بسم الله الرحمن الرحیم

پروژه کامپیوتری درس آمار و احتمال مهندسی

فاز دوم

دکتر میرمحسنی

مهرسا پوریا – 95101247

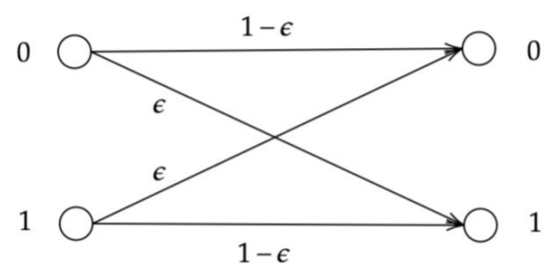
بهار 1397

مقدمه

در این پروژه قصد داریم با کاربرد های آمار و احتمال در سیستم های مخابراتی آشنا شویم. در فاز اول با تولید متغیر های تصادفی برنولی ، نمایی ، رایلی ، و گاوسی آشنا شدید و در بخش دوم می خواهیم به کمک متغیر های تصادفی تولید شده در فاز اول، مدل ساده شده ی سیستم های مخابراتی (دیجیتال) را شبیه سازی می کنیم.

1.کانال باینری متقارن (BSC)

در این قسمت سعی می کنیم که یکی از کانال های مخابراتی ساده و در عین حال مهم، کانال باینری متقارن را بررسی کنیم. این کانال این گونه عمل می کند که در هر بار ارسال یک بیت، با احتمال ϵ بیت مربوطه را تغییر می دهد. به شکل زیر توجه کنید.



شکل ۱: کانال باینری متقارن با پارامتر ϵ

الف) فرض کنید ورودی به این کانال باینری متقارن توزیع برنولی با پارامتر p داشته باشد. می خواهیم بدانیم دنباله ی خروجی این کانال از چه توزیعی پیروی می کند.

$$P(out = 0) = P(out = 0 \mid in = 0\ ) \times P(in = 0) + P(out = 0 \mid in = 1\ ) \times P(in = 1)$$

$$P(out = 1) = P(out = 1 \mid in = 0\ ) \times P(in = 0) + P(out = 1 \mid in = 1\ ) \times P(in = 1)$$

$$P(out = 0) = (1 - \varepsilon)(1 - p) + (\varepsilon)(p) = 1 + 2p\varepsilon - \varepsilon - p$$

$$P(out = 1) = (\varepsilon)(1 - p) + (1 - \varepsilon)(p) = \varepsilon + p - 2p\varepsilon$$

بنابراین خروجی مدل نیز توزیع برنولی با پارامتر $(\varepsilon + p - 2p\varepsilon)$ را دارد.

ب) تابعی بنویسید که دنباله ی ورودی کانال و مقدار ϵ را دریافت کند و خروجی آن دنباله ی خروجی کانال باشد. فرمت تابع شما باید به صورت زیر باشد :

Output = BSC(Input, ϵ)

بدین منظور ابتدا یک بردار هم طول دنباله ورودی تعریف می کنیم که از توزیع برنولی با پارامتر ϵ پیروی می کند. برای هر درایه ورودی اگر مقدار درایه متناظر آن در برداری که خودمان تعریف کردیم 1 بود (متناظر احتمال ϵ) آن بیت ورودی در خروجی عکس می شود و در غیر این صورت تغییری نمی کند. بنابراین مدلی ساخته ایم که با احتمال ϵ هر بیت ورودی را تغییر می دهد و با احتمال (ϵ -1) خودش را مخابره می کند.

پ) می خواهیم جمله 'Information is the resolution of uncertainty. C.Shannon' را از کانال باینری با e=0.01 عبور دهیم ، سپس جمله خروجی کانال را باسازی کنیم و همچنین خطای مربعی را با تعریف گفته شده محاسبه می کنیم.

ابتدا باید جمله گفته شده را به باینری تبدیل کنیم. بدین منظور ابتدا از تابع dec2bin استفاده می کنیم ، خروجی این تابع یک ماتریس 7 در 55 از نوع char است که متناظر 55 حرف جمله گفته شده است که هر حرف با 7 کاراکتر نمایش داده می شود.سپس با استفاده از unint8 نوع char را به عدد تبدیل می کنیم. عدد متناظر کاراکتر 0 48 و 1 ، 49 است با کم کردن مقدار صفر از خروجی uint8 ، ماتریس باینری عددی را محاسبه می کنیم. سپس با reshape کردن همه ی اعداد را پشت هم میگذاریم و دنباله باینری جمله را به دست می آوریم و به عنوان ورودی و با پارمتر e=0.01 به تابع نوشته شده در بخش قبل می دهیم. خروجی تابع دنباله باینری خروجی کانال مخابراتی است.

بازسازی خروجی کانال : اکنون با reshape کردن بردار خروجی را به صورت 7 در 55 در می اوریم و سپس ترانهاده می کنیم ، مقدار کاراکتر 0 یعنی 48 را به آن اضافه می کنیم و توسط تابع char ماتریس باینری متناظر حروف را به دست می آوریم سپس با تابع bin2dec هر سطر آن (7 رقم باینری) را به عدد معادل asci آنها تبدیل و به تابع char می دهیم تا 55 حرف خروجی را بسازیم و سپس به وسیله ی cellstr جمله ی خروجی کانال را به دست می آوریم.

