Landau symbol ("big O"): $O(\cdot)$ for order of magnitude of complexity

$O(N^k)$ → exponential computational complexity

$$\lim_{N \to \infty} \frac{O(N^k)}{N^k} = A \neq 0, \infty$$

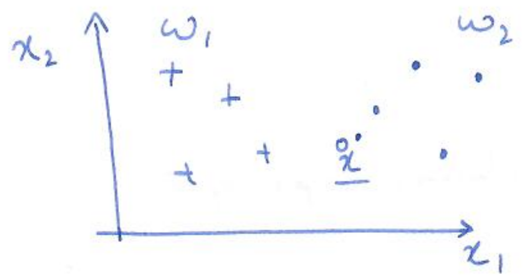
i.e. $O(N^2)$ means AN^2 is the biggest contributor to the complexity

$AN^2 + O(N)$, $A \neq 0$ = constant

4.2.2 k NN (k - Nearest Neighbours)

Idea: Follow the majority of k nearest training samples.

$k \in \mathbb{N}$



$k = 1$: look for the closest neighbour.
In this case ω_2

$k = 3$: We have 2 from ω_1 and 1 from ω_2.
So $\hat{y}(\underline{x}) = \omega_1$

k = 1 leads to overfitting and should not be used

4.3 Bayes Plug-in

We use Bayes decision theory here Ch 2.

Bayesian decision theory:

$$p(\underline{x}|\omega_j),\ p(\omega_j) \longrightarrow \text{MBR/MAP/ML/NP decision rule}$$

In practice
$p(\underline{x}|\omega_j), p(\omega_j) \to$ unknown

Idea: $p(\underline{x}/\omega_j)$ known except for some unknown parameters $\underline{\nu}_j$.

i.e. $p(\underline{x}|\omega_j;\ \underset{\downarrow}{\underline{\nu}_j})$

$\qquad\qquad\qquad\quad \hat{\ }$

eg. $\underline{x}/\omega_j \sim N(\underline{\mu}_j,\ \underline{\underline{C}}_j)$

$\underline{\nu}_j = \{\underline{\mu}_j, \underline{\underline{C}}_j\}$

Training:

Estimate $\underline{\nu}_j$, $p(\omega_j)$ from N_j training samples of class ω_j

$\Rightarrow \hat{\nu}_j, \hat{p}(\omega_j)$

Classification

use $\hat{p}(\underline{x}/\omega_j, \hat{\underline{\nu}}_j)$ and $\hat{p}(\omega_j)$ in Bayesian decision

How to ~~use~~ estimate $\underline{\nu}_j$, $p(\omega_j)$?

4.3.1 ML parameter estimation

A crash course: see SASP.

- given sample $\underline{x}$ of pdf $p(\underline{x};\underline{\vartheta})$

are known (for $\underline{x}$), not known (for $\underline{\vartheta}$)

- ML estimation:

$$\hat{\underline{\vartheta}}_{ML}(\underline{x}) = \arg\max_{\underline{\vartheta}} p(\underline{x};\underline{\vartheta}) \longrightarrow \text{likelihood}$$

$$= \arg\max_{\underline{\vartheta}} \ln p(\underline{x};\underline{\vartheta}) \rightarrow \text{log-likelihood}$$

- Necessary condition:

$$\underline{\nabla} \ln p(\underline{x};\underline{\vartheta})\Big|_{\underline{\vartheta}=\hat{\underline{\vartheta}}_{ML}} \stackrel{!}{=} \underline{0}$$

$$\underline{\nabla} = \begin{bmatrix} \frac{\partial}{\partial \vartheta_1} \\ \vdots \\ \frac{\partial}{\partial \vartheta_M} \end{bmatrix} \rightarrow \text{gradient vector}$$

$\Rightarrow$ a linear / non-linear eq.^{n} system

$\Rightarrow$ analytical / numerical sol.^{n}

<u>E4.4</u> ML estimation for Gaussian distribution.

$\underline{x}_n \in \mathbb{R}^d \quad (1 \le n \le N)$, iid $N(\underline{\mu}, \underline{\underline{C}})$

$$p(\underline{x}_n;\underline{\vartheta}) = \frac{1}{(2\pi)^{d/2}\sqrt{|\underline{\underline{C}}|}} \exp\left(-\frac{1}{2}(\underline{x}_n-\underline{\mu})^T C^{-1}(\underline{x}_n-\underline{\mu})\right)$$

$$p(x_1 \dots x_N;\underline{\vartheta}) \stackrel{iid}{=} \prod_{i=1}^{N} p(\underline{x}_i;\underline{\vartheta})$$

log - likelihood:

$$L(\underline{\gamma}) = \ln p(\underline{x}_1 \cdots \underline{x}_N ; \underline{\gamma})$$

$$= \text{constant} - \frac{N}{2} \ln |\underline{\underline{C}}| - \frac{1}{2} \sum_{n=1}^{N} (\underline{x}_n - \underline{\mu})^T \underline{\underline{C}}^{-1} (\underline{x}_n - \underline{\mu})$$

