STANFORD

Lecture 3

MMSE Estimation: & Information Measures

April 9, 2024

JOHN M. CIOFFI

Hitachi Professor Emeritus of Engineering

Instructor EE379B – Spring 2024

Announcements & Agenda

- Announcements
  - Problem Set 1 due Wednesday at 17:00
  - Most relevant reading – 1.5, 2.3, D.1, D.2, 4.1
  - Problem Set 2 due April 17 at 17:00

- Problem Set 2 = PS2 due 4/19 at 17:00
  1. 4.29 biases and error probability
  2. 4.36 MMSE spatial equalizer
  3. 2.10 Entropy and Dimensionality
  4. 2.14 Bandwidth vs Power
  5. 2.20 MMSE and Entropy

- Agenda
  - General MMSE & Gaussian
    - Autocorrelation/Cross-Correlation
  - Linear MMSE & The Orthogonality Principle
    - Biases and SNRs
  - Examples
  - Information Measures – generalizing Gaussian to all distributions
  - Mutual Information and MMSE

AWGNs, or whited AGNs:

MMSE is canonical &
asymptotically MAP,
and also close @ finite N.
Max reliable b is capacity.

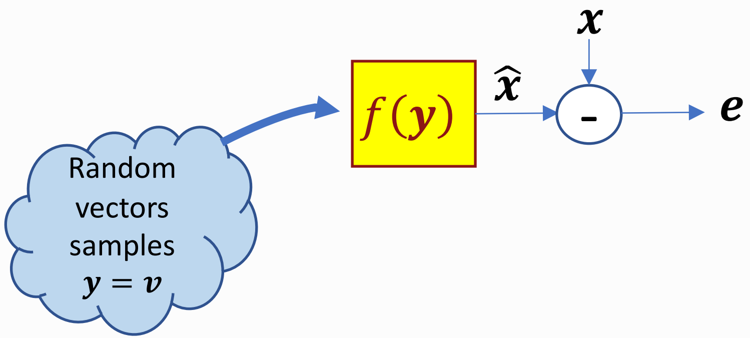
General MMSE and Gaussian

Section D.1

When all random processes are (stationary) Gaussian, Linear MMSE is the overall best MMSE.

The MMSE Estimation Problem

- Given random $\boldsymbol{x}$ and $\boldsymbol{y}$, design estimates $\boldsymbol{x}$ as $\widehat{\boldsymbol{x}} = f(\boldsymbol{y})$.
  - Designer knows both $p_{\boldsymbol{x},\boldsymbol{y}}$ and specific observed $\boldsymbol{y} = \boldsymbol{v}$.
  - $\boldsymbol{x}$ and $\boldsymbol{y}$'s distributions are continuous. (random code on $\boldsymbol{x}$.)



- The error is:
  - $\boldsymbol{e} = \boldsymbol{x} - f(\boldsymbol{y})$.

- Its mean-square is
  - $\mathbb{E}_{\boldsymbol{x},\boldsymbol{y}}[\|\boldsymbol{e}\|^2] = \mathbb{E}[\|\boldsymbol{x} - f(\boldsymbol{y})\|^2]$.
- Its minimum over f is the MMSE.

$$MMSE = \min_f \mathbb{E}\left[\|\boldsymbol{x} - f(\boldsymbol{y})\|^2\right].$$

- Solution is
  - the conditional mean of $\boldsymbol{x}$ given $\boldsymbol{y}$; $\widehat{\boldsymbol{x}} = \mathbb{E}[\boldsymbol{x}/\boldsymbol{y}]$,
  - From $p_{\boldsymbol{x}/\boldsymbol{y}}$ the *à posteriori* distribution (used for MAP detector).
  - Proof: see Appendix D.1
  - Its linear: $(\widehat{\boldsymbol{x}_1 + \boldsymbol{x}_2}) = \widehat{\boldsymbol{x}}_1 + \widehat{\boldsymbol{x}}_2$.

$$\widehat{\boldsymbol{x}} = \mathbb{E}[\boldsymbol{x}/\boldsymbol{y}] = W \cdot \boldsymbol{y}$$

Auto- & Cross- correlation

- **Autocorrelation** generalizes mean-square. When stationary,
  - it samples $R_{\boldsymbol{xx}}(\tau) = \mathbb{E}[\boldsymbol{x}(t) \cdot \boldsymbol{x}^*(t-\tau)]$ where time is the dimension and $\tau = kT'$ is correlation interval.
  - Vector process samples may correspond to:
    - frequency-time – time samples $x(kT)$ are the vector elements.
    - space time sets $\tau = 0$ – spatial samples (think antennas),
      - often at each $n \cdot \left(\frac{1}{T}\right)$.

$$R_{\boldsymbol{xx}} = \mathbb{E}[\boldsymbol{x} \cdot \boldsymbol{x}^*] \qquad R_{\boldsymbol{yy}} = \mathbb{E}[\boldsymbol{y} \cdot \boldsymbol{y}^*]$$

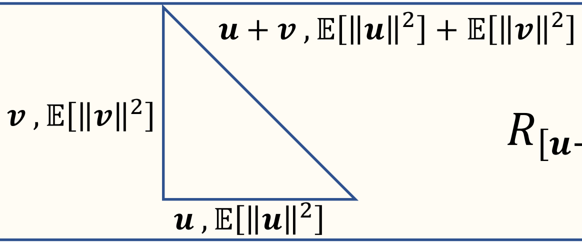
- **Energy** over symbol if FT, over space dimensions:
  - Both cases $\tau = 0$ (non xtalk/energy) terms are on diagonal.

$$\mathcal{E}_{\boldsymbol{x}} = trace\{R_{\boldsymbol{xx}}\} = \mathbb{E}[\boldsymbol{x}^* \cdot \boldsymbol{x}] = \mathbb{E}[\|\boldsymbol{x}\|^2]$$

- **Cross correlation** "generalizes" inner product:
  - It samples $R_{xx}(\tau) = \mathbb{E}[x(t) \cdot y^*(t-\tau)]$.
  - Vectors can be different lengths L_x and L_y.
  - "uncorrelated" (=0) → **orthogonal.**
  - Nondiagonal dimensions.

$$R_{\boldsymbol{xy}} = \mathbb{E}[\boldsymbol{x} \cdot \boldsymbol{y}^*] \qquad R_{\boldsymbol{yx}} = \mathbb{E}[\boldsymbol{y} \cdot \boldsymbol{x}^*]$$

