Machine Learning applied to neuroscience and cognition
CA6b-Planning
Fridays 9h30: 12h30

[1] 13/02/2015  Bayesian approaches applied to cognition (Sophie Deneve)
[2] 20/02/2015  Bayesian approaches applied to neural processing (Sophie Deneve)
[3] 27/02/2015  TD on population coding (Charlotte Le Mouel)
[4] 06/03/2015  Normative approaches applied to perception (Angela Yu)
[5] 13/03/2015  TD on Bayesian methods (Izzet Burak Yildiz)
[6] 20/03/2015  Supervised and unsupervised learning (Sophie Deneve) Projects
[7] 27/03/2015  TD on PCA applied to neural data (Matty Chalk)
[8] 03/04/2015  Efficient coding in sensory systems (Sophie Deneve)
[9] 17/04/2015  TD on Sparse coding (David Chulz)
[10] 24/04/2015 Introduction to graphical models (Sophie Deneve)
[11] 15/05/2015  TD on belief propagation (Pantelis Leptougos)
[12] 22/05/2015  Coding in balanced spiking networks (Sophie Deneve)
[13] 29/05/2015  TD on balanced spiking networks (Erwan Ledoux)
[14] 05/06/2015  Project presentation
Machine learning and artificial intelligence

“Classic” AI: Turing test

Verbal communication = ultimate intelligence criteria?
Machine learning and artificial intelligence

What is hard about playing chess?
Curse of Dimensionality
Curse of dimensionality

Chess

Image processing

$M^7$

$M^{64}$

$M^{n\times n}!!!
Solution: find structures in data
Shared challenge in ML and Cognition

Too much data...
Shared challenge in ML and Cognition

Too few data...
What is Machine Learning?

• “The goal of machine learning is to build computer systems that can adapt and learn from their experience.”
  – Tom Dietterich
A Generic System

\[ x_1 \quad \rightarrow \quad y_1 \]
\[ x_2 \quad \rightarrow \quad y_2 \]
\[ \vdots \]
\[ x_N \quad \rightarrow \quad y_L \]

Input Variables: \( x = (x_1, x_2, \ldots, x_N) \)

Hidden Variables: \( h = (h_1, h_2, \ldots, h_K) \)

Output Variables: \( y = (y_1, y_2, \ldots, y_L) \)

Parameters: \( w = (w_1, w_2, \ldots, w_M) \)

Training examples: \( (x^1, t^1), (x^2, t^2), \ldots, (x^D, t^D) \)
Example

Handwritten Digit Recognition

\[ y = f_w (x, h) \]

x: pixelized or pre-processed image.

y: digit class.

h: left/right handed

W: parameters
Different types of learning

• **Supervised learning:**
  1. Classification (discrete $y$),
  2. Regression (continuous $y$).

• **Unsupervised learning** (**no** $t$).
  1. Clustering ($h = \text{different groups of types of data}$).
  2. Density estimation ($h = \text{parameters of probability dist.}$)
  3. Reduction ($h = \text{a few latent variable describing high dimensional data}$).

• **Reinforcement learning** ($y = \text{actions, rewards}$).
Polynomial Curve Fitting

\[ y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M = \sum_{j=0}^{M} w_j x^j \]
Sum-of-Squares Error Function

\[ E(w) = \frac{1}{2} \sum_{n=1}^{N} \{y(x_n, w) - t_n\}^2 \]
$0^{th}$ Order Polynomial

$M = 0$
1\textsuperscript{st} Order Polynomial

\[ M = 1 \]
3\textsuperscript{rd} Order Polynomial

\[ M = 3 \]
$9^{\text{th}}$ Order Polynomial

$M = 9$
Over-fitting

Root-Mean-Square (RMS) Error: $E_{RMS} = \sqrt{\frac{2E(w^*)}{N}}$
## Polynomial Coefficients

<table>
<thead>
<tr>
<th></th>
<th>$M = 0$</th>
<th>$M = 1$</th>
<th>$M = 3$</th>
<th>$M = 9$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$w_0^*$</td>
<td>0.19</td>
<td>0.82</td>
<td>0.31</td>
<td>0.35</td>
</tr>
<tr>
<td>$w_1^*$</td>
<td></td>
<td>-1.27</td>
<td>7.99</td>
<td>232.37</td>
</tr>
<tr>
<td>$w_2^*$</td>
<td></td>
<td></td>
<td>-25.43</td>
<td>-5321.83</td>
</tr>
<tr>
<td>$w_3^*$</td>
<td></td>
<td></td>
<td>17.37</td>
<td>48568.31</td>
</tr>
<tr>
<td>$w_4^*$</td>
<td></td>
<td></td>
<td></td>
<td>-231639.30</td>
</tr>
<tr>
<td>$w_5^*$</td>
<td></td>
<td></td>
<td></td>
<td>640042.26</td>
</tr>
<tr>
<td>$w_6^*$</td>
<td></td>
<td></td>
<td></td>
<td>-1061800.52</td>
</tr>
<tr>
<td>$w_7^*$</td>
<td></td>
<td></td>
<td></td>
<td>1042400.18</td>
</tr>
<tr>
<td>$w_8^*$</td>
<td></td>
<td></td>
<td></td>
<td>-557682.99</td>
</tr>
<tr>
<td>$w_9^*$</td>
<td></td>
<td></td>
<td></td>
<td>125201.43</td>
</tr>
</tbody>
</table>
Data Set Size:

$9^{th}$ Order Polynomial
Data Set Size:

9th Order Polynomial

\[ N = 100 \]
Regularization

- Penalize large coefficient values

\[ \tilde{E}(w) = \frac{1}{2} \sum_{n=1}^{N} \{y(x_n, w) - t_n\}^2 + \frac{\lambda}{2} \|w\|^2 \]
Regularization:

\[ \ln \lambda = -18 \]
Regularization:
Regularization: $\ln \lambda$ vs. $E_{\text{RMS}}$
## Polynomial Coefficients

<table>
<thead>
<tr>
<th></th>
<th>$\ln \lambda = -\infty$</th>
<th>$\ln \lambda = -18$</th>
<th>$\ln \lambda = 0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$w_0^*$</td>
<td>0.35</td>
<td>0.35</td>
<td>0.13</td>
</tr>
<tr>
<td>$w_1^*$</td>
<td>232.37</td>
<td>4.74</td>
<td>-0.05</td>
</tr>
<tr>
<td>$w_2^*$</td>
<td>-5321.83</td>
<td>-0.77</td>
<td>-0.06</td>
</tr>
<tr>
<td>$w_3^*$</td>
<td>48568.31</td>
<td>-31.97</td>
<td>-0.05</td>
</tr>
<tr>
<td>$w_4^*$</td>
<td>-231639.30</td>
<td>-3.89</td>
<td>-0.03</td>
</tr>
<tr>
<td>$w_5^*$</td>
<td>640042.26</td>
<td>55.28</td>
<td>-0.02</td>
</tr>
<tr>
<td>$w_6^*$</td>
<td>-1061800.52</td>
<td>41.32</td>
<td>-0.01</td>
</tr>
<tr>
<td>$w_7^*$</td>
<td>1042400.18</td>
<td>-45.95</td>
<td>-0.00</td>
</tr>
<tr>
<td>$w_8^*$</td>
<td>-557682.99</td>
<td>-91.53</td>
<td>0.00</td>
</tr>
<tr>
<td>$w_9^*$</td>
<td>125201.43</td>
<td>72.68</td>
<td>0.01</td>
</tr>
</tbody>
</table>
Probability Theory

Apples and Oranges
Probability Theory

Marginal Probability
\[ p(X = x_i) = \frac{c_i}{N}. \]

Joint Probability
\[ p(X = x_i, Y = y_j) = \frac{n_{ij}}{N}. \]

Conditional Probability
\[ p(Y = y_j | X = x_i) = \frac{n_{ij}}{c_i}. \]
### Probability Theory

**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**

\[
p(X = x_i, Y = y_j) = \frac{n_{ij}}{N} = \frac{n_{ij}}{c_i} \cdot \frac{c_i}{N}
\]

\[
= p(Y = y_j | X = x_i) p(X = x_i)
\]
The Rules of Probability

• Sum Rule
  \[ p(X) = \sum_Y p(X, Y) \]

• Product Rule
  \[ p(X, Y) = p(Y|X)p(X) \]
Bayes’ Theorem

\[ p(Y|X) = \frac{p(X|Y)p(Y)}{p(X)} \]

\[ p(X) = \sum_Y p(X|Y)p(Y) \]

posterior \(\propto\) likelihood \(\times\) prior
Probability Densities

\[ p(x \in (a, b)) = \int_{a}^{b} p(x) \, dx \]

\[ P(z) = \int_{-\infty}^{z} p(x) \, dx \]

\[ p(x) \geq 0 \quad \text{and} \quad \int_{-\infty}^{\infty} p(x) \, dx = 1 \]
Expectations

\[ \mathbb{E}[f] = \sum_{x} p(x) f(x) \]

\[ \mathbb{E}[f] = \int p(x) f(x) \, dx \]

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

Conditional Expectation (discrete)

