ENGN 4520: Introduction to Machine Learning

Alex Smola, RSISE ANU
Teaching Period
Problem Sheet — Week 2
April 30 to June 8, 2001
The due date for these problems is Monday, May 14

A Theory

Problem 7 (Games, 8 Points)

1. *Compute the expected loss and the variance when betting one dollar on even (18 out of 37), and when betting on a pair (2 out of 37) numbers at the casino.*
2. *Assume we toss a coin m times (m even). What is ths probability of observing heads $\frac{m}{2}$ times, what is the probability of never observing heads (***Hint:*** Binomial distribution).*
3. *What is the probability that a tainted coin with* $\Pr(\text{Head}) = 0.6$ *instead of a normal coin with* $\Pr(\text{Head}) = 0.5$ *generates* 60 *out of* 100 *heads. Assume that it is twice as likely that the coin has not been tampered with (***Hint:*** use Bayes' formula).*
4. *Show that if we play the game of matching two coins (both heads or both tails vs. mixed heads and tails) it is sufficient that one coin produces heads and tails with equal probability to ensure that the game is fair (i.e. matches and mismatches occur with equal probability).*
5. **Bonus question** — *don't play! Go to the Canberra Casino and analyze the wheel of fortune. At which choice are you losing the least amount of money?*

Problem 8 (Normal Distributions, 5 Points)

Assume that we have a normal distribution with mean $\mu \in \mathbb{R}^{m+n}$ and covariance matrix $\Sigma \in \mathbb{R}^{(m+n)\times(m+n)}$.

1. *Write out the density for $\mathbf{x} \in \mathbb{R}^{m+n}$ distributed according to this normal distribution.*
2. *Now assume that we observe the first m variables. Compute the conditional mean of the remaining n variables under these circumstances.*
3. *Compute the conditional variance, given the first m variables.*

Problem 9 (Median and Trimmed Mean, 7 Points)

Assume that we observe m numbers $X := \{x_1, \ldots, x_m\} \subset \mathbb{R}$ generated iid according to some $p(x|\theta)$.

1. *Under the assumption that $p(x|\mu, \sigma)$ is a normal distribution with mean μ and variance σ, compute the maximum likelihood values of μ, σ given X.*
2. *Under the assumption that $p(x|\mu, \sigma)$ is a symmetric Laplacian distribution with mean μ and (known) variance σ, show that the maximum likelihood value of μ is the median of X.*
3. *Compute the maximum likelihood variance for the Laplacian distribution.*

4. **Bonus Question**. *For the probability distribution*

$$p(x|\sigma,\mu) = c(\sigma) \begin{cases} \exp\left(-\frac{1}{2\sigma}(x-\mu)^2\right) & \text{for } |x-\mu| \leq \sigma \\ \exp\left(-|x-\mu| + \frac{\sigma}{2}\right) & \text{otherwise} \end{cases} \quad \text{for some } c(\sigma)$$

show that obtaining the maximum likelihood value with respect to μ leads to a trimmed estimation procedure (i.e. we throw out the largest and smallest few x_i.

B Programming

Problem 10 (Perceptron Algorithm, 20 Points)

1. *Implement in C/MATLAB the perceptron algorithm presented in the lecture. It should take $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)$ and η as an argument and return $\mathbf{w}$.*

   Hint: *return $\mathbf{x}_1, \ldots, \mathbf{x}_m$ as a matrix and $y_1, \ldots, y_m$ as a vector.*

2. *Write a data generation program in C/MATLAB which takes $\mathbf{w}^*, b^*, p, m, n$ as input and which produces m pairs of points $\mathbf{x} \in \mathbb{R}^n$ and $y \in \{\pm 1\}$ where $\mathbf{x}$ is drawn from a normal distribution with zero mean and unit variance. Moreover, with probability $1-p$ we have $y = \operatorname{sgn}\left(\mathbf{w}^\top \mathbf{x} + b\right)$ and otherwise y is randomly assigned either 1 or -1.*

   Hint: *you can use the* `randn` *and* `rand` *routines of MATLAB.*

3. *Test your implementation of the Perceptron Algorithm for noise free ($p = 0$) random training data by computing*

$$\psi := \|(\mathbf{w}, b)\|^{-1} \langle (\mathbf{w}, b), (\mathbf{w}^*, b^*) \rangle$$

   and the number of errors during learning for the following range of parameters $n \in \{2, 5, 10, 20, 50, 100\}$ and $m \in \{10, 20, 50, 100, 200, 1000\}$ and subsequently plotting ψ vs. the different values in a 3D plot (use `surf` *in MATLAB).*

   Hint: *write the program for a fixed m, n first and then script the data generation with two for loops over m and n.*

4. *Test your implementation for $n = 10$ and $m \in \{10, 20, 50, 100, 200, 1000\}$ for $p \in \{0, 0.01, 0.02, 0.05, 0.1, 0.2\}$. Again, plot ψ and the number of errors during training.*