خطای مربعی را نیز با رابطه زیر برای l بیت مخابره شده حساب می کنیم.

$$\mathcal{E} = \frac{1}{l}\sum_{i=1}^{l}\Big(Output(i) - Input(i)\Big)^2$$

برای run 10 مختلف جمله بازسازی شده و خطای مربعی به صورت زیر است :

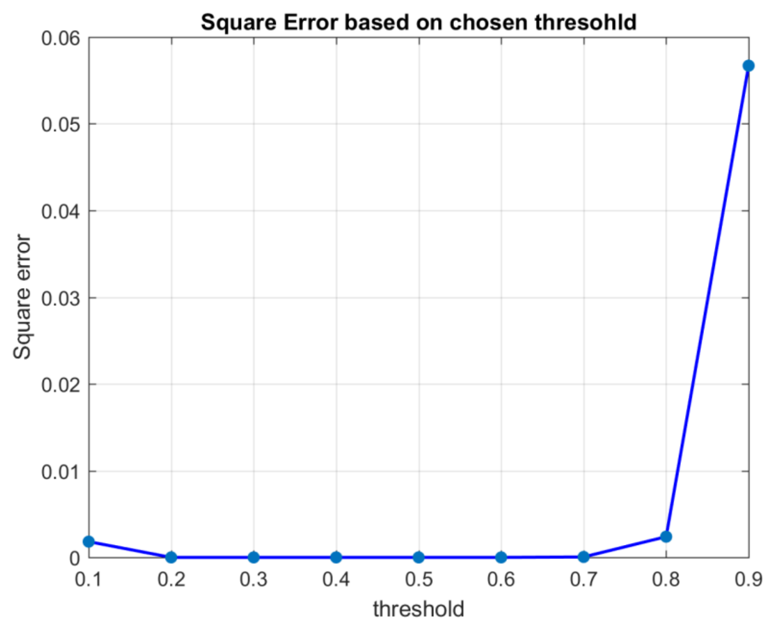
```
Run 1
square error : 0.0051948
Infgrmation ys tha Re{olution of uncertaintY. C.Shajnon
Run 2
square error : 0.0077922
Hfformation is |he resglution of uncerdqioty. C.Qhannon
Run 3
square error : 0.0077922
Informatinn is the resolution of uncErtiinty.(C□shanNon
Run 4
square error : 0.0025974
    nformation is the resolution of uncertaynty. C.Shann/n
Run 5
square error : 0.0077922
Ioformation is the reso,utiOn of uncertainty> C.Shanno~
Run 6
square error : 0.0051948
Informathon is the reso|ution Of uncertaInty.$C.Shannon
Run 7
square error : 0.0025974
MnforMa4ion is the resolution of uncertainty. C.Shanfon
Run 8
square error : 0.0051948
Information is the resolution of unce{tainty. C.Shannon
Run 9
square error : 0
InforeAtion is phe resolu4ion gf uncertainty. C.Shannon
Run 10
square error : 0.0077922
Informa|ion is!the resolution of Uncertainty. CnShannon
mean square error : 0.0051948
```

شکل 2 : خروجی های کانال باینری متقارن در ده بار استفاده از آن و میانگین خطای مربعی

ت) در این قسمت با استفاده از کدینگ تکراری تلاش میکنیم خطای کانال را کاهش دهیم. بدین ترتیب که به جای ارسال یگانه هر بیت آن را ده بار ارسال می کنیم و در خروجی از آن ده بیت میانگین می گیریم و با یک ترشولد انتخابی مقایسه می کنیم و اگر میانگین ده ارسال یک بیت از آن ترشولد کمتر بود صفر و اگر بیشتر بود آن بیت را یک در نظر می گیریم.

برای ترشولد های بین 0 تا 1 با گام 0.1 خطای مربعی بر حسب ترشولد به شکل زیر است :

برای به دست اوردن خطا 100 بار هر پروسه را برای هر ترشولد تکرار کرده ایم و میانگین خطا ها را گزارش کرده ایم.



شکل ۳ : خطای مربعی بر حسب ترشولد مقایسه در کدینگ تکراری

تحلیل :

ما ده بار یک بیت را ارسال می کنیم. بنابراین X ها با هم برابر و برابر بیتی است که در هر مرحله ارسال می شود. در خروجی Y ها را دریافت می کنیم هرکدام به احتمال e و مستقلا نسبت به X تغییر کرده اند. Z را به عنوان میانگین Y ها تعریف می کنیم. Z می تواند مقادیر ۰ تا ۱ را با گام ۰/۱ به خود بگیرد.