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

Approximate Expectation (discrete and continuous)
Variances and Covariances

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

\[ \text{cov}[x, y] = \mathbb{E}_{x,y} \left[ \{x - \mathbb{E}[x]\} \{y - \mathbb{E}[y]\} \right] = \mathbb{E}_{x,y}[xy] - \mathbb{E}[x]\mathbb{E}[y] \]

\[ \text{cov}[x, y] = \mathbb{E}_{x,y} \left[ \{x - \mathbb{E}[x]\}\{y^T - \mathbb{E}[y^T]\} \right] = \mathbb{E}_{x,y}[xy^T] - \mathbb{E}[x]\mathbb{E}[y^T] \]
The Gaussian Distribution

\[ N(x|\mu, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp \left\{ -\frac{1}{2\sigma^2}(x - \mu)^2 \right\} \]

\[ N(x|\mu, \sigma^2) > 0 \]

\[ \int_{-\infty}^{\infty} N(x|\mu, \sigma^2) \, dx = 1 \]
Gaussian Mean and Variance

\[ \mathbb{E}[x] = \int_{-\infty}^{\infty} \mathcal{N} \left( x | \mu, \sigma^2 \right) x \, dx = \mu \]

\[ \mathbb{E}[x^2] = \int_{-\infty}^{\infty} \mathcal{N} \left( x | \mu, \sigma^2 \right) x^2 \, dx = \mu^2 + \sigma^2 \]

\[ \text{var}[x] = \mathbb{E}[x^2] - \mathbb{E}[x]^2 = \sigma^2 \]
The Multivariate Gaussian

\[ \mathcal{N}(\mathbf{x}|\mu, \Sigma) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \exp \left\{ -\frac{1}{2} (\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu) \right\} \]
Determinant

Determinant of a matrix: |A|

See:  
http://www.mathsisfun.com/algebra/matrix-determinant.html

Or: ... type det(A) in matlab, like I always do 😊
Curve Fitting Re-visited

\[
y(x, w) = \mathcal{N}(t | y(x_0, w), \beta^{-1})
\]
Math reminder

Logs transform products into sums:

\[ \ln(a_1 a_2 \ldots a_K) = \ln \left( \prod_{k=1}^{K} a_k \right) = \sum_{k=1}^{K} \ln(a_k) \]

Exp is the inverse of log, transforms sums into products:

\[ \ln(\exp(x)) = x \]

\[ \exp \left( \sum_{k=1}^{K} a_k \right) = \prod_{k=1}^{K} \exp(a_k) \]
Maximum Likelihood

\[ p(t|x, w, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n | y(x_n, w), \beta^{-1}) \]

\[
\ln p(t|x, w, \beta) = -\frac{\beta}{2} \sum_{n=1}^{N} \{y(x_n, w) - t_n\}^2 + \frac{N}{2} \ln \beta - \frac{N}{2} \ln(2\pi) \]

\[ \beta E(w) \]

Determine \( w_{ML} \) by minimizing sum-of-squares error, \( E(w) \).

\[
\frac{1}{\beta_{ML}} = \frac{1}{N} \sum_{n=1}^{N} \{y(x_n, w_{ML}) - t_n\}^2
\]
Predictive Distribution

\[ p(t|x, \mathbf{w}_{ML}, \beta_{ML}) = \mathcal{N}(t | y(x, \mathbf{w}_{ML}), \beta_{ML}^{-1}) \]
MAP: A Step towards Bayes

\[ p(w \mid \alpha) = \mathcal{N}(w \mid 0, \alpha^{-1}I) = \left( \frac{\alpha}{2\pi} \right)^{(M+1)/2} \exp \left\{ -\frac{\alpha}{2} w^T w \right\} \]

\[ p(w \mid x, t, \alpha, \beta) \propto p(t \mid x, w, \beta) p(w \mid \alpha) \]

\[ \beta \tilde{E}(w) = \frac{\beta}{2} \sum_{n=1}^{N} \left\{ y(x_n, w) - t_n \right\}^2 + \frac{\alpha}{2} w^T w \]

Determine \( w_{MAP} \) by minimizing regularized sum-of-squares error, \( \tilde{E}(w) \).
Bayesian Predictive Distribution

\[ p(t \mid x, x, t) = \int \int \cdots \int p(t \mid x, x, t, w) p(w \mid x, t) \, dw_1 \cdots dw_N \]

\[ p(t \mid x, x, t) = \mathcal{N} (t \mid m(x), s^2(x)) \]
Bayesian perception
All of our decisions are subject to uncertainty.
Ambiguities
Perception as estimation

\[ \hat{x} = f(s) \]

- **\( x \)**: Events
- **\( s \)**: Sensory input
- **CNS**: Central Nervous System
Perception as bayesian Inference

\[ P(x|s) = \frac{P(s|x)P(x)}{P(s)} \]