- **Pythagorus** IF uncorrelated $R_{\boldsymbol{uv}} = 0$.
  - Generalizes "variances of uncorrelated random variables add."



$$R_{[\boldsymbol{u}+\boldsymbol{v}][\boldsymbol{u}+\boldsymbol{v}]} = R_{\boldsymbol{uu}} + R_{\boldsymbol{vv}}$$

The Joint Gaussian Distribution

- is completely specified by its autocorrelation (and cross correlation):

$$R \triangleq R_{\begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix}[\boldsymbol{x}^* \quad \boldsymbol{y}^*]} = \begin{bmatrix} R_{\boldsymbol{xx}} & R_{\boldsymbol{xy}} \\ R_{\boldsymbol{yx}} & R_{\boldsymbol{yy}} \end{bmatrix}.$$

- Its marginal distributions for $\boldsymbol{x}$ and $\boldsymbol{y}$
  - are also Gaussian.
- Its conditional distributions are Gaussian.
  - In particular, with non-zero mean $\mathbb{E}[\boldsymbol{x}/\boldsymbol{y}]$.

$$\text{real:} \quad p(\boldsymbol{x},\boldsymbol{y}) = (2\pi)^{-\frac{N_x+N_y}{2}} \cdot |R|^{-1/2} \cdot e^{-\frac{1}{2}\left\{[\boldsymbol{x}^*\boldsymbol{y}^*]\cdot R^{-1}\cdot\begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix}\right\}}$$

$$\text{complex:} \quad p(\boldsymbol{x},\boldsymbol{y}) = (\pi)^{-[N_x+N_y]} \cdot |R|^{-1} \cdot e^{-\left\{[\boldsymbol{x}^*\boldsymbol{y}^*]\cdot R^{-1}\cdot\begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix}\right\}}$$

- Singularity?
  - $|R_{\boldsymbol{yy}}| > 0$ with nonsingular noise.
  - $|R_{\boldsymbol{xx}}|$? $|R|$? – use pseudoinverse and determinant as product of **nonzero** eigenvalues.

$$\mathbb{E}[\boldsymbol{x}/\boldsymbol{y}] = \underbrace{R_{\boldsymbol{xy}} \cdot R_{\boldsymbol{yy}}^{-1}} \cdot \boldsymbol{y}$$

It's linear (for Gaussian).

$W = R_{\boldsymbol{xy}} \cdot R_{\boldsymbol{yy}}^{+}$ if singular.

MMSE and AWGN's best transmission are fundamentally connected.

Linear MMSE & The Orthogonality Principle

Section D.2

Decomposition into pass spaces and null spaces becomes critical in canonical design (think VC).

Linear MMSE: <u>any</u> joint distribution of x and y

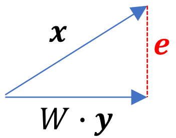
- Given random $\boldsymbol{x}$ and $\boldsymbol{y}$, receiver estimates $\boldsymbol{x}$, $\hat{\boldsymbol{x}} = W \cdot \boldsymbol{y}$.
  - It knows both $p_{\boldsymbol{x},\boldsymbol{y}}$ and specific observed $\boldsymbol{y} = \boldsymbol{v}$.
- The error is:

$$\boldsymbol{e} = \boldsymbol{x} - \sum_{n=1}^{N} \boldsymbol{w}_n \cdot \boldsymbol{y}_n = \boldsymbol{x} - W \cdot \boldsymbol{y}.$$

- Its mean-square is
  - $\mathbb{E}_{\boldsymbol{x},\boldsymbol{y}}[\|\boldsymbol{e}\|^2] = \mathbb{E}\,[\|\boldsymbol{x} - W \cdot \boldsymbol{y}\|^2]$.

Orthogonality Principle

- Its minimum occurs when $\mathbb{E}[\boldsymbol{e} \cdot \boldsymbol{y}_n^*] = \boldsymbol{0}$ for all n:
  - Proof is in Appendix D.2.
  - That is, the error and the estimator's input are uncorrelated.



- Minimum $\hat{\boldsymbol{x}} = \underbrace{R_{\boldsymbol{xy}} \cdot R_{\boldsymbol{yy}}^{-1}}_{W} \cdot \boldsymbol{y}$, linear in $\boldsymbol{y}$, **so true MMSE if Gaussian**.
  - The true MMSE estimator may not be linear if non-Gaussian.
- Also again: $(\widehat{\boldsymbol{x}_1 + \boldsymbol{x}_2}) = \hat{\boldsymbol{x}}_1 + \hat{\boldsymbol{x}}_2$ or $\widehat{A \cdot \boldsymbol{x}} = A \cdot \hat{\boldsymbol{x}}$.

MMSE Matrix: $R_{\boldsymbol{ee}} = R_{\boldsymbol{xx}} - R_{\boldsymbol{xy}} \cdot R_{\boldsymbol{yy}}^{-1} \cdot R_{\boldsymbol{yx}} = R_{\boldsymbol{x}/\boldsymbol{y}}^{\perp}$

$= R_{\boldsymbol{xx}} - W \cdot R_{\boldsymbol{yx}}$

Vector and Matrix Norms

- The trace of an autocorrelation matrix is its norm (and also equal to mean-squared length of random vector).
- MMSE $=\mathbb{E}[||\boldsymbol{e}||^2] = trace\{\, R_{\boldsymbol{ee}}\}$.
- The **trace** of a square autocorrelation matrix is also equal to the **sum** of its eigenvalues $\mathcal{E}_{e',n}$:

$$\boldsymbol{e}' = Q \cdot \boldsymbol{e} \qquad diagonal\text{:}\ R_{\boldsymbol{e'e'}} = Q \cdot\ R_{\boldsymbol{ee}} \cdot Q^* \,.$$

$$||\boldsymbol{e}||^2 = ||\boldsymbol{e}'||^2 \text{ because } QQ^* = Q^*Q = I \,.$$

- The **determinant** of an autocorrelation matrix is the **product** of its eigenvalues $\mathcal{E}_{e',n}$:
- MMSE $=\mathbb{E}[||\boldsymbol{e}||^2]$ and $\ln|\, R_{\boldsymbol{ee}}| = \sum_n \ln \mathcal{E}_{e',n}$.
- The minimization of each component of $\boldsymbol{e}$ is *variables separable* (has its own row of W),
  - so then the sum is minimized,
  - but this means each of the $\boldsymbol{e}'$ also (W→$Q \cdot W$) minimized,
  - so then $|\, R_{\boldsymbol{ee}}| = |\, R_{\boldsymbol{e'e'}}|$ is also minimized
  - → Minimizing sum (trace) here **is same as minimizing product (determinant).**

Matrix SNR?

