-30	Exercise 1	Matlab Tutorial, Probability Density Estimation, GMM, EM	
Thu, 2017-11-02	Linear Discriminant Functions II	Logistic Regression, Iteratively Reweighted Least Squares, Softmax Regression, Error Function Analysis	
Mon, 2017-11-06	Linear SVMs	Linear SVMs, Soft-margin classifiers, nonlinear basis functions	
Thu, 2017-11-09	Non-Linear SVMs	Soft-margin classifiers, nonlinear basis functions, Kernel trick, Mercer's condition, Nonlinear SVMs	

First exercise on 24.10.

http://www.vision.rwth-aachen.de/courses/

Textbooks

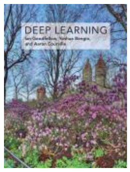
- The first half of the lecture is covered in Bishop's book.
- For Deep Learning, we will use Goodfellow & Bengio.



Christopher M. Bishop
Pattern Recognition and Machine Learning
Springer, 2006

(available in the library's "Handapparat")

I. Goodfellow, Y. Bengio, A. Courville
Deep Learning
MIT Press, 2016



- Research papers will be given out for some topics.
  - Tutorials and deeper introductions.
  - Application papers

How to Find Us

- Office:
  - UMIC Research Centre
  - Mies-van-der-Rohe-Strasse 15, room 124

- Office hours
  - If you have questions about the lecture, contact Paul or Sabarinath.
  - My regular office hours will be announced (additional slots are available upon request)
  - Send us an email before to confirm a time slot.

Questions are welcome!

Machine Learning

- Statistical Machine Learning
  - Principles, methods, and algorithms for learning and prediction on the basis of past evidence
- Already everywhere
  - Speech recognition (e.g. Siri)
  - Machine translation (e.g. Google Translate)
  - Computer vision (e.g. Face detection)
  - Text filtering (e.g. Email spam filters)
  - Operation systems (e.g. Caching)
  - Fraud detection (e.g. Credit cards)
  - Game playing (e.g. Alpha Go)
  - Robotics (everywhere)

What Is Machine Learning Useful For?

Automatic Speech Recognition

What Is Machine Learning Useful For?

Computer Vision
(Object Recognition, Segmentation, Scene Understanding)

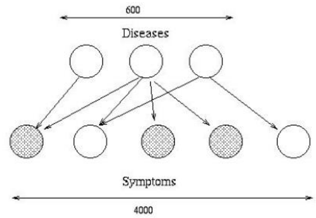
What Is Machine Learning Useful For?

Information Retrieval
(Retrieval, Categorization, Clustering, ...)

What Is Machine Learning Useful For?

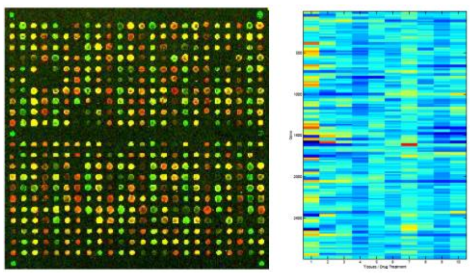
Financial Prediction
(Time series analysis, ...)

What Is Machine Learning Useful For?



Medical Diagnosis
(Inference from partial observations)

What Is Machine Learning Useful For?

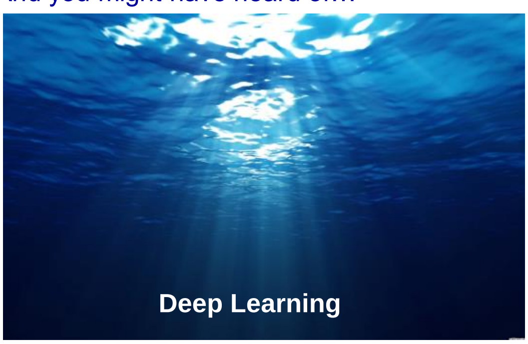


Bioinformatics
(Modelling gene microarray data,...)

What Is Machine Learning Useful For?

Autonomous Driving
(DARPA Grand Challenge,...)

