(a) \[ H(x) = - \sum_{x_k \in x} p_x(x_k) \log_2 p_x(x_k) \]
\[ = -0.4 \log_2 0.4 - 0.2 \log_2 0.3 - 0.3 \log_2 0.3 \]
\[ = 1.57 \quad \text{bits/information bit} \]

\[ H(x) \leq R_{opt} \leq H(x) + \frac{1}{N} \]
\[ \Rightarrow \quad 1.57 \leq R_{opt} \leq 1.57 + \frac{1}{N} \]

(b)

\[
\begin{array}{cccccccc}
& & & & & & 0.10 & & \\
& & & & & & 0.20 & & \\
& & & & & & 0.30 & & \\
& & & 0.34 & & 0.50 & & 0 & & \\
& & & 0 & & 0.50 & & 1.0 & & \\
& & & 0 & & 0.20 & & 1.0 & & \\
& & & 0 & & 0.10 & & 0.10 & & \\
& & & 0 & & 0 & & 0.10 & & \\
& & & 0 & & 0 & & 0 & & \\
\end{array}
\]

\[ R = \frac{3 \times 0.82 + 4 \times 0.18}{2} = 1.59 \quad \text{bits/information bit} \]

\[ 1.57 \leq 1.59 \leq 2.07 \quad \checkmark \]

(c) For \( N = 2 \)

\[ \frac{H(x, y) + H(x, y)}{2} \leq R \leq \frac{H(x) + H(x, y) + 1}{2} \]

\[ 1.57 + H(x, y) \leq 1.8 \leq 1.57 + H(x, y) + 1 \]

\[ \Rightarrow \quad 0.77 \leq H(x, y) \leq 0.73 \]

\[ \Rightarrow \quad 0 \leq H(x, y) \leq 0.23 \]

And we know:

\[ H(x_1, x_2, \ldots, x_n) \leq H_{opt} \leq H(x_1, x_2, \ldots, x_n) + 1 \]

\[ \Rightarrow \quad 0 \leq N \cdot H_{opt} \leq H(x) + H(x_2 | x_1) + H(x_3 | x_1, x_2) + \ldots + H(x_n | x_1, x_2, \ldots, x_{n-1}) \]

\[ \leq H(x) + (N-1) \cdot H(x_2 | x_1) + 1 \]

\[ \leq 1.57 + 0.23 (N-1) + 1 \]

\[ \Rightarrow \quad 0 \leq R \leq \frac{1.57 + 0.23 (N-1)}{N} + \frac{1}{N} \]
2) (a) First, select \( \varepsilon = 0.01 \).

Now, choose \( N \) large enough such that
\[
P(A_{0.01}^{(N)}) \geq 1 - 10^{-10}
\]

This gives us our typical set. Then encode/decode as in class: encode only the typical set and map everybody else to a single sequence.

(b) \( H(x) = - \sum_{x \in X} p_x(x) \log p_x(x) = 2.023 \) bits/observation

Need to find the size of \( B^{(N)}_\varepsilon \).

Use ideas from class:
\[
1 \geq \sum_{x \in B^{(N)}_\varepsilon} p_x(x)
\]
\[
\geq \sum_{x \in B^{(N)}_\varepsilon} 2^{-H(x) + \varepsilon} / c
\]
\[
= |B^{(N)}_\varepsilon| \cdot 2^{-H(x) + \varepsilon} / c
\]
\[
\geq c \cdot 2^{-N(H(x) + \varepsilon)}
\]
\[
\Rightarrow R \leq \frac{1}{N} \log_2 \left| B^{(N)}_\varepsilon \right| \leq \frac{1}{N} \left( N (H(x) + \varepsilon) + \log_2 c \right)
\]
\[
= H(x) + \varepsilon + \frac{\log_2 c}{N}
\]

Therefore, for large \( N \), no effect on rate.