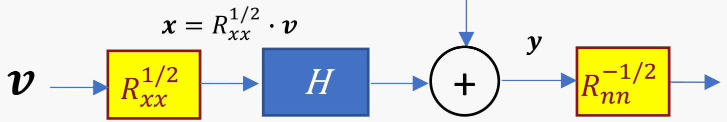
$\mathcal{E}_{\boldsymbol{x}} \rightarrow trace\,\{R_{\boldsymbol{xx}}\}$

$R_{\boldsymbol{xx}} = \mathbb{E}[\boldsymbol{xx}^*]$

AWGN $\boldsymbol{n} \sim [R_{\boldsymbol{nn}} = \mathbb{E}\,[\boldsymbol{nn}^*]]$

$\boldsymbol{SNR}_{out} = R_{\boldsymbol{yy}} \cdot R_{\boldsymbol{nn}}^{-1}$

$R_{\boldsymbol{vv}} = I$



$\boldsymbol{y}' = \widetilde{H} \cdot \boldsymbol{v} + \boldsymbol{n}'$

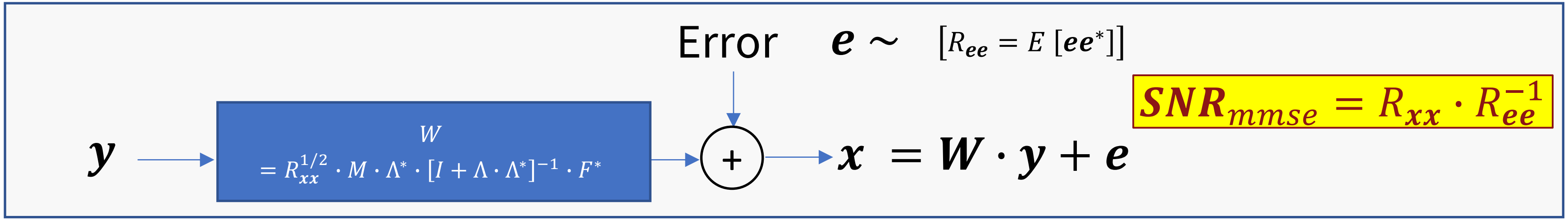
$R_{\boldsymbol{yy}} = H \cdot R_{\boldsymbol{xx}} \cdot H^* + R_{\boldsymbol{nn}}$

$$\widetilde{H} \triangleq R_{nn}^{-1/2} \cdot H \cdot R_{xx}^{1/2} = F \cdot \Lambda \cdot M^*$$

- It's like the parallel channels (take determinants) $SNR_{out} = |R_{\boldsymbol{yy}}| \cdot |R_{\boldsymbol{nn}}^{-1}| = \frac{|R_{\boldsymbol{yy}}|}{|R_{\boldsymbol{nn}}|}$.
  - Vector code from $\boldsymbol{v}$ to $\boldsymbol{y}'$.
  - Bit rate is then $b = \log_2(SNR_{out})$.

$$SNR_{out} = \frac{|R_{\boldsymbol{yy}}|}{|R_{\boldsymbol{nn}}|} = \frac{|H \cdot R_{\boldsymbol{xx}} \cdot H^* + R_{\boldsymbol{nn}}|}{|R_{\boldsymbol{nn}}|} = |\tilde{H} \cdot \tilde{H}^* + I| = |\Lambda^2 + I| = \prod_{n=1}^{N}(SNR_n + 1)$$

- This set depends on $R_{\boldsymbol{xx}}$ choice, likely with fixed $trace\,\{R_{\boldsymbol{xx}}\}$.
  - Water-fill $R_{\boldsymbol{xx}} = M \cdot diag\{\mathcal{E}_{water-fill}\} \cdot M^*$ maximizes the matrix SNR or effectively its determinant.



- How about "backward channel's," $\boldsymbol{x} = \boldsymbol{W} \cdot \boldsymbol{y} + \boldsymbol{e}$, (MMSE) $\boldsymbol{SNR}$?

$$SNR_{mmse} = \frac{|R_{\boldsymbol{xx}}|}{|R_{\boldsymbol{ee}}|} = |W \cdot R_{\boldsymbol{yy}} \cdot W^* + R_{\boldsymbol{ee}}| / |R_{\boldsymbol{ee}}| = |\Lambda^2 + I| = \prod_{n=1}^{N} (1 + SNR_n)$$

$$= SNR_{out}$$

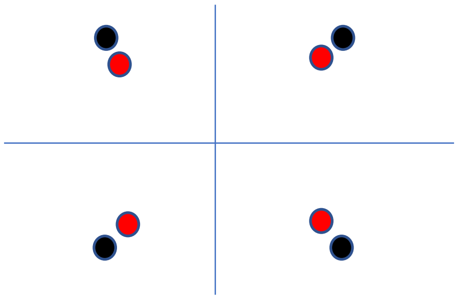
- Bit rate is again $b = \log_2(SNR_{mmse})$, $\mathbb{I}(\boldsymbol{x};\boldsymbol{y})$ is symmetric.
- $M^* \cdot W$ will estimate $\boldsymbol{v}$ (linearity of MMSE estimates).
- Optimizing determinants is same as optimizing MSE/traces.

Forward and backward have same SNR and "bit rate" (continuous x distribution, or Random AEP good-code sense.)

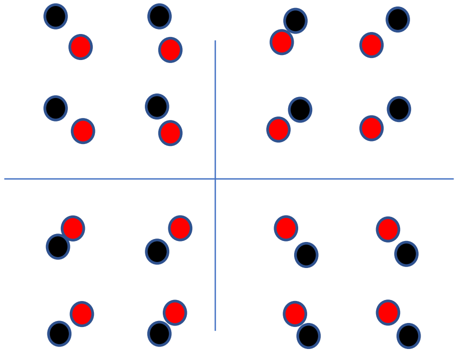
MMSE is always a Biased Estimate

- Biased-Estimate Definition: $\mathbb{E}\left[\hat{x}/_{x}\right] \neq \boldsymbol{x}$.