And you might have heard of…



Machine Learning

- Goal
  - ***Machines that learn to perform a task from experience***
- Why?
  - Crucial component of every intelligent/autonomous system
  - Important for a system's adaptability
  - Important for a system's generalization capabilities
  - Attempt to understand human learning

Machine Learning: Core Questions

- ***Learning to perform a task from experience***

- Learning
  - Most important part here!
  - We do not want to encode the knowledge ourselves.
  - The machine should learn the relevant criteria automatically from past observations and adapt to the given situation.

- Tools
  - Statistics
  - Probability theory
  - Decision theory
  - Information theory
  - Optimization theory

Machine Learning: Core Questions

- ***Learning to perform a task from experience***

- Task
  - Can often be expressed through a mathematical function

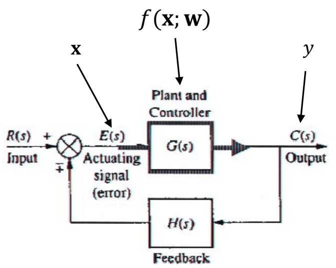
$$y = f(\mathbf{x}; \mathbf{w})$$

  - $\mathbf{x}$: Input
  - y: Output
  - $\mathbf{w}$: Parameters (this is what is "learned")

- Classification vs. Regression
  - Regression: continuous y
  - Classification: discrete y
    - E.g. class membership, sometimes also posterior probability

Example: Regression

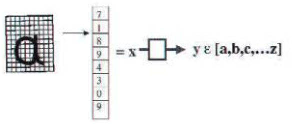
- Automatic control of a vehicle



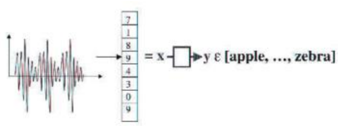
Examples: Classification

- Email filtering $x \in [\text{a-z}]^+ \rightarrow \square \rightarrow y \in [\textbf{important, spam}]$

- Character recognition

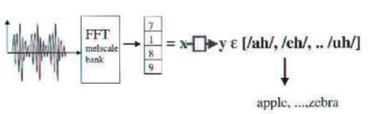


- Speech recognition



Machine Learning: Core Problems

- Input x:



- Features
  - Invariance to irrelevant input variations
  - Selecting the "right" features is crucial
  - Encoding and use of "domain knowledge"
  - Higher-dimensional features are more discriminative.

- Curse of dimensionality
  - Complexity increases exponentially with number of dimensions.

Machine Learning: Core Questions

- ***Learning to perform a task from experience***

- Performance measure: Typically *one number*
  - % correctly classified letters
  - % games won
  - % correctly recognized words, sentences, answers

- Generalization performance
  - Training vs. test
  - "All" data

Machine Learning: Core Questions

- ***Learning to perform a task from experience***

- Performance: “99% correct classification”
  - Of what???
  - Characters? Words? Sentences?
  - Speaker/writer independent?
  - Over what data set?
  - …

- “The car drives without human intervention 99% of the time on country roads”

Machine Learning: Core Questions

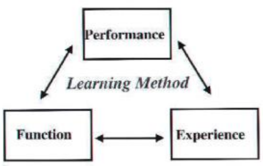
- ***Learning to perform a task from experience***

- What data is available?
  - Data with labels: *supervised learning*
    - Images / speech with target labels
    - Car sensor data with target steering signal
  - Data without labels: *unsupervised learning*
    - Automatic clustering of sounds and phonemes
    - Automatic clustering of web sites
  - Some data with, some without labels: *semi-supervised learning*
  - Feedback/rewards: *reinforcement learning*

Machine Learning: Core Questions

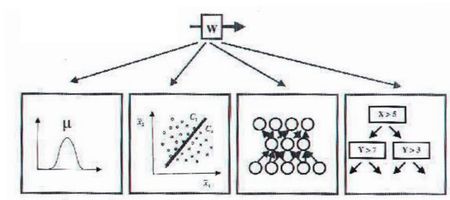
- ***Learning to perform a task from experience***

- Learning
  - Most often learning = optimization
  - Search in hypothesis space
  - Search for the “best” function / model parameter $\mathbf{w}$
    - I.e. maximize $y = f(\mathbf{x}; \mathbf{w})$ w.r.t. the performance measure