$$X_1, X_2, .., X_{10}$$

$$Y_1, Y_2, .., Y_{10}$$

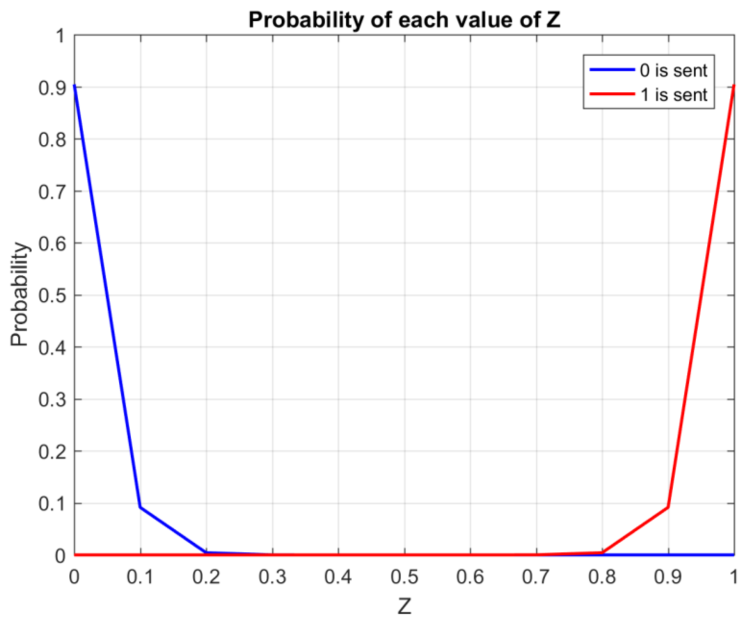
$$Z = \frac{Y_1 + Y_2 + \cdots + Y_{10}}{10}$$

حال اگر بیت ارسالی صفر و یا یک بوده باشد احتمال هر یک از مقادیر Z به شکل زیر است :

$$P(Z = i) = \binom{10}{10i} \varepsilon^{10i} (1 - \varepsilon)^{10-10i} \qquad i = 0, 0.1, 0.2, \dots, 1 \qquad 0 \; is \; send$$

$$P(Z = i) = \binom{10}{10i} \varepsilon^{10-10i} (1 - \varepsilon)^{10i} \qquad i = 0, 0.1, 0.2, \dots, 1 \qquad 1 \; is \; send$$

حال برای e=0.01 این توزیع ها را رسم می کنیم.



شکل 4 : احتمال اینکه میانگین ده ارسال هر بیت در خروجی برابر مقادیر بین صفر و یک باشد

ملاحظه می کنیم در هر دو حالت ارسال صفر و یک مقدار میانگیری شده فقط تا دو دهم می تواند جابه جا شود و احتمال بقیه حالت ها تقریبا صفر است بنابراین می توان گفت که ترشولد بین مقادیر 0.3 تا 0.7 برای این اپسیلون حد مناسبی برای مقایسه است. که در خروجی خطای مربعی بر حسب ترشولد این امر به وضوح دیده می شود و مقدار خطا برای این ترشلود ها تقریبا صفر است.

علت بیشتر بودن خطای ترشولد های بالاتر (شکل 3) بیشتر بودن تعداد بیت های 1 در پیام است که باعث می شود در ترشولد های بالاتر که خطا ی بیت های یک بیشتر به آن وابسته است خطا افزایش بیابد.

برای هر e به علت تقارن موجود در کانال حد 0.5 بهترین ترشولد برای مینیمم کردن خطای یک یک تک بیت است.

ث) در این قسمت توسط تابع برنولی که در فاز قبلی نوشته بودیم ماتریس کدینگ گفته شده را به فرمت زیر می سازیم :

$$C = RandomCode(n)$$

که هر درایه آن مستقل از بقیه توزیع برنولی با پارامتر 0.5 (ساده ترین حالت گفته شده) دارد.

n : تعداد ستون های ماتریس و تعداد سطر های ماتریس طبق متن پروژه 26 است.

$$C = \begin{pmatrix} Codeword\ of\ "A" \\ Codeword\ of\ "B" \\ \vdots \\ Codeword\ of\ "Z" \end{pmatrix}_{26\times n} = \begin{pmatrix} 1 & 0 & 1 & \cdots & 1 \\ 0 & 0 & 0 & \dots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & 1 & \cdots & 0 \end{pmatrix}_{26\times n}$$

شکل 5: نمونه یک ماتریس کدینگ تصادفی

ج) همینگ زیاد از ویژگی های خوب یک ماتریس کدینگ است . در این قسمت می خواهیم توزیع همینگ بین دو سطر ماتریس کدینگ تصادفی ای را که ساختیم به دست آوریم :

یافتن همینگ (تعداد بیت های متفاوت) بین دو سطر متمایز i و j :

ابتدا متغیر تصادفی X_k که قدر مطلق تفاضل بین k امین بیت دو سطر متمایز i و j ام است.

حال متغیر تصادفی HD_{ij} را به صورت مجموع این X_k ها تعریف می کنیم.

می توان همینگ را به صورت جمع xor بیت به بیت نیز تعریف کرد.

ابتدا توزیع X_k ها را به دست می آوریم : می دانیم توزیع هر بیت برنولی با پارامتر p=0.5 است.

احتمال یک بودن قدرمطلق تفاضل بدین معناست که یکی صفر باشد و دیگری یک باشد :

$$P(X_k = 1) = p(1-p) + p(1-p) = 2p - 2p^2$$

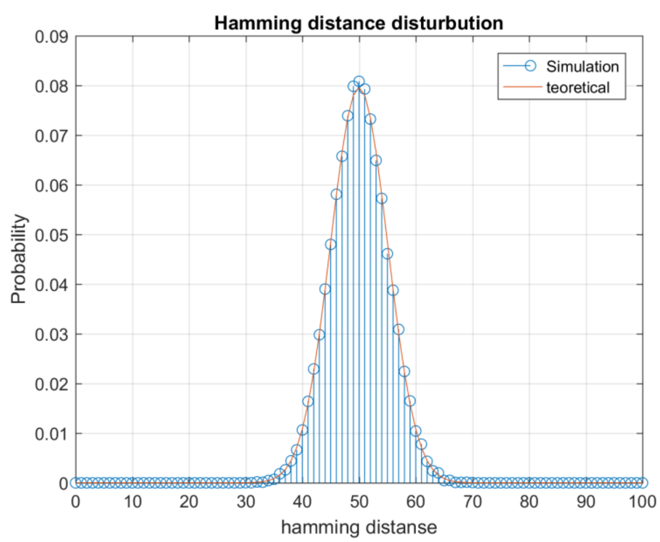
احتمال یک بودن قدرمطلق تفاضل این است که هر دو درایه دوسطر یا یک باشند و یا صفر باشند :