- MMSE estimates always have bias (if noise is nonzero), See Appendix D.2:
  - $\mathbb{E}\left[\hat{x}/_{x}\right] = \left(I - R_{ee} \cdot R_{xx}^{-1}\right) \cdot \boldsymbol{x} = \left(I - \boldsymbol{SNR}^{-1}\right) \cdot \boldsymbol{x}$



decision regions same



decision regions change

MMSE trades a little signal reduction for simultaneous noise reduction when minimizing the error, now on every dimension.

See PS2.1 (Prob 4.29)

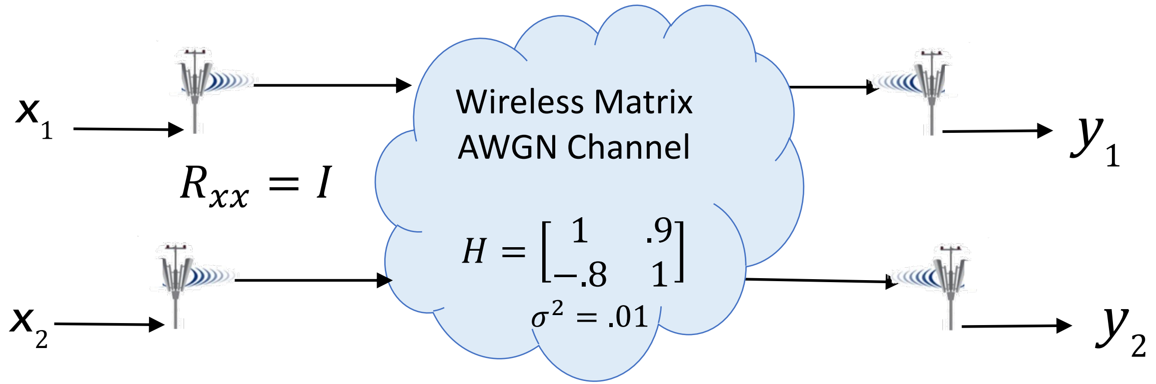
- For scalar case above, removal is scale up (by $\frac{SNR_{mmse}}{SNR_{mmse}-1}$).

- MIMO case, same per dimension, scale up (by $\frac{SNR_{mmse,n}}{SNR_{mmse,n}-1}$) **IF** MMSE R_{ee} is diagonal (vector coding).
  - IF not diagonal? (we'll learn what to do in later lectures.)

Linear Matrix MMSE Examples

[See PS2.2 (Prob 4.36)](#)

2 x 2 Antenna System




```
>> H=[1 .9
-.8 1];
>> Rxx=eye(2);
>> Rnn=.01*eye(2)
>> Ryy=H*Rxx*H'+Rnn;
>> Ryx=H;
>> W=(Ryx')*inv(Ryy) =
   0.5780  -0.5199
   0.4627   0.5780
>> W*H =
   0.9939   0.0003
   0.0003   0.9945
>> Ree=Rxx-W*Ryx =
   0.0061  -0.0003
  -0.0003   0.0055
>> snr=det(Rxx)/det(Ree);
>> log2(snr) =   14.8693
```

- Strong Crosstalk case from Chapter 1

Basically, the same L1:31 result, even without the "M" discrete modulator but why with no transmit M matrix?

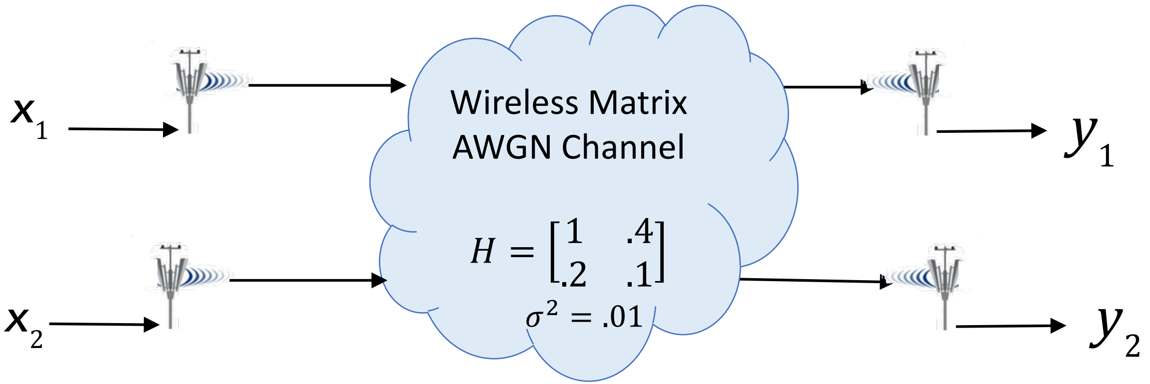
```
>> Mstar =

   0.4197   0.9076
   0.9076  -0.4197
```

$R_{xx} = I$ is close to water-fill (equal energy this channel);
$R_{xx} = M \cdot I \cdot M^*$; so "lucky" that its already close to best.

ML detector is only per-dimension independent if R_{xx} and R_{ee} are diagonal.

2 x 2 Antenna System



- This channel water-filled nonzero energy only on 1 dimension in L1.

Previously in L1, $b = \log_2(1 + 2 \cdot g_2) = 6.93$ bits/subsymbol
But this time, two dimensions are used, and the ML detectors are interdependent

```
>>H = 1.0000    0.4000
      0.2000    0.1000
>> Rxx=eye(2);
>> Ryy=H*Rxx*H'+Rnn;
>> Ryx=H;
>> W=(Ryx')*inv(Ryy) =
   0.9524   -0.4762
   0.0000    1.6667
>> W*H = 0.8571    0.3333
         0.3333    0.1667
>> Ree=Rxx-W*Ryx =
   0.1429   -0.3333
  -0.3333    0.8333
>> snr=det(Rxx)/det(Ree) =  126.0000
>> b=log2(snr)  =    6.9773  (only for VC)
```

not diagonal; ML detect is NOT parallel

```
>> SNR=inv(diag(diag(Ree))) =  7.0000       0
                                   0   1.2000

>> log2(diag(SNR)) =
   2.8074
   0.2630
>> sum(log2(diag(SNR))) =    3.0704
```

Thus, data rate loss can occur with independent detectors and MMSE.