$\underline{\gamma} = \{\underline{\mu}, \underline{\underline{C}}\}$ contains d elements of $\underline{\mu}$

and $\frac{d(d+1)}{2}$ " " $\underline{\underline{C}} = \underline{\underline{C}}^T$

$$\nabla L(\underline{\gamma}) \stackrel{!}{=} \underline{0}$$

4-21

4.29.1

$$\hat{\underline{\mu}} = \frac{1}{N} \sum_{n=1}^{N} \underline{x}_n \qquad \text{for } \underline{\mu} = E[\underline{x}]$$

$$\hat{\underline{\underline{C}}} = \frac{1}{N} \sum_{n=1}^{N} (x_n - \hat{\mu}) (\underline{x} - \hat{\underline{\mu}})^T \quad \text{for } \underline{\underline{C}} = E[(\underline{x}-\underline{\mu})(\underline{x}-\underline{\mu})^T]$$

slide 4.29

4.29.1

E 4.5 cont : Solution

Method of Lagrange multiplier:

$$\max_{P_j, \lambda} \tilde{L}(P_1 \cdots P_c, \lambda) = L(P_1 \cdots P_c) + \lambda (P_1 + \cdots + P_c - 1)$$

$$\left.\frac{\partial \tilde{L}}{\partial P_j}\right|_{P=\hat{P}_j} = \frac{N_j}{P_j} + \lambda \stackrel{!}{=} 0 \Rightarrow \hat{P}_j = -\frac{1}{\lambda} N_j \sim N_j$$

$$\sum_{j=1}^{C} \hat{P}_j = -\frac{1}{\lambda} \sum_{j=1}^{C} N_j = \frac{-N}{\lambda} = 1 \qquad \Rightarrow \hat{P}_j = \frac{N_j}{N}$$

$$\Rightarrow \lambda = -N$$

4.3.2 Gaussian classifier

Idea: • Bayes plug-in method
• Assumption: Gaussian likelihood
$$\underline{x} \mid \omega_j \sim N(\underline{\mu}_j, \underline{\underline{C}}_j)$$

ML estimate of $\underline{\mu}_j$, $\underline{\underline{C}}_j$, $P(\omega_j)$ according to E 4.4, E 4.5 from training examples.

slide 4-30

4-30.1

Comparision to nearest mean (4.2.1)

• nearest mean:
$$\min_j \quad D_{maha} = (\underline{x} - \underline{\mu}_j)^T \underline{\underline{C}}_j^{-1} (\underline{x} - \underline{\mu}_j)$$

or $\max_j \; -D_{maha} = -(\underline{x} - \underline{\mu}_j)^T \underline{\underline{C}}_j^{-1} (\underline{x} - \underline{\mu}_j)$

• Gaussian classifier + MAP decision rule:
$$p(\underline{x} \mid \omega_j; \underline{\hat{v}}_j) \sim N(\hat{\underline{\mu}}_j, \hat{\underline{\underline{C}}}_j)$$
$$\max_j \; \ln p(\underline{x} \mid \omega_j; \underline{v}_j) \cdot \hat{P}(\omega_j)$$
$$= -\frac{1}{2} (\underline{x} - \mu_j)^T C_j^{-1} (\underline{x} - \mu_j) - \frac{1}{2} \ln |\hat{\underline{\underline{C}}}_j| + \ln \hat{P}_j$$

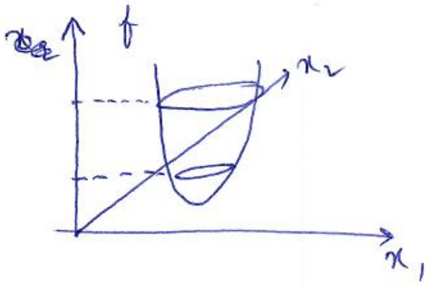
Ch5 : Advanced Optimization Techniques

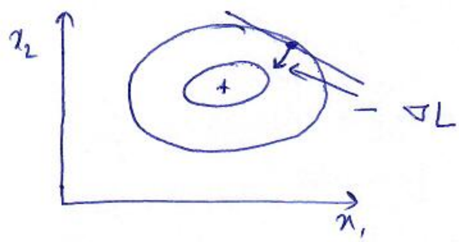
5.1 Challenges in Optimization

Slide 5-1

Sl5.4.1

2 D visualization $f(\underline{x}) = f(x_1, x_2)$





Contour lines of $f(\underline{x})$

$= \{ \underline{x} \mid f(\underline{x}) = \text{constant} \}$

Slide 5-5