Deep Learning

*Lecture 1 – Introduction, linear models*

Niklas Wahlström
Division of Systems and Control
Department of Information Technology
Uppsala University

niklas.wahlstrom@it.uu.se
www.it.uu.se/katalog/nikwa778
What is the course about?
Machine learning

"Machine learning is about learning, reasoning and acting based on data."

“It is one of today’s most rapidly growing technical fields, lying at the intersection of computer science and statistics, and at the core of artificial intelligence and data science.”


Deep learning

“Deep learning allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction.”


Example: Image classification

Input: pixels of an image
Output: object identity
Each hidden layer extracts increasingly abstract features.

Zeiler, M. D. and Fergus, R. Visualizing and understanding convolutional networks
Computer Vision - ECCV (2014).
ex) Cancer diagnosis

Systems for detecting cell divisions (mitosis) in histology images can be used to improve (or automate) cancer diagnosis.

- Learn a model with
  - input: RBG histology image (pixel values)
  - output: number and locations (in the image) of mitosis detections
- Training data: Histology images labeled by experts.

**Task:** Colorize gray-scale photos.

- Learn a model with
  - input: gray-scale pixel values.
  - output: color pixel values.

ex) Colorization (II/II)

Model applied to legacy grayscale photos.
ex) Autonomous driving

- Recognizing objects in a street view
- Learn a model with
  input: the RGB values of the image
  output: the location of pedestrians in the image
ex) Higgs Machine Learning Challenge

**HiggsML:** Crowd-sourcing initiative by CERN (hosted at Kaggle)

- Separate $H \rightarrow \tau\tau$ from background noise.
- Lean a model with
  - input: 30-dimensional vector of “features” recorded during the experiment.
  - output: “signal” or “background”
- Deep Learning methods among the winning methods.

Course information
# Lecture outline

1. Introduction
2. Feed forward neural networks
3. Optimization
4. Convolutional neural networks I
5. Convolutional neural networks II
6. Bias-variance trade-off, cross validation
7. Regularization
8. Variational inference
9. Variational autoencoders
10. Summary and guest lecture
Hand-in assignments

4 hand-in assignments (HAs):

- Covers (mainly) implementation aspects of deep learning.
- Deadlines for all HAs available on course homepage.
- Everyone has 7 joker days, which can be used with any HA.
- You are encouraged to collaborate...
- ... but you should write submit your own report and code.

Optional helpdesks are scheduled after each lecture.
Optional project

- After the course you are encouraged to carry out a project
- Preferably 1-4 students in each team
- The projects must include real data
- Awarded 3hp extra
- Conducted during summer, see course homepage for the dates
Course literature


  We will not follow the book strictly though. Another great resource is

- Michael A. Nielson *Neural Networks and Deep Learning* Determiniation Press, 2015. neuralnetworksanddeeplearning.com

  First lectures are well covered by lecture notes linked from the course homepage.

  All of them available online.
Teachers

Teachers involved in the course (in approximate order of appearance):

Niklas Wahlström
Room: 2319

Thomas Schön
Room: 2209

Joakim Lindblad
Room: 2141

Jalil Taghia
Room: 2321

Fredrik Gustafsson
Room: 2303

Nicolas Pielawski
Room: xxxx

Anindya Gupta
Room: xxxx

Lecturers

Teaching assistants

All room numbers are at ITC Polacksbacken.
You can reach us by email: <firstname.lastname>@it.uu.se.
Course applicants
Linear regression and classification
Supervised machine learning

Methods for automatically learning (training, estimating, ... ) a model for the relationship between

- the input $x$, and the
- the output $y$

from observed training data

$$\mathcal{T} \overset{\text{def}}{=} \{(y_1, x_1), (y_2, x_2), \ldots, (y_n, x_n)\}.$$
Regression vs. classification

We will distinguish between two types of problems: **regression** and **classification**

**Regression** is when the output $y$ is quantitative, e.g.
- Climate models ($y =$ “increase in global temperature”)
- Economic models ($y =$ “change in GDP”)

**Classification** is when the output $y$ is qualitative, e.g.
- Spam filters ($y \in \{\text{spam, good email}\}$)
- Diagnosis systems ($y \in \{\text{ALL, AML, CLL, CML, no leukemia}\}$)
- Fingerprint verification ($y \in \{\text{match, no match}\}$)
ex) A regression problem

Consider a linear model to explain the data

\[ \hat{y} = wx + b \]
What is a good model?