(This loss can be recovered with Chapter 5's MMSE GDFE, in addition to using vector coding, so more than 1 solution)

See PS2.2 (Prob 4.36)

Time-Frequency Block $1 + .9 \cdot D^{-1}$

```
>> H=toeplitz([1 zeros(1,7)]',[1 .9 zeros(1,7)]);
>> Rxx=eye(9);
>> Rnn=.181*eye(8);
>> Ryy=H*Rxx*H'+Rnn;
>> Ryx=H;
>> W=(Ryx')*inv(Ryy);
>> P=W*H;
>> size(P) % =  9  9

>> Ree=Rxx-W*Ryx;
>> snr=det(Rxx)/det(Ree) = 2.4089e+07
>> SNR=inv(diag(diag(Ree)));
>> bn = 0.5*log2(diag(SNR))' =
 0.8769   1.0096   1.0691   1.0907   1.0902   1.0673   1.0054   0.8681   0.6085
>> sum(bn/9) =   0.9651
>> 10*log10(2^(2*ans)-1) =  4.4885 dB
```

Repeat for 8→32:

```
>> sum(bn/33)  =  1.0753

>> 10*log10(2^(2*ans)-1) =    5.3654 dB
```

Best infinite length is 5.7 dB.
(with dimension-by-dimension linear)

- See Chapter 3, 379A MMSE-LE example

Best with full ML is 8.8 dB, but requires input WF energy distribution .

Entropy and Estimation:
generalizing energy to all distributions

See PS2.3 (Prob 2.10)

Rest of L3 repeats EE379A early coding,
but here emphasizes the MMSE-canonical connection.

Information Measures and Energy/Mean-Square

Gaussian Distribution	Any Distribution
Mean-square energy $\mathcal{E}_x = \mathbb{E}[\|\boldsymbol{x}\|^2]$	Entropy $\mathcal{H}_{\boldsymbol{x}}$ *Section 2.3.1*
Mean-square error $\sigma_e^2 = \mathbb{E}[\|\boldsymbol{e}\|^2]$	Conditional Entropy $\mathcal{H}_{\boldsymbol{x}/\boldsymbol{y}}$ *Section 2.3.2*
Signal-to-Noise $\mathcal{E}_x \,/\, \sigma_e^2$	Mutual Information $\mathbb{I}(\boldsymbol{x};\boldsymbol{y}) = \mathcal{H}_{\boldsymbol{x}}$-$\mathcal{H}_{\boldsymbol{x}/\boldsymbol{y}}$ *Section 2.3.2*

The information-carried by random variable/process generalizes the energy concepts from MMSE/Gaussian analysis to a general distribution.

These information measures correspond to bits/symbol quantities, and for the Gaussian case are basically the $\log_2$ of the corresponding energy measure.

Entropy – measure information (source)

- **Entropy:** $\mathcal{H}_{\tilde{x}} = \mathbb{E}\left[\log_2\left(\frac{1}{p_{\tilde{x}}}\right)\right] = \sum_{i=0}^{|C|-1} p_{\tilde{x}}(i) \cdot \log_2\left(\frac{1}{p_{\tilde{x}}(i)}\right)$ Discrete $p_{\tilde{x}}(i)$

- Measures a distribution's many values, its ***information***, by probability (think subsymbols).

- Generalizes bits/subsymbol, especially when the constellation size $|C| \geq M^{1/\bar{N}} = 2^{\tilde{b}}$.

example: $p_{\tilde{x}}(i) = \frac{1}{M}$ (uniform) → $|C| = 2^{\tilde{b}}$

Uniform → $\mathcal{H}_{\tilde{x}} = \log_2\left(M^{1/\bar{N}}\right) = \tilde{b}$ $\left(|C| = 2^{\tilde{b}+\tilde{\rho}}\right)$ $\tilde{\rho} = 0$; uncoded)

- Uniform distribution has maximum entropy

$\mathcal{H}_{\tilde{x}} \leq \log_2|C|$ Binary example: $p_{\tilde{x}}(0) = \frac{1}{128}$ and $p_{\tilde{x}}(1) = \frac{127}{128}$

$$\mathcal{H}_{\tilde{x}} = \frac{\log_2(128)}{128} + \frac{127}{128} \cdot \log_2\left(\frac{128}{127}\right) = .06 < 1$$

See PS2.3 (Prob 2.10)

Information left after given another random vector

- Conditional entropy $\mathcal{H}_{\tilde{x}/\tilde{y}} = \mathbb{E}\left[\log_2\left(\frac{1}{p_{\tilde{x}/\tilde{y}}}\right)\right] = \sum_{j=0}^{|Y|-1}\sum_{i=0}^{|C|-1} p_{\tilde{x}\tilde{y}}(i,j)\cdot\log_2\left(\frac{1}{p_{\tilde{x}/\tilde{y}}(i,j)}\right)$

$$\mathcal{H}_{\tilde{x}/\tilde{y}} = \mathcal{H}_{\tilde{x}\tilde{y}} - \mathcal{H}_{\tilde{y}}$$

- Measures $\tilde{x}$'s residual randomness/info when $\tilde{y}$ is known/given

$\tilde{x}\,;\tilde{y}$	0	1	$p_{\tilde{x}}$
0	3/8	1/8	1/2
1	1/8	3/8	1/2
$p_{\tilde{y}}$	1/2	1/2	

$$\mathcal{H}_{\tilde{x}\tilde{y}} = \frac{6}{8}\cdot\log_2\frac{8}{3} + \frac{2}{8}\cdot\log_2 8 = 1.811$$

$$\mathcal{H}_{\tilde{x}} = 1 = \mathcal{H}_{\tilde{y}}$$

$$\mathcal{H}_{\tilde{x}/\tilde{y}} = 1.811 - 1 = .811\ bits/subsymbol$$

- If x and y are independent, then $\mathcal{H}_{\tilde{x}/\tilde{y}} = \mathcal{H}_{\tilde{x}}$

Continuous Distribution – DIFFERENTIAL Entropy

- Differential Entropy $\mathcal{H}_{\tilde{x}} = \mathbb{E}\left[\log_2\left(\frac{1}{p_{\tilde{x}}}\right)\right] = -\int_{-\infty}^{\infty} p_{\tilde{x}}(u) \cdot \log_2\left(\frac{1}{p_{\tilde{x}}(u)}\right) \cdot du$

- Differential Entropy $\mathcal{H}_{\tilde{x}}$ is not same as an integral-to-sum via a discrete approximation of $p_{\tilde{x}}(u)$.
  - They differ by a constant that depends on the approximation-interval size.
- Differential Entropy $\mathcal{H}_{\tilde{x}}$ does still however measure information content when subsymbols in codewords are chosen (usually at random) from $p_{\tilde{x}}(u)$.

