Spring 2005: EECS126 Practice Problems for Midterm 2

No Collaboration Permitted

Be as clear and precise in your answers as possible

Come to the front if you have a question.

Problem 2.1 *(30pts) True or False. Prove or show a counterexample:*

a. *10pts. Let X and Y be two rv's. Then:*

$var(\frac{X+Y}{2}) \leq \max\{var(X), var(Y)\}$

b. *10pts Let $X_1, X_2, ..., X_n$ all be identically distributed as Gaussian $N(0,1)$*

Then $Y = \frac{\sum_{i=1}^{n} X_i}{\sqrt{(n)}}$ is also Gaussian $N(0,1)$

c. *10pts For every continuous r.v. X that takes strictly positive values there exists some $s > 0$ for which the moment generating function $M_x(s)$ exists and is well defined.*

Problem 2.2 *(40pts) Linear Least Square Estimation.*

a. *20pts. Let X, Y, Z_1, Z_2 be independent rv.'s with known means and variances.*

Given

$$U = X - Y + Z_1$$

$$V = X + Y + Z_2$$

find $LLSE(X|U,V)$ and $LLSE(Y|U,V)$, as well as the resulting mean-squared errors of the estimates.

b. *20pts. Let X be a Bernoulli(p) and N_0, N_1, N_2 be independent Gaussian rv's N(0, σ_0^2), N(0, σ_1^2),N(0, σ_2^2) respectively.*

Given two observations:

$$Y_1 = X + N_0 + N_1$$

$$Y_2 = X + N_0 + N_2$$

find $LLSE(X|Y_1, Y_2)$ as well as the resulting mean-squared error of the estimate.

(**This last problem is far (about a factor of 3) longer than what will appear on the midterm. It is here to give you extra practice as well as possibly be a lot of fun. If you like this sort of thing, think about taking 174 and/or 229 to learn more.**)

Problem 2.3 *(55pts) Random error-correcting codes: This problem explores a particular kind of error-correcting code that is random in nature and is used over an "erasure" or packet-drop channel. This is an idealization of what happens over the Internet or a wireless network in which packets can be lost, but the receiver knows when this loss happens.*

We assume that each packet X_i carries a single bit (either $+1$ or -1) and is received as $Y_i = X_i * Z_i$ where the Z_i are independent and identically distributed Bernoulli $(1-\delta)$ taking on values $Z_i = 1$ (packet makes it through) or $Z_i = 0$ (packet is dropped). δ therefore represents the probability of dropping a packet.

a. *5pts Suppose that in advance, we randomly generate two independent "codewords" $\tilde{X}_1^n$ and $\check{X}_1^n$ each of length n with all $2n$ constituent symbols chosen independently using $P(+1) = P(-1) = \frac{1}{2}$.*

   Calculate the probability that the two generated codewords happen to be exactly the same in each of the n positions simultaneously? (ie. $P(\tilde{X}_1^n = \check{X}_1^n)$)

b. *5pts Now suppose that the codeword $\tilde{X}_1^n$ is sent across the channel and the receiver has received Y_1^n. The received sequence Y_1^n is considered "compatible" with a codeword $\hat{X}_1^n$ if for every $1 \leq i \leq n$, either $Y_i = \hat{X}_i$ or $Y_i = 0$ (that bit was dropped).*

   Show that the transmitted codeword $\tilde{X}_1^n$ is always compatible with Y_1^n.

c. *5pts Calculate the probability that the received sequence Y_1^n will be compatible with a wrong codeword $\check{X}_1^n$, where the $\check{X}_i$ are all independent of each other and the $\tilde{X}_i$. (ie. What is $P(\forall 1 \leq i \leq n, Y_i = \check{X}_i * Z_i)$?)*

d. *5pts Next, we suppose that instead of having only one competing independent codeword $\check{X}_1^n$, there are actually $M-1$ such competing independent codewords. We say that an "error" occurs if two or more codewords (including the true one $\tilde{X}_1^n$) are compatible with the received sequence Y_1^n. Show that the probability of error P_e is less than or equal to M times the probability that you calculated in part [c].*

e. *10pts Build on parts [c] and [d] to give an explicit upper bound on P_e as a function of M, δ, n.*

   Give a formula for how large M can be as a function of δ, n, ϵ while still guaranteeing that $P_e < \epsilon < 1$. [Use logs where it makes sense.].

f. *5pts The formulas calculated in part [e] are actually too conservative, so we will take a different approach. Let L be the random variable counting the number of zeros in the Z_1^n sequence. Suppose that we knew that there were exactly $L = l$ zeros in the Z_1^n sequence.* Conditioned on this event, *revise your calculation from part [e] bounding P_e.*

g. *10pts Use Chernoff bounding techniques to get a bound on the probability of having $L \geq n(\delta + \beta)$ for $\beta > 0$.*

h. 10pts *Combine parts [f] and [g] to give new bounds on P_e as a function of M, δ, n.*

Compare to what you had calculated in [e].

i. Extra Bonus *Suppose that the message set consisted of nR bits (and hence 2^{nR} distinct possible messages) denoted B_k for $k = 1, 2, \ldots nR$ for $0 \leq R \leq 1$. How high can you make R while still getting a low P_e as n gets large using the bounds of [e]? Using the bounds of [h]? Do you think any higher rates could be achievable with other communication schemes?*

j. Extra Bonus *Generating 2^{nR} codewords would require $n2^{nR}$ fair coin tosses — too big to even store for all but small values of n. Show that the following scheme using only $n + n^2R$ fair coin tosses still results in pairs of independent codewords for any pair of distinct bit strings B_1^{nR}.*

$$X_i(B_1^{nR}) = G_{0,i} + \sum_{k=1}^{nR} G_{k,i} B_k \bmod 2$$

where the $G_{j,i}$ are all independent fair coin tosses for all $j = 0, 1, \ldots, nR$ and $i = 1, 2, \ldots, n$.

Everything you did in parts [a-i] only used this sort of pairwise independence.

k. Super Bonus *Given the $G_{j,i}$, what is the computational complexity of encoding a given message string into a codeword?*

Given the packet-drop model here, what is the computational complexity of decoding? [Think about solving the right system of linear equations.]