

Special Cases of Maximum-Likelihood Estimation

ECE 6279: Spatial Array Processing
Spring 2011
Lecture 28

Prof. Aaron D. Lanterman

School of Electrical & Computer Engineering
Georgia Institute of Technology
AL: 404-385-2548
<lanterma@ece.gatech.edu>

Copyright 2011 Aaron Lanterman



Our Deterministic Gaussian Model

- N_s sources in additive noise

$$\underline{\mathbf{y}}(l) = \sum_{n=1}^{N_s} \mathbf{e}(\boldsymbol{\theta}_n) s_n(l) + \underline{\mathbf{n}}(l)$$

$$\underline{\mathbf{n}} \sim CN(0, \mathbf{K}_n)$$

$$\mathbf{s}(l) = \begin{bmatrix} s_1(l) \\ \vdots \\ s_{N_s}(l) \end{bmatrix} \quad \begin{array}{l} \text{Modeled as deterministic} \\ \text{parameters we need to} \\ \text{estimate} \end{array}$$

Copyright 2011 Aaron Lanterman



The Single-Source Case

- Recall ML estimate of signal for a fixed Θ

$$\hat{\mathbf{s}} = \left[\mathbf{D}^H(\Theta) \mathbf{K}_n^{-1} \mathbf{D}(\Theta) \right]^{-1} \mathbf{D}(\Theta)^H \mathbf{K}_n^{-1} \mathbf{y}$$

- Suppose we only have one source

$$\hat{s} = \left[\mathbf{e}^H(\boldsymbol{\theta}) \mathbf{K}_n^{-1} \mathbf{e}(\boldsymbol{\theta}) \right]^{-1} \mathbf{e}(\boldsymbol{\theta})^H \mathbf{K}_n^{-1} \mathbf{y}$$

Copyright 2011 Aaron Lanterman



A Blast from the Past

$$\hat{s} = \frac{\mathbf{e}^H(\boldsymbol{\theta}) \mathbf{K}_n^{-1} \mathbf{y}}{\underbrace{\mathbf{e}^H(\boldsymbol{\theta}) \mathbf{K}_n^{-1} \mathbf{e}(\boldsymbol{\theta})}_{\equiv \mathbf{a}^H}} = \mathbf{a}^H \mathbf{y}$$

- Recall $\mathbf{a}^H \propto \mathbf{e}^H(\boldsymbol{\theta}) \mathbf{K}_n^{-1}$ are the weights that maximize SNR

–Showed this
in Lecture 13

Copyright 2011 Aaron Lanterman



Another Blast from the Past

- If $\mathbf{K}_n = \sigma_n^2 \mathbf{I}$ then

$$\hat{s} = \frac{\mathbf{e}^H(\boldsymbol{\theta}) \mathbf{K}_n^{-1} \mathbf{y}}{\mathbf{e}^H(\boldsymbol{\theta}) \mathbf{K}_n^{-1} \mathbf{e}(\boldsymbol{\theta})} = \frac{\mathbf{e}^H(\boldsymbol{\theta})}{\mathbf{e}^H(\boldsymbol{\theta}) \mathbf{e}(\boldsymbol{\theta})} \mathbf{y}$$

$$= \frac{\mathbf{e}^H(\boldsymbol{\theta})}{M} \mathbf{y}$$

- Old delay-and-sum conventional beamformer



Copyright 2011 Aaron Lanterman

Maximum-Likelihood Procedure

- ML procedure said to maximize $\text{tr}\{\mathbf{D}(\mathbf{D}^H \mathbf{D})^{-1} \mathbf{D}^H \hat{\mathbf{R}}_y\}$ over Θ

- For one source

$$\text{tr}\{\mathbf{e}(\mathbf{e}^H \mathbf{e})^{-1} \mathbf{e}^H \hat{\mathbf{R}}_y\} = \frac{1}{M} \text{tr}\{\mathbf{e} \mathbf{e}^H \hat{\mathbf{R}}_y\}$$

$$= \frac{1}{M} \text{tr}\{\mathbf{e}^H \hat{\mathbf{R}}_y \mathbf{e}\} = \underbrace{\mathbf{e}^H(\boldsymbol{\theta}) \hat{\mathbf{R}}_y \mathbf{e}(\boldsymbol{\theta})}_{\text{Power out of our old conventional delay-and-sum beamformer!}} / M$$

Power out of our old conventional delay-and-sum beamformer!



Copyright 2011 Aaron Lanterman

Take-Home Messages (1)

- If you have a **single source in white Gaussian noise**, maximizing the SNR turns out to be optimal (in the ML estimation sense) under the deterministic Gaussian model
- If you have non-Gaussian noise or multiple sources, maximizing SNR is **not** the optimal ML procedure
 - But usually still pretty useful!