$$P(X_k = 0) = p(p) + (1-p)(1-p) = 1 + 2p + 2p^2$$

پس هرکدام از X_k ها توزیع برنولی با پارامتر $2p - 2p^2$ دارند (دراینجا p=0.5 پس $2p - 2p^2 = 0.5$) که این نتیجه می دهد که جمع آنها که همان متغیر تصادفی همینگ خواسته شده است توزیع دو جمله ای با پارامتر های n و0.5 دارد :

$$P(\mathrm{HD_{ij}} = k) = \binom{n}{k}(0.5)^n \qquad\qquad k = 0, 1, \dots, n$$

حال توسط شبیه سازی با متلب توزیع فاصله ی همینگ را برای همه ی حالت های غیر تکراری ماتریس رندوم بخش قبل و تکرار صد باره ی این فرایند به دست می اوریم. نتیجه به صورت زیر است. همچنین مقدار تئوری را نیز رسم کرده ایم که به خوبی با مقادیر شبیه سازی شده هماهنگی دارد. (n را برابر 100 گرفتیم.)



شکل 6 : توزیع فاصله ی همینگ بین دو سطر متفاوت i و j تئوری و شبیه سازی

چ) ویژگی تعیین کننده ی خوب بودن یک ماتریس کد تصحیح خطا، کمینه ی فاصله ی همینگ آن است که خود یک متغیر تصادفی است. که در این قسمت توزیع آن را محاسبه می کنیم.

توزیع HD_{min} :

می دانیم توزیع فاصله ی همینگ های بین دو سطر متفاوت i و j دو جمله ای با پارامتر n و 0.5 است. سوالی که در اینجا مطرح می شود این است که ایا فاصله ی همینگ های بین دو سطر متفاوت از هم مستقل هستند و یا خیر. همانگونه که گفتیم فاصله ی همینگ بین دو سطر را می توان به صورت جمع xor بیت به بیت آنها تعریف کرد. می توان ثابت کرد که xor دو بیت از هر دو آن بیت ها مستقل است اگر آن بیت ها برنولی با پارامتر 0.5 باشند. (شرطی که در اینجا نیز برقرار است.) به وضوح xor بیت به بیت سطر یک با بقیه سطر ها از هم مستقل است اما برای مثال بین xor سطر 2 و 3 با xor های 1 و 2 و 1و 3 رابطه ی xor ای وجود دارد اما چون شرط گفته شده برقرار است با وجود این رابطه استقلال بنابر تعریف آماری برقرار است و هر تابعی از xor ها که در این جا جمع سطر های آنها که همان همینگ مورد نظر است از هم مستقل اند . بنابراین می توان فرض استقلال را در نظر گرفت.

اثبات اینکه xor دو بیت که توزیع برنولی 0.5 دارند از خود آن دو بیت مستقل است :

$$X_3 = X_1 \oplus X_2,$$

for any $a, b \in \{0,1\}$

$$\begin{aligned}
P(X_1 = a, X_3 = b) &= P(X_1 = a, X_1 \oplus X_2 = b) \\
&= P(X_1 = a, X_2 \oplus a = b) \\
&= P(X_1 = a, X_2 = a \oplus b) \\
&= P(X_1 = a)P(X_2 = a \oplus b) \\
&= \frac{1}{2} \cdot \frac{1}{2},
\end{aligned}$$

where the last two steps use the fact that X_1 and X_2 are independent, and both are Bernoulli(1/2) random variables. We also have, for any $a \in \{0,1\}$

بنابراین مسله ی یافتن مینیمم همینگ را می توان به صورت یافتن مینیمم تعدادی متغیر iid با توزیع Bionomial(n,0.5) بیان کرد که این تعداد برابر تعداد سطر های متمایز یعنی انتخاب 2 از 26 (325) در ماتریس کدینگ ما است. بنابراین :

$$P(\mathrm{HD_{min}} \le \mathrm{k}) = 1 - P(\mathrm{HD_{min}} > \mathrm{k}) = 1 - \prod_{i=1}^{n} \prod_{j=i+!}^{n} P(\mathrm{HD_{ij}} > k) = 1 - P(\mathrm{HD_{ij}} > k)^{\binom{26}{2}}$$

$$P(\mathrm{HD_{ij}} > k) = 1 - \sum_{i=0}^{k} \binom{n}{i} (0.5)^n$$

$$P(\mathrm{HD_{min}} \le \mathrm{k}) = 1 - (1 - \sum_{i=0}^{k} \binom{n}{i} (0.5)^n)^{\binom{26}{2}}$$

$$P(\mathrm{HD_{min}} = \mathrm{k}) = P(\mathrm{HD_{min}} \le \mathrm{k}) - P(\mathrm{HD_{min}} \le \mathrm{k} - 1)$$

$$P(\mathrm{HD_{min}} = \mathrm{k}) = 1 - \left(1 - \sum_{i=0}^{k} \binom{n}{i} (0.5)^n\right)^{\binom{26}{2}} - 1 + \left(1 - \sum_{i=0}^{k-1} \binom{n}{i} (0.5)^n\right)^{\binom{26}{2}}$$