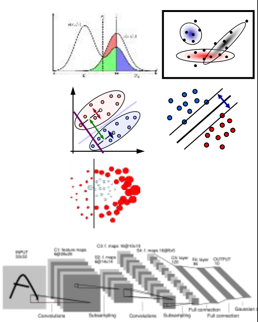
Machine Learning: Core Questions

- Learning is optimization of $y = f(\mathbf{x}; \mathbf{w})$
  - $\mathbf{w}$: characterizes the family of functions
  - $\mathbf{w}$: indexes the space of hypotheses
  - $\mathbf{w}$: vector, connection matrix, graph, …



Course Outline

- Fundamentals
  - Bayes Decision Theory
  - Probability Density Estimation
- Classification Approaches
  - Linear Discriminants
  - Support Vector Machines
  - Ensemble Methods & Boosting
- Deep Learning
  - Foundations
  - Convolutional Neural Networks
  - Recurrent Neural Networks



Topics of This Lecture

- Review: Probability Theory
  - Probabilities
  - Probability densities
  - Expectations and covariances
- Bayes Decision Theory
  - Basic concepts
  - Minimizing the misclassification rate
  - Minimizing the expected loss
  - Discriminant functions

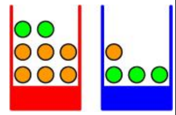
Probability Theory

"Probability theory is nothing but common sense reduced to calculation."

Pierre-Simon de Laplace, 1749-1827

Probability Theory

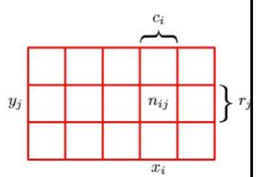
- Example: apples and oranges
  - We have two boxes to pick from.
  - Each box contains both types of fruit.
  - What is the probability of picking an apple?



- Formalization
  - Let $B \in \{r, b\}$ be a random variable for the box we pick.
  - Let $F \in \{a, o\}$ be a random variable for the type of fruit we get.
  - Suppose we pick the red box 40% of the time. We write this as
    $$p(B = r) = 0.4 \qquad p(B = b) = 0.6$$
  - The probability of picking an apple *given* a choice for the box is
    $$p(F = a \mid B = r) = 0.25 \qquad p(F = a \mid B = b) = 0.75$$
  - What is the probability of picking an apple?
    $$p(F = a) = ?$$

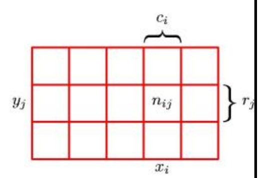
Probability Theory

- More general case
  - Consider two random variables $X \in \{x_i\}$ and $Y \in \{y_j\}$
  - Consider N trials and let
    $$n_{ij} = \#\{X = x_i \wedge Y = y_j\}$$
    $$c_i = \#\{X = x_i\}$$
    $$r_j = \#\{Y = y_j\}$$



- Then we can derive
  - Joint probability $$p(X = x_i, Y = y_j) = \frac{n_{ij}}{N}$$
  - Marginal probability $$p(X = x_i) = \frac{c_i}{N}.$$
  - Conditional probability $$p(Y = y_j \mid X = x_i) = \frac{n_{ij}}{c_i}$$

Probability Theory



- Rules of probability
  - Sum rule
    $$p(X = x_i) = \frac{c_i}{N} = \frac{1}{N}\sum_{j=1}^{L} n_{ij} = \sum_{j=1}^{L} p(X = x_i, Y = y_j)$$
  - Product rule
    $$p(X = x_i, Y = y_j) = \frac{n_{ij}}{N} = \frac{n_{ij}}{c_i} \cdot \frac{c_i}{N}$$
    $$= p(Y = y_j \mid X = x_i)\, p(X = x_i)$$

The Rules of Probability

- Thus we have

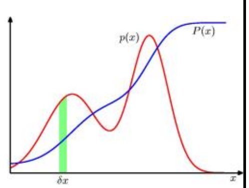
Sum Rule	$p(X) = \sum_Y p(X, Y)$
Product Rule	$p(X, Y) = p(Y \mid X)\, p(X)$

- From those, we can derive

Bayes' Theorem	$p(Y \mid X) = \frac{p(X \mid Y)\, p(Y)}{p(X)}$
where	$p(X) = \sum_Y p(X \mid Y)\, p(Y)$