(c) codeword length \( \frac{1}{N} \log c \) as \( N \rightarrow \infty \)

\[
R = H(x) + \varepsilon + \frac{1}{N} \log_2 (N(H(x) + c))
\]
\[
\Rightarrow \quad R = H(x) + \varepsilon + \frac{\log_2 (N(H(x) + c))}{N}
\]

Since the last term \( \rightarrow 0 \) as \( N \rightarrow \infty \), no impact on achievable rate, but he will need to choose \( c \) (slightly) smaller \( \varepsilon \) or \( \varepsilon / 2 \).
3) (a) 
\[ H(X) = \sum_{x_k \in X} p_X(x_k) \log_2 p_X(x_k) = \log_2 3 = 1.58 \text{ bits/symbol} \]
\[ = 15.800 \text{ bits/s} \]

\[ C_{155c} = 1 - H(X) \]
\[ = 1 + 0.9 \log_2 0.9 + 0.1 \log_2 0.1 = 0.531 \text{ bits/symbol} \]
\[ = 21.240 \text{ bits/s} \]

Yes! Choose R somewhere in between and then follow constructions from class.

(b) No. For any channel output sequence, there is some (small) probability it came from any input; hence, \( P_e > 0 \) for any rate.

(c) \[ I(X;Y) = H(Y) - H(Y|X) \]
\[ \leq \log_2 4 - 1 = 1 \]
and I can achieve by using:
\[ p_X(0) = \frac{1}{2}, \quad p_X(1) = 0, \quad p_X(2) = \frac{1}{2}, \quad p_X(3) = 0 \]

and, note that if I use these input distributions, I can get 1 bit/channel use with no errors.
\( h(z_1, z_2) = h(z_1) + h(z_2) \)

\[ = \frac{1}{2} \log_2 \left( \frac{2 \pi e}{\sigma_1^2} \right) + \frac{1}{2} \log_2 \left( \frac{2 \pi e}{\sigma_2^2} \right) \]

\[ = \frac{1}{2} \log_2 \left( \frac{4 \pi e^2}{(1 - \rho^2)} \right) \]

\( I(x, y|x) = h(y) - h(y|x) \)

\[ = h(y) - h(z) \]

\[ = h(y) - \frac{1}{2} \log_2 \left( \frac{4 \pi e^2}{(1 - \rho^2)} \right) \]

Now let's calculate statistics of \( y \)

\[ y_1 = x + z_1 \]
\[ y_2 = x + z_2 \]

\[ E[y_1^2] = E[(x + z_1)^2] = E[x^2] + 2E[xz_1] + E[z_1^2 \rho^2] \leq \sigma + 1 \]

\[ E[y_2^2] = E[(x + z_2)^2] \leq \sigma + 1 \]

\[ E[y_1 y_2] = E[(x + z_1)(x + z_2)] = E[x^2] + E[xz_1] + E[z_1 z_2] + E[z_2^2 \rho^2] \leq \sigma + 1 \]

\[ K_y = \begin{bmatrix} \sigma + 1 & \rho \sigma^2 \\ \rho \sigma^2 & \sigma + 1 \end{bmatrix} \]

This is fixed for any distribution. Thus, \( y \)

Gaussian maximizes \( h(y) \), hence choose \( x \sim N(0, \sigma) \)

and

\[ C = \frac{1}{2} \log_2 \left( \frac{4 \pi e^2}{\rho \sigma \rho + 1} \right) - \frac{1}{2} \log_2 \left( \frac{4 \pi e^2}{(1 - \rho^2)} \right) \]

\[ = \frac{1}{2} \log_2 \left( \frac{2 \rho + 1 - 2 \rho \rho^2}{1 - \rho^2} \right) = \frac{1}{2} \log_2 \left( 1 + \frac{\rho^2}{1 + \rho} \right) \]

(c) Standard Gaussian: \( C = \frac{1}{2} \log_2 (1 + \rho) \)
Idea: Encode a lot of information bits in each symbol and repeat it until you are sure it got across.

Solution:

Need $P_e < \delta$. So choose $N$ large enough such that probability that two go through the "good" channel is high enough.

$$2^{-\alpha N} < \delta$$

is sufficient. One symbol goes through "good" channel in first $\alpha N / 2$ tries with probability $1 - \delta / 2$ and same in second $\alpha N / 2$. Decoder: look over block of length $N$ for $\alpha N / 2$ that match with probability $1$, only match comes from those symbols that went through "good" channel.

Encoder: Need $R$ bits/symbol or $NR$ bits per codeword. Thus, choose $X_i$ between 0 and $\sqrt{P}$ (guaranteeing $E[X_i^2] \leq P$) as:

$$X = 0, 0.00...0, b_1, b_2, b_3,..., b_{NR}$$

enough to always make $x_i \leq \sqrt{P}$

and repeat the symbol $N$ times on the channel:

$$\overline{X, X, X, ..., X}$$