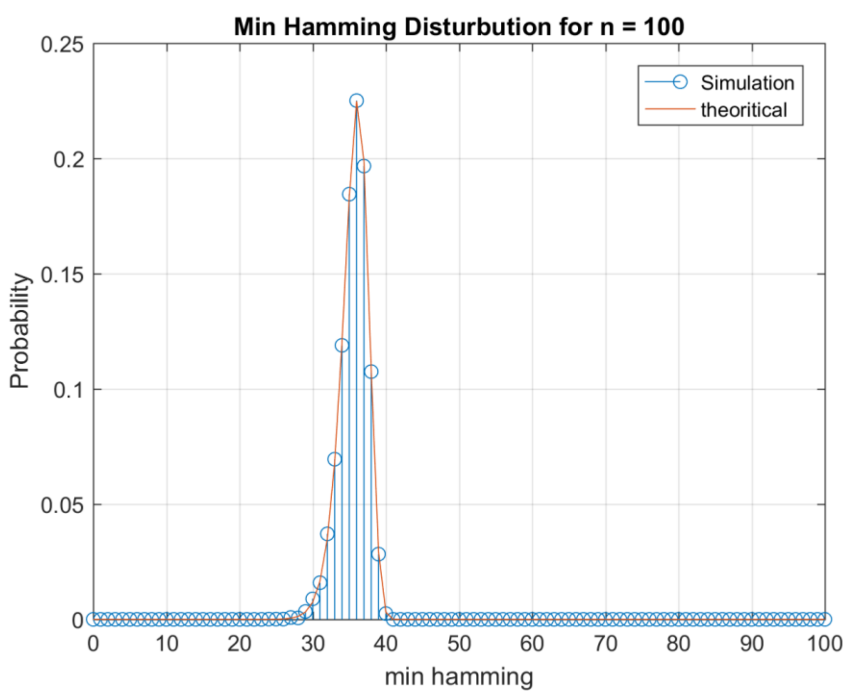
$$P(\mathrm{HD_{min}} = \mathrm{k}) = \left(1 - \sum_{i=0}^{k-1} \binom{n}{i} (0.5)^n\right)^{325} - \left(1 - \sum_{i=0}^{k} \binom{n}{i} (0.5)^n\right)^{325}$$

$$k = 0, 1, \dots, n$$

البته این نتیجه با فرض استقلال صحیح است.

حال به وسیله ی شبیه سازی تلاش می کنیم این توزیع را به دست آوریم و با توزیع نظری بالا مقایسه کنیم.

نتیجه به صورت زیر است ، که نشان می دهد نتایج شبیه سازی و نظری مطابقت خوبی با هم دارند و موید فرض استقلال است : (n=100 در نظر گرفته شده است.)



شکل 7 : توزیع مینیمم فاصله ی همینگ بین دو سطر متفاوت در ماترس کدینگ تصادفی تئوری و شبیه سازی

درباره ی توزیع مینیمم متغیر های iid در لینک زیر نیز می توان نتیجه جالبی دید که توزیع آن را توزیع Gumbel بیان میکند.

https://stats.stackexchange.com/questions/134190/asymptotic-distribution-of-the-max-min-of-iid-binomial-variables

درباره ی توزیع Gumbel :

https://en.wikipedia.org/wiki/Gumbel_distribution

شایان ذکر است که اگر برای کدینگ تصحیح خطای دلخواهی، کمینه ی فاصله ی همینگ برابر m باشد، این کدینگ می تواند در برابر [m/2]-1 بیت خطا مقاوم باشد. زیرا در این صورت حتی در صورت [m/2]-1 بیت خطا باز هم نزدیکترین کد همینگ موجود باز هم کد همینگ صحیح است ، برای مثال اگر کد یک و دو کمترین همینگ را داشته باشند (در اینجا 8) تا همینگ 3 هم باز هم کد صحیح تشخیص داده می شود و چون این حالت مینیمم است بقیه کد ها حتی از هم فاصله ی بیشتری دارند و تشخیص در آنها تا این مقدار خطا باز هم به درستی انجام می شود. (البته در متن تمرین کف تقسیم بر دو گفته شده است اما با توجه به مسله جواب صحیح سقف به نظر می رسد.) :

شکل 8 : مثالی از تشخیص در کد همینگ

محاسبه ی خطا :

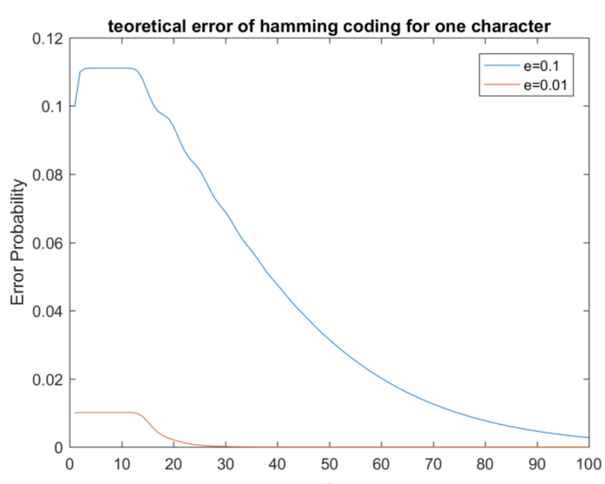
احتمال خطا برای مخابره یک حرف را می یابیم.