Probability Densities

- Probabilities over continuous variables are defined over their probability density function (pdf) $p(x)$
  $$p(x \in (a, b)) = \int_a^b p(x)\, \mathrm{d}x$$



- The probability that x lies in the interval $(-\infty, z)$ is given by the cumulative distribution function
  $$P(z) = \int_{-\infty}^{z} p(x)\, \mathrm{d}x$$

Expectations

- The average value of some function $f(x)$ under a probability distribution $p(x)$ is called its expectation

$$\mathbb{E}[f] = \sum_x p(x) f(x) \qquad \mathbb{E}[f] = \int p(x) f(x)\,\mathrm{d}x$$

discrete case — continuous case

- If we have a finite number N of samples drawn from a pdf, then the expectation can be approximated by

$$\mathbb{E}[f] \simeq \frac{1}{N}\sum_{n=1}^{N} f(x_n)$$

- We can also consider a conditional expectation

$$\mathbb{E}_x[f|y] = \sum_x p(x|y) f(x)$$

Variances and Covariances

- The variance provides a measure how much variability there is in $f(x)$ around its mean value $\mathbb{E}[f(x)]$.

$$\mathrm{var}[f] = \mathbb{E}\left[(f(x) - \mathbb{E}[f(x)])^2\right] = \mathbb{E}[f(x)^2] - \mathbb{E}[f(x)]^2$$

- For two random variables x and y, the covariance is defined by

$$\begin{aligned}\mathrm{cov}[x,y] &= \mathbb{E}_{x,y}\left[\{x - \mathbb{E}[x]\}\{y - \mathbb{E}[y]\}\right] \\ &= \mathbb{E}_{x,y}[xy] - \mathbb{E}[x]\mathbb{E}[y]\end{aligned}$$

- If $\mathbf{x}$ and $\mathbf{y}$ are vectors, the result is a covariance matrix

$$\begin{aligned}\mathrm{cov}[\mathbf{x},\mathbf{y}] &= \mathbb{E}_{\mathbf{x},\mathbf{y}}\left[\{\mathbf{x} - \mathbb{E}[\mathbf{x}]\}\{\mathbf{y}^{\mathrm{T}} - \mathbb{E}[\mathbf{y}^{\mathrm{T}}]\}\right] \\ &= \mathbb{E}_{\mathbf{x},\mathbf{y}}[\mathbf{x}\mathbf{y}^{\mathrm{T}}] - \mathbb{E}[\mathbf{x}]\mathbb{E}[\mathbf{y}^{\mathrm{T}}]\end{aligned}$$

Bayes Decision Theory

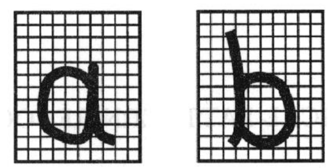
Thomas Bayes, 1701-1761

"The theory of inverse probability is founded upon an error, and must be wholly rejected."

R.A. Fisher, 1925

Bayes Decision Theory

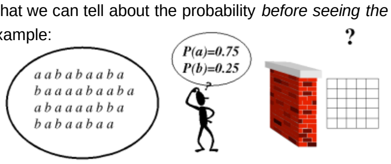
- Example: handwritten character recognition



- Goal:
  - Classify a new letter such that the probability of misclassification is minimized.

Bayes Decision Theory

- Concept 1: Priors (a priori probabilities) $\boxed{p(C_k)}$
  - What we can tell about the probability *before seeing the data*.
  - Example:

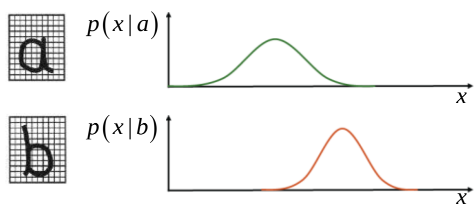


$$C_1 = a \qquad p(C_1) = 0.75$$
$$C_2 = b \qquad p(C_2) = 0.25$$

- In general: $\sum_k p(C_k) = 1$

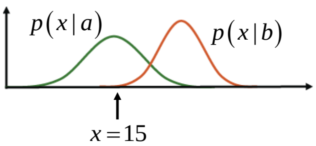
Bayes Decision Theory

- Concept 2: Conditional probabilities $\boxed{p(x|C_k)}$
  - Let x be a feature vector.
  - x measures/describes certain properties of the input.
    - E.g. number of black pixels, aspect ratio, …
  - $p(x|C_k)$ describes its likelihood for class C_k.