- Maximum $\mathcal{H}_{\tilde{x}}$ occurs when $p_{\tilde{x}}(u)$ is **Gaussian** (any mean), with constant average energy.

$$\int_{-\infty}^{\infty} p_{\tilde{x}}(u) \cdot \|u\|^2 \cdot du = \mathcal{E}_{\tilde{x}}$$

Complex $\mathcal{H}_{\tilde{x}} = \log_2(\pi e \mathcal{E}_{\tilde{x}})$ bits/clpx-subsymbol

Real $\mathcal{H}_{x} = \frac{1}{2}\log_2(2\pi e \bar{\mathcal{E}}_{x})$ bits/dimension

- More generally, $\text{trace}\{R_{\tilde{x}\tilde{x}}\} = \mathcal{E}_{\tilde{x}}$.

$\mathcal{H}_{\tilde{x}} = \log_2|\pi e R_{\tilde{x}\tilde{x}}|$ bits/cplx-subsymbol

Gaussian MMSE & conditional entropy

- Complex scalar Gaussian x $\quad p_x(u) = \frac{1}{\pi\sigma_x^2} e^{-\frac{|x|^2}{\sigma_x^2}}$ $\quad \mathcal{H}_x = \log_2\{\pi \cdot e \cdot \sigma_x^2\}$

- Conditional x/y ? $\quad \mathcal{H}_{x/y} = \log_2\left\{\pi \cdot e \cdot \sigma_{x/y}^2\right\}$ $\quad \sigma_{x/y}^2 = \sigma_x^2 - {r_{xy}^2}/{\sigma_y^2}$ = MMSE

- Vector $\boldsymbol{x}$? $\quad \mathcal{H}_{\boldsymbol{x}} = \log_2\left\{(\pi e)^{\bar{N}} \cdot |R_{\boldsymbol{xx}}|\right\}$ $\quad R_{\boldsymbol{x/y}}^{\perp} = R_{\boldsymbol{xx}}^2 - R_{\boldsymbol{x/y}} \cdot R_{\boldsymbol{yy}}^{-1} \cdot R_{\boldsymbol{x/y}}$ = MMSE

(Appendix D on MMSE)

$$\mathcal{H}_{\boldsymbol{x/y}} = \log_2\left\{(\pi e)^{\bar{N}} \cdot \left|R_{\boldsymbol{x/y}}^{\perp}\right|\right\}$$

$\bar{N}$ is the number of complex dimensions = $N/2$

See PS2.5 (Prob 2.20)

Relation to MMSE Estimation

- If $\tilde{\boldsymbol{x}}$ and $\tilde{\boldsymbol{y}}$ are jointly Gaussian, then $p_{\tilde{\mathbf{x}}/\tilde{\mathbf{y}}}$ is also Gaussian and has mean as MMSE estimate $\mathbb{E}[\tilde{\boldsymbol{x}}/\tilde{\boldsymbol{y}}]$ and autocorrelation $R_{\boldsymbol{ee}} = R_{\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}} - R_{\tilde{\boldsymbol{x}}\tilde{\boldsymbol{y}}} \cdot R_{\tilde{\boldsymbol{y}}\tilde{\boldsymbol{y}}}^{-1} \cdot R_{\tilde{\boldsymbol{y}}\tilde{\boldsymbol{x}}}$.

- $\mathcal{H}_{\tilde{\mathrm{x}}/\tilde{\mathbf{y}}} = \log_{\mathbf{2}}|\pi e R_{\boldsymbol{ee}}|$ - that is, the entropy is essentially just the log of the MMSE (Gaussian).

  - Entropy generalizes MMSE to any probability distribution.
  - Measures the information content of the "miss" in estimating $\tilde{\boldsymbol{x}}$ from $\tilde{\boldsymbol{y}}$ for any $p_{\tilde{\boldsymbol{x}}\tilde{\boldsymbol{y}}}$.

See PS2.5 (Prob 2.20)

Mutual Information and SNR

Subsection 2.3.2

See PS2.5 (Prob 2.20)

For Gaussian, $\mathcal{I}$ and (geo) SNR are in 1-to-1 relationship
MMSE and best rate are essentially same thing.

Mutual Information ~ SNR

- Mutual Information is:

$$\mathbb{I}(\tilde{x};\tilde{y})=\mathbb{E}\left[\log_2\left(\frac{p_{\tilde{x}\tilde{y}}}{p_{\tilde{x}}\cdot p_{\tilde{y}}}\right)\right] = \mathcal{H}_{\tilde{x}} - \mathcal{H}_{\tilde{x}/\tilde{y}} = \mathcal{H}_{\tilde{y}} - \mathcal{H}_{\tilde{y}/\tilde{x}}.$$

$$= \mathcal{H}_{\tilde{x}} - \mathcal{H}_{\tilde{x}/\tilde{y}} = \mathcal{H}_{\tilde{y}} - \mathcal{H}_{\tilde{y}/\tilde{x}}.$$

  - For discrete example $\mathbb{I}$ = 1-.811 = .189 bits/subsymbol.

- $\mathbb{I}(\tilde{x};\tilde{y})$ is symmetric in $\tilde{x}$ and $\tilde{y}$ (MMSE forward and backward channel).

- $\mathbb{I}(\tilde{x};\tilde{y})$ measures common ("mutual") information between $\tilde{x}$ and $\tilde{y}$, $\mathbb{E}\left[\log_2\left(\frac{p_{\tilde{x}/\tilde{y}}}{p_{\tilde{y}}}\right)\right]$.
  - On average, $\mathbb{I}(\tilde{x};\tilde{y})$ measures how much bigger is unconditional info versus conditional info, in bits.