$$P(error\,|\,(\min hamming = m\,, n\,bits)) = \sum_{i=1+\left[\left[\frac{m}{2}\right]-1\right]_+}^{n} \binom{n}{i} (1-\varepsilon)^{n-i}\varepsilon^{i}$$

$$P(error\,|n\,bits) = \sum_{m=0}^{n} P(error\,|\,(\min hamming = m\,, n\,bits))P(\min hamming = m)$$

$$P(error\,|n\,bits) = \sum_{m=0}^{n}\left(\left(\sum_{i=1+\left[\left[\frac{m}{2}\right]-1\right]_+}^{n} \varepsilon^{i}\;(1-\varepsilon)^{n-i}\binom{n}{i}\right) \times \left(1-\sum_{i=0}^{m-1}\binom{n}{i}(0.5)^{n}\right)^{325} - \left(1-\sum_{i=0}^{m}\binom{n}{i}(0.5)^{n}\right)^{325}\right)$$

این احتمال را برای e=0.01 و e=0.1 بر حسب n می کشیم. نتیجه به صورت زیر است :



شکل ۹ : احتمال خطا در مخابره یک حرف برحسب n برای دو e گفته شده

احتمال خطا برای مخابره یک بیت در این حالت مستقل از بقیه نیست و به تعداد خطا ها وابسته است اما حتما از مقدار خطای یک حرف کمتر است و بنابر این با افزایش n به سمت صفر می رود. اما توزیع تعداد خطا ها (توزیع عدد خطای مربعی) مشابه بالا و با کمی تغییر در رابطه بالا به دست می آید.

$$P(\#error = k \mid (\min hamming = m\,, n\,bits)) = \begin{cases} 0 & k \le \left[\left\lceil \frac{m}{2} \right\rceil - 1\right]_{+} \\ \binom{n}{k} (1-\varepsilon)^{n-k} \varepsilon^{k} & o.w \end{cases}$$

$$P(\#error = k \mid n\,bits) = \sum_{m=0}^{n} P(error \mid (\min hamming = m\,, n\,bits)) P(\min hamming = m)$$

$$P(\#error = k \mid n\,bits) = \sum_{m=0}^{n} \left(\begin{cases} 0 & k \le \left[\left\lceil \frac{m}{2} \right\rceil - 1\right]_{+} \\ \binom{n}{k} (1-\varepsilon)^{n-k} \varepsilon^{k} & o.w \end{cases} \times \left(1 - \sum_{i=0}^{m-1} \binom{n}{i} (0.5)^{n}\right)^{325} - \left(1 - \sum_{i=0}^{m} \binom{n}{i} (0.5)^{n}\right)^{325} \right)$$

نکته اینجاست که بت های یک حرف از هم مستقل نیستند و خطای یک حرفی که بدست آوردیم معتبر تر است.

ح) در این قسمت چون 26 حرف برای کد کردن پیام داده شده ناکافی است یک ماتریس کدینگ جدید می سازیم که 54 سطر دارد و حروف بزرگ و کوچک و . و فاصله را کد میکند . سپس به هر کاراکتر یک سطر نسبت می دهیم. ابتدا توسط تابع dec2bin پیام را به باینری کد میکنیم سپس از روی تناظر کد های باینری و کد تصادفی خروجی تابع Randomcode54 جمله را کد می کنیم و سپس حروف را پشت سر هم می گذاریم و بردار صفر و یک ها را از کانال عبور می دهیم. خطای حاصل از کانال بدون اعمال قاعده ی همینگ را گزارش می کنیم. سپس بردار خروجی را n تا n تا جدا کرده و آن را به نزدیکترین کد همینگ تبدیل می کنیم و با استفاده از تناظر بین کد ها آن را به کد باینری تبدیل می کنیم . خطای مربعی بعد از اعمال این قاعده و همچنین جمله باسازی شده را گزارش می کنیم. همچنین کمترین فاصله ی همینگ را نیز گزارش می کنیم :

```
for e = 0.01 and n = 100

squareerrorbeforenearesthamming =

    47


detectedmessage =

Information is the resolution of uncertainty. C.Shannon


squareerrorafternearesthamming =

     0


minhamming =

    36
```

ملاحضه می کنیم همانگونه که از نتیجه بخش قبل انتظار میرفت خطا با کدینگ همینگ به سمت صفر می رود. و متن مخابره شده کاملا با معنا و صحیح است.

خ) قسمت قبل را برای e=0.1 و n=200 تکرار می کنیم. این بار نیز با وجود افزایش احتمال خطا همچنان خطا بعد اعمال قاعده همینگ صفر می ماند . این به عات افزایش n و به تبع آن مینیمم همینگ و مقاوم شدن در مواجه با خطا هاست. در عبور از کانال 1046 خطا داشته ایم که با اعمال قاعده همینگ به صفر رسیده است.