Bayes Decision Theory

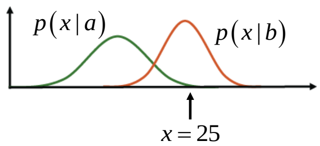
- Example:



- Question:
  - Which class?
  - Since $p(x|b)$ is much smaller than $p(x|a)$ the decision should be 'a' here.

Bayes Decision Theory

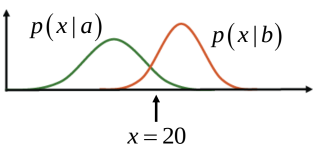
- Example:



- Question:
  - Which class?
  - Since $p(x|a)$ is much smaller than $p(x|b)$, the decision should be 'b' here.

Bayes Decision Theory

- Example:



- Question:
  - Which class?
  - Remember that $p(a) = 0.75$ and $p(b) = 0.25$…
  - I.e., the decision should be again 'a'.
  - $\Rightarrow$ How can we formalize this?

Bayes Decision Theory

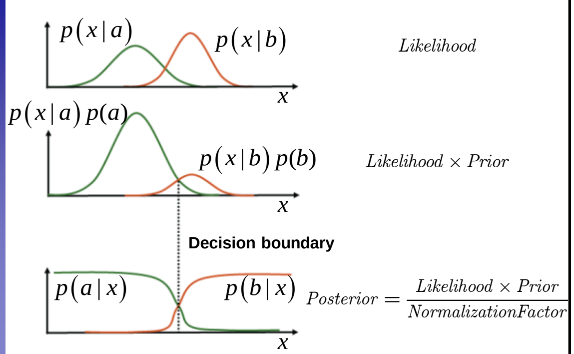
- Concept 3: Posterior probabilities $\boxed{p(C_k|x)}$
  - We are typically interested in the *a posteriori* probability, i.e. the probability of class C_k given the measurement vector x.
- Bayes' Theorem:

$$p(C_k|x) = \frac{p(x|C_k)\,p(C_k)}{p(x)} = \frac{p(x|C_k)\,p(C_k)}{\sum_i p(x|C_i)\,p(C_i)}$$

- Interpretation

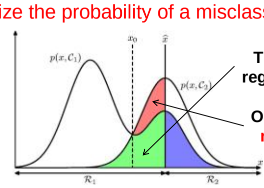
$$Posterior = \frac{Likelihood \times Prior}{Normalization\ Factor}$$

Bayes Decision Theory



Bayesian Decision Theory

- Goal: Minimize the probability of a misclassification



The green and blue regions stay constant.

Only the size of the red region varies!

$$\begin{aligned} p(\text{mistake}) &= p(\mathbf{x} \in \mathcal{R}_1, \mathcal{C}_2) + p(\mathbf{x} \in \mathcal{R}_2, \mathcal{C}_1) \\ &= \int_{\mathcal{R}_1} p(\mathbf{x}, \mathcal{C}_2)\,\mathrm{d}\mathbf{x} + \int_{\mathcal{R}_2} p(\mathbf{x}, \mathcal{C}_1)\,\mathrm{d}\mathbf{x}. \\ &= \int_{\mathcal{R}_1} p(\mathcal{C}_2|\mathbf{x})p(\mathbf{x})d\mathbf{x} + \int_{\mathcal{R}_2} p(\mathcal{C}_1|\mathbf{x})p(\mathbf{x})d\mathbf{x} \end{aligned}$$

Bayes Decision Theory

- Optimal decision rule
  - Decide for C_1 if
$$p(\mathcal{C}_1|x) > p(\mathcal{C}_2|x)$$
  - This is equivalent to
$$p(x|\mathcal{C}_1)p(\mathcal{C}_1) > p(x|\mathcal{C}_2)p(\mathcal{C}_2)$$
  - Which is again equivalent to (Likelihood-Ratio test)
$$\frac{p(x|\mathcal{C}_1)}{p(x|\mathcal{C}_2)} > \underbrace{\frac{p(\mathcal{C}_2)}{p(\mathcal{C}_1)}}_{\text{Decision threshold } \theta}$$