- $\mathbb{I}(\tilde{x};\tilde{y}) = \log_2\frac{|R_{\tilde{x}\tilde{x}}|}{|R_{ee}|} = \log_2\frac{|R_{\tilde{y}\tilde{y}}|}{|R_{nn}|} = \log_2\left(\left(1+SNR_{geo}\right)^{\bar{N}}\right)$ for the matrix AWGN.

- OR as earlier for vector coding $\mathbb{I}(\tilde{x};\tilde{y}) = \sum_{n=1}^{\bar{N}}\log_2 SNR_{mmse,n}$ for the matrix AWGN.

Law of Large Numbers, repeat 379A

Theorem 2.1.1 (Law of Large Numbers (LLN)) *The LLN observes that a stationary random variable z's sample average over its observations $\{z_n\}_{n=1,...,N}$ converges to its mean with large N such that*

$$\lim_{N\to\infty} \quad Pr\left\{\left|\left(\frac{1}{N}\sum_{n=1}^{N} z_n\right) - \mathbb{E}[z]\right| > \epsilon\right\} \to 0 \;\; \textit{weak form} \qquad (2.13)$$

$$\lim_{N\to\infty} \quad Pr\left\{\frac{1}{N}\sum_{n=1}^{N} z_n = \mathbb{E}[z]\right\} = 1 \;\; \textit{strong form} \; . \qquad (2.14)$$

- Distribution of z must be the same (stationary) for all random selections.
- The random z can be function of random variable ($z = f(x)$)and the sample mean converges to $\mathbb{E}[f(x)]$.
  - E.g., $z_n = \|\boldsymbol{x}_n\|^2$ where the vector $\boldsymbol{x}_n$ might also have (a growing) N components (energy sample or length of the vector).
  - LLN then states that all the energy (really points in selection from any distribution with$\mathbb{E}[\|\boldsymbol{x}\|^2] \le \mathcal{E}_x$) of a hypersphere are are at its surface with probability 1. Points on the interior have probability zero. It is also a sum of independent terms, and thus Gaussian (central limit theorem).
  - The marginal distributions for the vector $\boldsymbol{x}_n$'s element selections, and thus for $\boldsymbol{x}_n$ also, would be Gaussian if this $N\to\infty$-sequence has max entropy (uniform).
- The function of most interest in coding is $f(x)$ = $-\log_2[p_x(x)]$- that is the function itself is probability distribution's log.
  - The **sample average** of this function **converges** to the **entropy**.
  - This suggests choosing codewords (this means each subsymbol in the codeword) at random from stationary distribution,
    - and then repeat at higher level for several codes chosen at random.
  - These are discrete codes, even when $\boldsymbol{x}$ is continuous, but their average follows the entropy (and mutual information).
  - Generalizes **sphere-packing** (which was for the AWGN only).

Random coding generalizes 379A sphere packing

- Pick subsymbols $\boldsymbol{x}_n$ randomly (independently) from (stationary) distribution $p_{\tilde{\boldsymbol{x}}}$ for each of $M = 2^b$ c'words.
  - This is one **random code**.
- Repeat the exercise for another code, and …. many more.
- Compute the average performance of all these random selected codes:
  - As $\bar{N} \to \infty$, this average performance is outstanding (as we'll see), as long as $\tilde{b} < \mathbb{I}(\tilde{\boldsymbol{x}};\tilde{\boldsymbol{y}})$.
  - So at least one good one must exist.
- Entropy per subsymbol is

$$\tilde{\mathcal{H}}_{\boldsymbol{x}} = \frac{-1}{\bar{N}} \cdot E\left[\log_2(p_{\boldsymbol{x}})\right] = \frac{-1}{\bar{N}} \sum_{n=1}^{\bar{N}} E\left[\log_2(p_{\tilde{\boldsymbol{x}}_n})\right] \quad ,$$

- LLN with function $\log_2\left[p_{\tilde{z}}^{-1}\right]$ is the sample-average entropy estimate.

$$\hat{\tilde{\mathcal{H}}}_{\tilde{\boldsymbol{x}}} = \frac{-1}{\bar{N}} \cdot \sum_{n=1}^{\bar{N}} \log_2\left[p(\tilde{\boldsymbol{x}}_n)\right] = \frac{-1}{\bar{N}} \cdot \log_2\left[p(\boldsymbol{x})\right] \quad .$$, which converges to (constant) $\tilde{\mathcal{H}}_{\boldsymbol{x}}$

- The constant means the ave code has uniform distribution of codewords (asymptotically), $2^{\bar{N}\cdot\tilde{\mathcal{H}}_{\boldsymbol{x}}}$ of them.

Asymptotic Equal Partition (AEP)

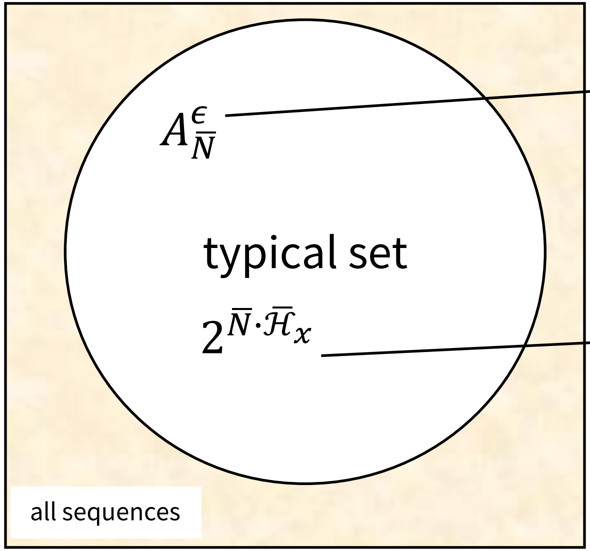
AEP Typical Sets ~ spheres/ball packing

- The set is $$A^{\epsilon}_{\overline{N}}(\boldsymbol{x}) \triangleq \left\{ \boldsymbol{x} = [\tilde{\boldsymbol{x}}_1, \tilde{\boldsymbol{x}}_2, ..., \tilde{\boldsymbol{x}}_{\overline{N}}] \,\middle|\, 2^{-\overline{N}\cdot\mathcal{H}_{\tilde{\boldsymbol{x}}}-\epsilon} \leq p(\tilde{\boldsymbol{x}}_1, \tilde{\boldsymbol{x}}_2, ..., \tilde{\boldsymbol{x}}_{\overline{N}}) \leq 2^{-\overline{N}\cdot\mathcal{H}_{\tilde{\boldsymbol{x}}}+\epsilon} \right\}$$