نتایج :

```
for e = 0.1 and n = 200

squareerrorbeforenearesthamming =

        1046


detectedmessage =

Information is the resolution of uncertainty. C.Shannon


squareerrorafternearesthamming =

     0


minhamming =

    78
```

د) متن سوال ناواضح است ، زیرا گفته نشده خطای چه چیزی ، اگر منظور خطا در مخابره یک بیت باشد باید با توجه با احتمال خطا ای که در قسمت های قبل به دست آودیم با ثابت نگاه داشتن e نامعادله را برای کمتر کردن خطا از 0.05 بر حسب n حل کنیم و بدین ترتیب n بهینه را بیابیم.

مدولاسیون دامنه ی سیگنال

الف) قاعده تصمیم گیری مبنا بر بیشینه کردن احتمال پسین به صورت زیر است :

$$argmax\ p(s|r) = argmax\ \frac{p(r|s)p(s)}{p(r)} = argmax\ p(r|s)p(s)$$

پس ما می خواهیم $p(r|s)p(s)$ را ماکسیمم کنیم. بنابراین در هر حالت باید s ای را انتخاب کنیم که این احتمال را بیشتر می کند یا به عبارت دیگر :

$$if\ P(s=1)f(r|s=1) > P(s=0)f(r|s=0):\ \text{"1"}\ is\ sent$$

$$if\ P(s=1)f(r|s=1) < P(s=0)f(r|s=0):\ \text{"0"}\ is\ sent$$

حال می دانیم که هر یک از احتمال های سمپل صفر و یا یک بودن در سمپل ورودی برابر 0.5 است و توزیع خروجی به شرط ورودی را نیز به دست می آوریم.

$$r = \pm\sqrt{Eg} + n\ , n \sim Gaussian(0, N)$$

$$s = 1 \quad , \qquad r = \sqrt{Eg} + n\ , \quad f(r|s=1) = \frac{1}{\sqrt{2\pi N^2}} e^{-\frac{(r-\sqrt{Eg})\text{^}2}{2N}} \sim Gaussian(\sqrt{Eg}, N)$$

$$s = 0 \quad , \qquad r = -\sqrt{Eg} + n\ , \quad f(r|s=0) = \frac{1}{\sqrt{2\pi N^2}} e^{-\frac{(r+\sqrt{Eg})^2}{2N}} \sim Gaussian(-\sqrt{Eg}, N)$$

پس داریم :

$$P(s=1)f(r|s=1) = 0.5 \times \frac{1}{\sqrt{2\pi N^2}} e^{-\frac{(r-\sqrt{Eg})^2}{2N}}$$

$$P(s=0)f(r|s=0) = 0.5 \times \frac{1}{\sqrt{2\pi N^2}} e^{-\frac{(r+\sqrt{Eg})^2}{2N}}$$

پس قاعده تصمیم گیری را می توان این گونه نوشت :

$$0.5 \times \frac{1}{\sqrt{2\pi N^2}} e^{-\frac{(r-\sqrt{Eg})^2}{2N}} > 0.5 \times \frac{1}{\sqrt{2\pi N^2}} e^{-\frac{(r+\sqrt{Eg})^2}{2N}} \qquad : 1\ is\ sent$$

$$e^{-\frac{(r-\sqrt{Eg})^2}{2N}} > e^{-\frac{(r+\sqrt{Eg})^2}{2N}} \qquad : 1\ is\ sent$$

تابع نمایی تابعی صعودی است :

$$-\frac{\left(r-\sqrt{Eg}\right)^2}{2N} > -\frac{\left(r+\sqrt{Eg}\right)^2}{2N} \qquad : 1\ is\ sent$$

باتوجه به اینکه دو طرف هم علامت هستند با حذف منفی ها جهت نامساوی ها عوض می شود و قسمت های مشابه باز ازز دو طرف خط می خورند :

$$\left(r-\sqrt{Eg}\right)^2 < \left(r+\sqrt{Eg}\right)^2 \qquad : 1\ is\ sent$$

$$\left(r-\sqrt{Eg}\right)^2 - \left(r+\sqrt{Eg}\right)^2 < 0 \qquad : 1\ is\ sent$$

$(2r\ )\left(-2\sqrt{Eg}\right) < 0 \quad : 1\ is\ sent$ اتحاد مزدوج

$$r > 0 \quad : 1\ is\ sent$$

به شیوه ی کاملا مشابه می توان گفت :

$$r < 0 \quad : 0\ is\ sent$$

بنابراین قاعده تصمیم گیری به شکل زیر است :

$$r > 0 \quad : 1\ is\ sent$$

$$r < 0 \quad : 0\ is\ sent$$

ب) این بار نیز مشابه دفعه قبل عمل می کنیم و قاعده تصیمیم گیری را می یابیم.

$$argmax\ p(s|r) = argmax\ \frac{p(r|s)p(s)}{p(r)} = argmax\ p(r|s)p(s)$$

$$if\ P(s=1)f(r|s=1) > P(s=0)f(r|s=0):\ "1"\ is\ sent$$

$$if\ P(s=1)f(r|s=1) < P(s=0)f(r|s=0):\ "0"\ is\ sent$$

$$P(s=1)f(r|s=1) = p_1 \times \frac{1}{\sqrt{2\pi N}} e^{-\frac{\left(r-\sqrt{Eg}\right)^2}{2N^2}}$$

$$P(s=0)f(r|s=0) = (1-p_1) \times \frac{1}{\sqrt{2\pi N}} e^{-\frac{\left(r+\sqrt{Eg}\right)^2}{2N^2}}$$

$$p_1 \times \frac{1}{\sqrt{2\pi N}} e^{-\frac{\left(r-\sqrt{Eg}\right)^2}{2N^2}} > (1-p_1) \times \frac{1}{\sqrt{2\pi N}} e^{-\frac{\left(r+\sqrt{Eg}\right)^2}{2N^2}} \qquad : 1\ is\ sent$$

$$p_1 \times e^{-\frac{\left(r-\sqrt{Eg}\right)^2}{2N}} > (1-p_1) \times e^{-\frac{\left(r+\sqrt{Eg}\right)^2}{2N}} \qquad : 1\ is\ sent$$