Prior \( P(x) \) \hspace{1cm} Likelihood \( P(s|x) \)
Ex: Aperture problem

$X$: Direction of motion

$S$: Sensory input
Ex: Aperture problem

- **Direction**
- **Sensory input**

**Likelihood** $P(s|x)$

**Prior** $P(x)$

**Posterior** $P(x|s)$
Integrating from multiple edges

Cue combination

\[ p(s_1, s_2 \mid x) = P(s_1 \mid x) P(s_2 \mid x) \]

\[ P(x \mid s_1, s_2) = \frac{1}{Z} p(s_1 \mid x) p(s_2 \mid x) \]
Integrating from multiple edges
Integrating from multiple edges

\[ P(s_1 | x) \]

\[ P(s_2 | x) \]

\[ P(x) \]

\[ P(x | s) \]
Example of illusions explained by the prior on low velocity

Velocity perception as bayesian inference: Weiss, Simoncelli and Adelson, 2002
Use of a prior in sensorimotor control

Kording and Wolpert, Nature, 2004
Prediction:

Experimental results

Experimental result:
Ex: The ventriloquism effect
Combining cues from several modalities

\[ p(x \mid s_a, s_v) = \frac{P(x) P(s_a \mid x) P(s_v \mid x)}{Z} \]
Combining cues from several modalities: Gaussian distributions

\[ p(s_a \mid x) = \frac{1}{\sqrt{2\pi}\sigma_a} \exp \left( -\frac{(x - x_a)^2}{2\sigma_a^2} \right) \]

\[ p(s_v \mid x) = \frac{1}{\sqrt{2\pi}\sigma_v} \exp \left( -\frac{(x - x_v)^2}{2\sigma_v^2} \right) \]

\[ p(x \mid s_a, s_v) = \frac{1}{\sqrt{2\pi}\sigma_b} \exp \left( -\frac{(x - x_b)^2}{2\sigma_b^2} \right) \]
Combining cues from several modalities, Gaussian distributions

\[ x_b = \frac{\sigma_a^2}{\sigma_a^2 + \sigma_v^2} x_v + \frac{\sigma_v^2}{\sigma_a^2 + \sigma_v^2} x_a \]

Visual weight
Visual cue
Auditory weight
Auditory cue
Analogy with the center-of-mass

Visual more reliable: Visual capture

Auditory more reliable: Auditory capture:
Ernst and Banks, 2002

From: Banks et al, Nature, 2002
Temporal predictions

\[ p(x_{t+1} | s_{0\rightarrow t+1}) = \frac{p(s_{t+1} | x_{t+1}) \sum_{x_t} p(x_{t+1} | x_t) p(x_t | s_{0\rightarrow t})}{Z} \]
Gaussian distribution: Kalman filter

\[ \hat{x}_{t+1} = (1 - k_t) \hat{x}_t^f + k_t x_t^s \]

\[ k_t = \frac{\sigma_s^2}{\sigma_s^2 + \sigma_t^2} \]

Kalman gain
Ex: Optimal motor control with gaussian distributions

\[ \hat{x}(t+1) = (1 - K(t)) \hat{x}_p(t) + K(t) x_s(t) \]

Reliability of internal estimates versus sensory feedback

Gharahmani and Wolpert, 2002
Experimental evidence: Sensorimotor control

Purely sensory feedback

Purely forward estimate (no sensory feedback)

Optimal kalman filter

Wolpert et al, 1995
Control of saccade sequences

\[ \hat{x}_{t+1} = \left(1 - k_t \right) \hat{x}_t^f + k_t x_t^s \]

\[ k_t = \frac{\sigma_s^2}{\sigma_s^2 + \sigma_t^2} \]

Munuera, Morel and Deneve, J Neurosci 2009
Experimental evidence: saccadic eye movements

\[ k = 1 \]

\[ k = 0 \]

\[ \sigma^2 \]

\[ s \]

\[ t \]

\[ \sigma_s \]

\[ \sigma_t \]

\[ k_t = \frac{\sigma_s^2}{\sigma_s^2 + \sigma_t^2} \]
Bayesian perception: multiple causes

Infering 3D structures from 2D images.
Knill and Richards, 1996

\[ p(x_1 | s) = \frac{p(x_1) \sum_{x_2} p(s | x_1, x_2) p(x_2)}{Z} \]
Bayesian perception and action: multiple causes

Explaining away

\[ p(x_1 | s) = \frac{p(x_1) \sum_{x_2} p(s | x_1, x_2) p(x_2)}{Z} \]
Example: Cues should be integrated only when they have the same source.

Kording et al, PloS 2007

«Strong fusion»
\[x \leftarrow S_v \rightarrow S_a\]

«Weak fusion»
\[x_v \leftarrow S_v \rightarrow x \leftarrow S_a \rightarrow x_a\]

Common source likely

Common source unlikely
Cues should be integrated only when they have the same source.

Kording et al, PloS 2007

Explaining away

“Same source”

Attraction

“Different source”

Repulsion

“Different source”
Experimental results

Kording et al, PloS 2007