![Graph showing different linear models and training data](image)

- A linear model
- Another linear model
- A third linear model
- A fourth linear model
- Training data $\mathcal{T}$

**Introduction**
Learning using maximum likelihood

Assume that each data point can be described by a linear model + noise

\[ y_i = wx_i + b + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0, \sigma^2_{\varepsilon}) \]

**Maximum likelihood:** Think of \( \varepsilon \) (dotted) as Gaussian random variables, and **choose the model** (solid) such that the resulting \( \varepsilon \) are as likely as possible.
Gaussian (Normal) distribution

Probability density function (PDF) for the scalar Gaussian distribution

\[ N(z \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(z-\mu)^2}{2\sigma^2}} \]

- \( \mu \) is the mean (expected value of the distribution)
- \( \sigma \) is the standard deviation
- \( \sigma^2 \) is the variance

\( z \sim N(\mu, \sigma^2) \) means that \( z \) is a Gaussian random variable with mean \( \mu \) and variance \( \sigma^2 \). \( \sim \) reads “distributed according to”.

\[ \mu = -1, \sigma^2 = 0.5 \]
\[ \mu = 7, \sigma^2 = 0.2 \]
\[ \mu = 4, \sigma^2 = 4 \]
Maximum likelihood

A linear model with Gaussian noise

\[ y_i = wx_i + b + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0, \sigma^2), \quad i = 1, \ldots, n \]

The model can also be expressed as

\[ p(y_i | x_i, w, b) = \mathcal{N}(y_i | wx_i + b, \sigma^2) \]

Pick \( w \) and \( b \) which makes the data as likely as possible

\[ \hat{w}, \hat{b} = \operatorname*{argmax}_{w,b} p(y_1, \ldots, y_n | x_1, \ldots, x_n, w, b) \]

Assume all \( \varepsilon_i \) to be independent

\[ p(y_1, \ldots, y_n | x_1, \ldots, x_n, w, b) = \prod_{i=1}^{n} p(y_i | x_i, w, b) \]

\[ = \prod_{i=1}^{n} \mathcal{N}(y_i | wx_i + b, \sigma^2) \]

\[ = e^{-\frac{1}{2\sigma^2} \sum_{i=1}^{n} (y_i - wx_i - b)^2} \]
Linear regression

The least squares problem

\[ \hat{w}, \hat{b} = \arg\min_{w,b} \sum_{i=1}^{n} (wx_i + b - y_i)^2 \]
ex) A classification problem

**Beaver Body Temperature Data**

**Input** Body temperature \( x \)

**Output** Beaver is outside \( y = 1 \), or inside \( y = 0 \), of the retreat.

![Graph showing linear regression model and data points.](image)

Linear regression is not suitable since it is not constrained to \([0, 1]\).
Logistic regression

Consider a binary classification problem \( y \in \{0, 1\} \).

\[ p_i = \Pr(y_i = 1| x_i) \quad \text{and thus} \quad \Pr(y_i = 0| x_i) = 1 - p_i \]

Let the **odds** is the ration between the two class probabilities

\[ \frac{\Pr(y_i = 1| x_i)}{\Pr(y_i = 0| x_i)} = \frac{p_i}{1 - p_i} \in (0, \infty) \]

and **log odds** consequently

\[ \log \frac{p_i}{1 - p_i} \in (-\infty, \infty) \]

**Logistic regression** Assume a linear model for the log odds

\[ \log \frac{p_i}{1 - p_i} = wx_i + b \quad \Rightarrow \quad p_i = \frac{e^{wx_i+b}}{1 + e^{wx_i+b}} \]
Logistic function (aka sigmoid function)

The function $f : \mathbb{R} \mapsto [0, 1]$ defined as $f(z) = \frac{e^z}{1 + e^z}$ is known as the **logistic function**.
Logistic regression using maximum likelihood

Pick $w$ and $b$ which make data as likely as possible

$$\hat{w}, \hat{b} = \underset{w,b}{\operatorname{argmax}} \Pr(y_1, \ldots, y_n|x_1, \ldots, x_n, w, b)$$