Slide credit: Bernt Schiele

Generalization to More Than 2 Classes

- Decide for class k whenever it has the greatest posterior probability of all classes:
$$p(\mathcal{C}_k|x) > p(\mathcal{C}_j|x) \quad \forall j \neq k$$
$$p(x|\mathcal{C}_k)p(\mathcal{C}_k) > p(x|\mathcal{C}_j)p(\mathcal{C}_j) \quad \forall j \neq k$$
- Likelihood-ratio test
$$\frac{p(x|\mathcal{C}_k)}{p(x|\mathcal{C}_j)} > \frac{p(\mathcal{C}_j)}{p(\mathcal{C}_k)} \quad \forall j \neq k$$

Slide credit: Bernt Schiele

Classifying with Loss Functions

- Generalization to decisions with a loss function
  - Differentiate between the possible decisions and the possible true classes.
  - Example: medical diagnosis
    - Decisions: *sick* or *healthy* (or: *further examination necessary*)
    - Classes: patient is *sick* or *healthy*
  - The cost may be asymmetric:
$$loss(decision = healthy|patient = sick) >>$$
$$loss(decision = sick|patient = healthy)$$

Slide credit: Bernt Schiele

Classifying with Loss Functions

- In general, we can formalize this by introducing a loss matrix L_{kj}
$$L_{kj} = loss\ for\ decision\ \mathcal{C}_j\ if\ truth\ is\ \mathcal{C}_k.$$
- Example: cancer diagnosis

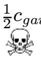
$$L_{cancer\ diagnosis} = \begin{array}{c|cc} & \multicolumn{2}{c}{\textbf{Decision}} \\ & \text{cancer} & \text{normal} \\ \hline \textbf{Truth}\ \text{cancer} & 0 & 1000 \\ \text{normal} & 1 & 0 \end{array}$$

Classifying with Loss Functions

- Loss functions may be different for different actors.
  - Example:

$$\begin{array}{cc} \text{“invest”} & \text{“don't invest”} \end{array}$$
$$L_{stocktrader}(subprime) = \begin{pmatrix} -\frac{1}{2}c_{gain} & 0 \\ 0 & 0 \end{pmatrix}$$

$$L_{bank}(subprime) = \begin{pmatrix} -\frac{1}{2}c_{gain} & 0 \\ & 0 \end{pmatrix}$$



⇒ Different loss functions may lead to different Bayes optimal strategies.

Minimizing the Expected Loss

- Optimal solution is the one that minimizes the loss.
  - But: loss function depends on the true class, which is unknown.
- Solution: Minimize the expected loss
$$\mathbb{E}[L] = \sum_k \sum_j \int_{\mathcal{R}_j} L_{kj} p(\mathbf{x}, \mathcal{C}_k)\,\mathrm{d}\mathbf{x}$$
- This can be done by choosing the regions $\mathcal{R}_j$ such that
$$\mathbb{E}[L] = \sum_k L_{kj} p(\mathcal{C}_k|\mathbf{x})$$
which is easy to do once we know the posterior class probabilities $p(\mathcal{C}_k|\mathbf{x})$

Minimizing the Expected Loss

- Example:
  - 2 Classes: C_1, C_2
  - 2 Decision: α_1, α_2
  - Loss function: $L(\alpha_j|\mathcal{C}_k) = L_{kj}$
  - Expected loss (= risk R) for the two decisions:

$$\begin{aligned}\mathbb{E}_{\alpha_1}[L] = R(\alpha_1|\mathbf{x}) &= L_{11}p(\mathcal{C}_1|\mathbf{x}) + L_{21}p(\mathcal{C}_2|\mathbf{x}) \\ \mathbb{E}_{\alpha_2}[L] = R(\alpha_2|\mathbf{x}) &= L_{12}p(\mathcal{C}_1|\mathbf{x}) + L_{22}p(\mathcal{C}_2|\mathbf{x})\end{aligned}$$