Lemma 2.3.6 **[AEP Lemma]** *For a typical set with $\overline{N} \to \infty$, the following are true:*

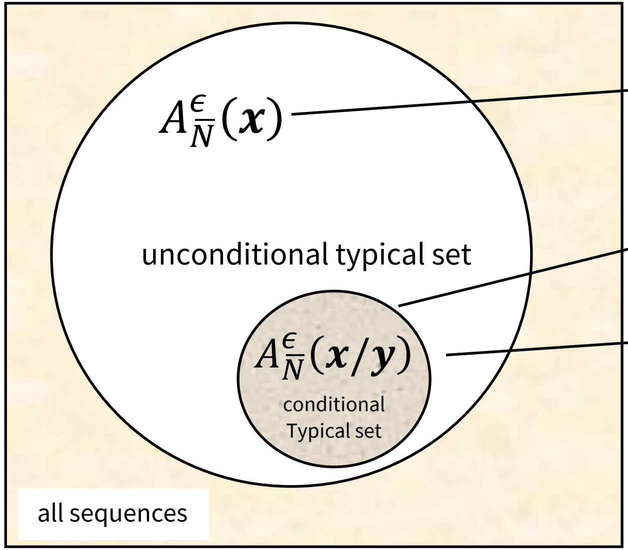
- $Pr\{A^{\epsilon}_{\overline{N}}(\boldsymbol{x})\} \to 1$
- *for any codeword* $\boldsymbol{x} \in A^{\epsilon}_{\overline{N}}$, $Pr\{\boldsymbol{x}\} \to 2^{-\overline{N}\cdot\mathcal{H}_{\tilde{\boldsymbol{x}}}}$

Decoder works well if only one codeword in conditional set for each $\boldsymbol{y}$ value, so good code spreads them uniformly.



Prob $\to 1$

number of sequences



$2^{\overline{N}\cdot H_{\tilde{\boldsymbol{x}}}}$

redundancy $\tilde{\rho} = H_{\tilde{\boldsymbol{x}}/\boldsymbol{y}}$ bits

$2^{\overline{N}\cdot H_{\tilde{\boldsymbol{x}}/\tilde{\boldsymbol{y}}}}$

$\mathcal{H}_{\tilde{\boldsymbol{x}}} = \mathcal{I}(\tilde{\boldsymbol{x}};\tilde{\boldsymbol{y}}) + \mathcal{H}_{\tilde{\boldsymbol{x}}/\tilde{\boldsymbol{y}}}$ bits/subsymbol $\leq \tilde{b} + \tilde{\rho}$

So pick $\tilde{\rho}$ large enough.

There are $2^{N\cdot H_{\tilde{\boldsymbol{x}}}} \cdot 2^{-N\cdot H_{\tilde{\boldsymbol{x}}/\tilde{\boldsymbol{y}}}} = 2^{N\cdot\mathcal{I}(\tilde{\boldsymbol{x}};\tilde{\boldsymbol{y}})}$ little sets
In the big set if "equally partitioned"

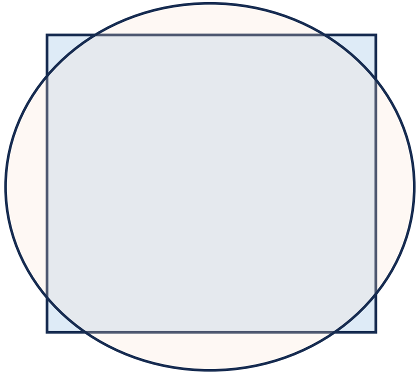
General Capacity Theorem

$$\frac{|A_N^\epsilon(\boldsymbol{x})|}{|A_N^\epsilon(\boldsymbol{x}/\boldsymbol{y})|} \to 2^{\mathcal{I}(\boldsymbol{x};\boldsymbol{y})} \qquad \text{since } \mathcal{I}(\boldsymbol{x};\boldsymbol{y}) = \mathcal{H}_{\boldsymbol{x}} - \mathcal{H}_{\boldsymbol{x}/\boldsymbol{y}}$$

- Good codes will have only 1 codeword per conditional entropy subset.
- MAP detector decision region is then ~ $A_N^\epsilon(\boldsymbol{x}/\boldsymbol{y})$ - on average; but we can find it for one good code.
- If $A_N^\epsilon(\boldsymbol{x})$ were any larger, all codes (good or bad) will have at least one $A_N^\epsilon(\boldsymbol{x}/\boldsymbol{y})$ that contains 2+ codewords, which mean the MAP has to "flip a coin" – not good (high error prob).
- SHANNON's CAPACITY THEOREM
  - Number of codewords is limited by mutual info $b \le \mathcal{I}(\boldsymbol{x};\boldsymbol{y})$.
  - Which is per-subsymbol equivalent with random code $\tilde{b} \le \mathcal{I}(\tilde{\boldsymbol{x}};\tilde{\boldsymbol{y}})$.
  - If maximized over input distributions $\tilde{b} < \tilde{C} \le \max_{p_{\tilde{\boldsymbol{x}}}} \mathcal{I}(\tilde{\boldsymbol{x}};\tilde{\boldsymbol{y}}) \frac{\text{bits}}{\text{subsymbol}}$.

The uniform part is most important.

- The Gaussian distribution corresponds to marginal of uniform distribution over a hypersphere.
  - This uniform distributions marginals are asymptotically Gaussian.
  - This is a special case where uniform and Gaussian are basically the same.
  - Because all the Gaussian infinite-length vectors (codewords) have same energy (zero variance of the energy).
- All the points (really volume) are (is) at the surface.
- The Gaussian marginal dist'n is important only for shaping gain (< 1.53 dB).
- The (AEP) uniform spacing of points (no matter where the majority of them sit, surface or otherwise) remains for the fundamental gain.



The uniform spacing separates codewords in the union of the hypersquare (orthotope) and hypersphere.

Thus, good codes can be based on sequences from uniformly spaced PAM/QAM subsymbols.

And the rest is MMSE Estimation,

With a chain-rule twist in Some situations

Vector Coding is always all MAP, All ML, all MMSE special case.

End Lecture 3