Assume all $y_i$ to be independent

$$\ln l(w, b) = \ln \Pr(y_1, \ldots, y_n|x_1, \ldots, x_n, w, b)$$

$$= \sum_{i=1}^{n} \ln \Pr(y_i|x_i, w, b)$$

$$= \sum_{\begin{array}{c} i=1 \\ y_i=1 \end{array}}^{n} \ln \Pr(y_i = 1|x_i, w, b) + \sum_{\begin{array}{c} i=1 \\ y_i=0 \end{array}}^{n} \ln \Pr(y_i = 0|x_i, w, b)$$

$$= \sum_{\begin{array}{c} i=1 \\ y_i=1 \end{array}}^{n} \ln \Pr(y_i = 1|x_i, w, b) + \sum_{\begin{array}{c} i=1 \\ y_i=0 \end{array}}^{n} \ln \Pr(y_i = 0|x_i, w, b)$$

This leads to the following optimization problem

$$\hat{w}, \hat{b} = \underset{w,b}{\operatorname{argmin}} - \sum_{i=1}^{n} y_i \ln(p_i) + (1 - y_i) \ln(1 - p_i)$$
The Beaver data example

Logistic regression model

Data \((x_i, y_i)\)

Temperature \(x\)

\[ \Pr(y = 1 \mid x) \]
Linear and logistic regression

Multidimensional input

Linear and logistic regression also work multidimensional inputs

\[ x_i = [x_{i1}, \ldots, x_{ip}]^T, \quad i = 1, \ldots, n \]

We assign one parameter for each input dimension

\[ w = [w_1, \ldots, w_p]^T \]

**Linear regression**

\[ \hat{y}_i = w^T x_i + b \]

**Logistic regression**

\[ \Pr(y_i = 1|x_i) = \frac{e^{w^T x_i + b}}{1 + e^{w^T x_i + b}} \]
# Linear and logistic regression

## Linear regression

**Output**

\[ y_i \in \mathbb{R} \]

**Model**

\[ \hat{y}_i = w^T x_i + b \]

**Loss**

\[ L_i = (y_i - \hat{y}_i)^2 \]

## Logistic regression

**Output**

\[ y_i \in \{0, 1\} \]

**Model**

\[ p_i = \Pr(y_i = 1|x_i) = \frac{e^{w^T x_i + b}}{1 + e^{w^T x_i + b}} \]

**Loss**

\[ L_i = -y_i \ln(p_i) - (1-y_i) \ln(1-p_i) \]

We find \( w \) and \( b \) by minimizing sum of the losses

\[ \hat{w}, \hat{b} = \arg\min_{w, b} \frac{1}{n} \sum_{i=1}^{n} L_i \]

- For linear regression the problem can be solved analytically.
- For logistic regression we need to use numerical optimization (will talk about it in the next lecture).
Hand-in assignment 1A
Classifying biopsies using logistic regression

Biopst data from $n = 699$ breast tumors

<table>
<thead>
<tr>
<th>Input</th>
<th>$x_{i1}$</th>
<th>clump thickness</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$x_{i2}$</td>
<td>uniformity of cell size</td>
</tr>
<tr>
<td></td>
<td>$x_{i3}$</td>
<td>uniformity of cell shape</td>
</tr>
<tr>
<td></td>
<td>$\vdots$</td>
<td>$\vdots$</td>
</tr>
<tr>
<td></td>
<td>$x_{i9}$</td>
<td>mitoses</td>
</tr>
</tbody>
</table>

Output | $y_i$ | benign/malignant |

Tasks:
- Derive the gradients of the cost function w.r.t. the parameters
- Implement the **logistic regression** model
- Update the parameters with **gradient descent**

The code will be extended to a neural network in HA 1B.
A few concepts to summarize lecture 1

**Machine Learning:** Deals with learning, reasoning and acting based on data.

**Regression:** Learning problem where the *output* is quantitative.

**Classification:** Learning problem where the *output* is qualitative.

**Maximum likelihood:** Learning objective based on probability theory.

**Linear Regression:** Linear model for regression problems.

**Logistic Regression:** Linear model for classification problems.