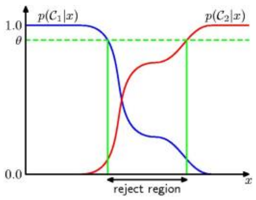
- Goal: Decide such that expected loss is minimized
  - I.e. decide α_1 if $R(\alpha_2|\mathbf{x}) > R(\alpha_1|\mathbf{x})$

Minimizing the Expected Loss

$$\begin{aligned}R(\alpha_2|\mathbf{x}) &> R(\alpha_1|\mathbf{x}) \\ L_{12}p(\mathcal{C}_1|\mathbf{x}) + L_{22}p(\mathcal{C}_2|\mathbf{x}) &> L_{11}p(\mathcal{C}_1|\mathbf{x}) + L_{21}p(\mathcal{C}_2|\mathbf{x}) \\ (L_{12} - L_{11})p(\mathcal{C}_1|\mathbf{x}) &> (L_{21} - L_{22})p(\mathcal{C}_2|\mathbf{x}) \\ \frac{(L_{12} - L_{11})}{(L_{21} - L_{22})} &> \frac{p(\mathcal{C}_2|\mathbf{x})}{p(\mathcal{C}_1|\mathbf{x})} = \frac{p(\mathbf{x}|\mathcal{C}_2)p(\mathcal{C}_2)}{p(\mathbf{x}|\mathcal{C}_1)p(\mathcal{C}_1)} \\ \frac{p(\mathbf{x}|\mathcal{C}_1)}{p(\mathbf{x}|\mathcal{C}_2)} &> \frac{(L_{21} - L_{22})}{(L_{12} - L_{11})}\frac{p(\mathcal{C}_2)}{p(\mathcal{C}_1)}\end{aligned}$$

$\Rightarrow$ Adapted decision rule taking into account the loss.

The Reject Option



- Classification errors arise from regions where the largest posterior probability $p(\mathcal{C}_k|\mathbf{x})$ is significantly less than 1.
  - These are the regions where we are relatively uncertain about class membership.
  - For some applications, it may be better to reject the automatic decision entirely in such a case and e.g. consult a human expert.

Image source: C.M. Bishop, 2006

Discriminant Functions

- Formulate classification in terms of comparisons
  - Discriminant functions

$$y_1(x), \ldots, y_K(x)$$

  - Classify x as class C_k if

$$y_k(x) > y_j(x) \quad \forall j \neq k$$

- Examples (Bayes Decision Theory)

$$\begin{aligned}y_k(x) &= p(\mathcal{C}_k|x) \\ y_k(x) &= p(x|\mathcal{C}_k)p(\mathcal{C}_k) \\ y_k(x) &= \log p(x|\mathcal{C}_k) + \log p(\mathcal{C}_k)\end{aligned}$$

Different Views on the Decision Problem

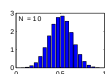
- $y_k(x) \propto p(x|\mathcal{C}_k)p(\mathcal{C}_k)$
  - First determine the class-conditional densities for each class individually and separately infer the prior class probabilities.
  - Then use Bayes' theorem to determine class membership.

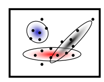
  $\Rightarrow$ *Generative methods*
- $y_k(x) = p(\mathcal{C}_k|x)$
  - First solve the inference problem of determining the posterior class probabilities.
  - Then use decision theory to assign each new x to its class.

  $\Rightarrow$ *Discriminative methods*
- Alternative
  - Directly find a discriminant function $y_k(x)$ which maps each input x directly onto a class label.

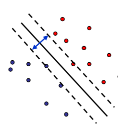
Next Lectures…

- Ways how to estimate the probability densities $p(x|\mathcal{C}_k)$
  - Non-parametric methods
    - Histograms
    - k-Nearest Neighbor
    - Kernel Density Estimation
  - Parametric methods
    - Gaussian distribution
    - Mixtures of Gaussians





- Discriminant functions
  - Linear discriminants
  - Support vector machines



$\Rightarrow$ *Next lectures…*

References and Further Reading

- More information, including a short review of Probability theory and a good introduction in Bayes Decision Theory can be found in Chapters 1.1, 1.2 and 1.5 of

Christopher M. Bishop
Pattern Recognition and Machine Learning
Springer, 2006