Copyright 2011 Aaron Lanterman

Take-Home Message (2)

- If you have a **single source in white Gaussian noise**, the conventional delay-and-sum beamformer is optimal in the ML sense under the deterministic Gaussian model
- If you have non-Gaussian noise or multiple sources the conventional beamformer is **not** the optimal ML procedure
 - But usually still pretty useful!



Copyright 2011 Aaron Lanterman

Capon's Argument (1)

- ML signal estimate for one source is
$$\hat{\mathbf{s}} = \frac{\mathbf{e}^H(\boldsymbol{\theta})\mathbf{K}_n^{-1}}{\mathbf{e}^H(\boldsymbol{\theta})\mathbf{K}_n^{-1}\mathbf{e}(\boldsymbol{\theta})}\mathbf{y}$$
- In Capon's application (see p. 358-359 of J&D), he didn't have a good estimate of \mathbf{K}_n
- Capon handwaved and said to try replacing \mathbf{K}_n with $\hat{\mathbf{R}}_y$

Copyright 2011 Aaron Lanterman



Capon's Argument (2)

- Totally ad-hoc from an ML standpoint; $\hat{\mathbf{R}}_y$ is a terrible estimate of \mathbf{K}_n
- But if we go ahead and do it, we get
$$\hat{\mathbf{s}} = \frac{\mathbf{e}^H(\boldsymbol{\theta})\hat{\mathbf{R}}_y^{-1}}{\mathbf{e}^H(\boldsymbol{\theta})\hat{\mathbf{R}}_y^{-1}\mathbf{e}(\boldsymbol{\theta})}\mathbf{y}$$
 which is the MVDR beamformer!

Copyright 2011 Aaron Lanterman



A Great Confusion

- MVDR (a.k.a. Capon's method) is sometimes erroneously referred to as "maximum likelihood"
 - That's nonsense! MVDR does not arise from any known statistical ML procedure
- Our earlier derivation of MVDR from a constrained optimization procedure is due to Lacoss (1971)

Copyright 2011 Aaron Lanterman



Recollections and Observations

- Remember MVDR was a solution to
$$\mathbf{w}^H\mathbf{R}_y\mathbf{w} \text{ s.t. } \mathbf{w}^H\mathbf{e}(\boldsymbol{\theta}) = 1$$
- Yes, this is an optimization problem; but it's not a maximum-likelihood optimization problem
 - Doesn't require a probability model for the data



Copyright 2011 Aaron Lanterman



Our Stochastic Gaussian Model

- N sources in additive noise

$$\underline{\mathbf{y}}(l) = \sum_{n=1}^N \mathbf{e}(\boldsymbol{\theta}_n) \underline{s}_n(l) + \underline{\mathbf{n}}(l)$$

$$\underline{\mathbf{n}} \sim CN(0, \mathbf{K}_n)$$

$$\underline{\mathbf{s}} = \begin{bmatrix} \underline{s}_1 \\ \vdots \\ \underline{s}_N \end{bmatrix} \sim CN(0, \mathbf{K}_s)$$

Copyright 2011 Aaron Lanterman



Maximum-Likelihood Estimation

- Goal: maximize

$$-L \ln \det \mathbf{K}_y(\boldsymbol{\xi}) - L \operatorname{tr}[\hat{\mathbf{K}}_y \mathbf{K}_y^{-1}(\boldsymbol{\xi})]$$

where

$$\mathbf{K}_y(\boldsymbol{\xi}) = \mathbf{D}(\boldsymbol{\Theta}) \mathbf{K}_s \mathbf{D}^H(\boldsymbol{\Theta}) + \mathbf{K}_n$$

over $\boldsymbol{\xi}$, which represents all the parameters (angles, signal powers, correlation coefficients)

Copyright 2011 Aaron Lanterman



Asymptotic ML Est. for Stochastic Model

- If you have
 - “Lots of snapshots,”
 - Uncorrelated sources,
 - and I.I.D. receiver noise
- Then VT-IV, pp. 985-991 shows that you can get rid of the power parameters $\sigma_1^2, \dots, \sigma_N^2$ and find ML estimates of angles by maximizing...

Copyright 2011 Aaron Lanterman



Asymptotic ML Procedure

- New goal: maximize

$$-\ln \det[\mathbf{P}_D \hat{\mathbf{K}}_y \mathbf{P}_D + \sigma_n^2 \mathbf{P}_D^\perp] - \operatorname{tr}[\mathbf{P}_D^\perp \hat{\mathbf{K}}_y] / \sigma_n^2$$

over $\boldsymbol{\Theta}$

$$\text{where } \mathbf{P}_D = \mathbf{D}(\mathbf{D}^H \mathbf{D})^{-1} \mathbf{D}^H$$

$$\mathbf{P}_D^\perp = \mathbf{I} - \mathbf{P}_D$$

Copyright 2011 Aaron Lanterman