با ln گرفتن از طرفین داریم :

$$\ln(p_1)+-\frac{\left(r-\sqrt{Eg}\right)^2}{2N}>\ln(1-p_1)+-\frac{\left(r+\sqrt{Eg}\right)^2}{2N} \qquad : 1\ is\ sent$$

$$\frac{\left(r+\sqrt{Eg}\right)^2}{2N}-\frac{\left(r-\sqrt{Eg}\right)^2}{2N}>\ln(1-p_1)-\ \ln(p_1) \qquad : 1\ is\ sent$$

$$\frac{1}{2N}(\left(r+\sqrt{Eg}\right)^2-\left(r-\sqrt{Eg}\right)^2)>\ln(1-p_1)-\ \ln(p_1) \qquad : 1\ is\ sent$$

$$\frac{1}{2N}(2r)(2\sqrt{Eg})>\ln(1-p_1)-\ \ln(p_1) \qquad : 1\ is\ sent$$

$$\frac{1}{2N}(2r)(2\sqrt{Eg})>\ln(\frac{1-p_1}{p_1}) \qquad : 1\ is\ sent$$

قاعده تصمیم گیری :

$$r>\ln(\frac{1-p_1}{p_1})\times\frac{N}{2\sqrt{Eg}} \qquad : 1\ is\ sent$$

$$r<\ln(\frac{1-p_1}{p_1})\times\frac{N}{2\sqrt{Eg}} \qquad : 0\ is\ sent$$

پ) احتمال خطا در تشخیص سمبل ارسالی را برحسبp1 ، Egو N به صورت تئوری

$$P(error)=p(r=1\,,s=0)+p(r=0,s=1)$$

$$P(error)=p(r=1|s=0)\times p(s=0)+p(r=0|s=1)\times p(s=1)$$

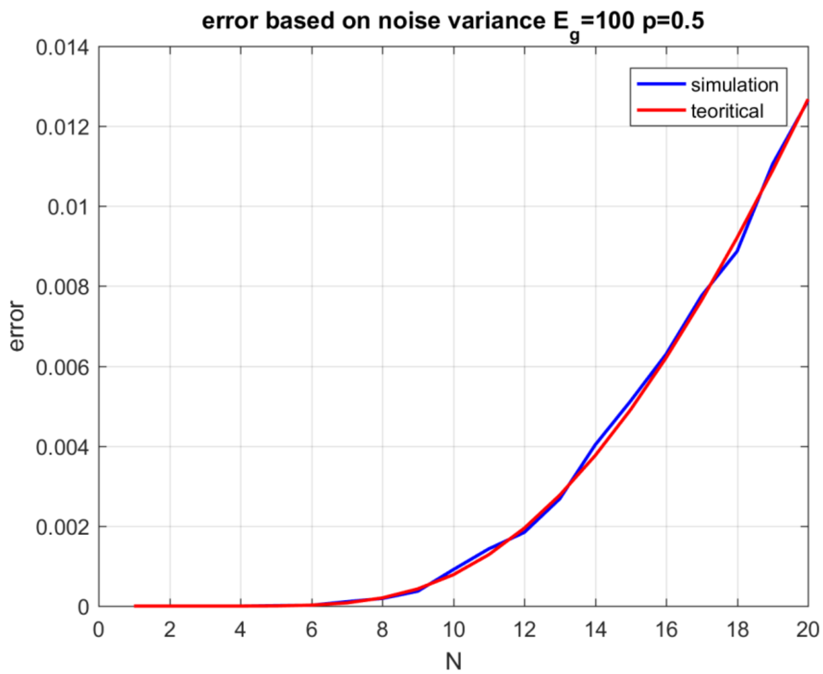
$$P(error)=(1-p_1)\int_{\ln(\frac{1-p_1}{p_1})\times\frac{N}{2\sqrt{Eg}}}^{+\infty}\frac{1}{\sqrt{2\pi N}}e^{-\frac{\left(r+\sqrt{Eg}\right)^2}{2N}}+p_1\int_{-\infty}^{\ln(\frac{1-p_1}{p_1})\times\frac{N}{2\sqrt{Eg}}}\frac{1}{\sqrt{2\pi N}}e^{-\frac{\left(r-\sqrt{Eg}\right)^2}{2N}}$$

$$P(error)=(1-p_1)\left(1-G\left(\frac{\ln\left(\frac{1-p_1}{p_1}\right)\times\frac{N}{2\sqrt{Eg}}+\sqrt{Eg}}{\sqrt{N}}\right)\right)+p_1G\left(\frac{\ln\left(\frac{1-p_1}{p_1}\right)\times\frac{N}{2\sqrt{Eg}}-\sqrt{Eg}}{\sqrt{N}}\right)$$

$$P(error)=0.5\left(1-G\left(\frac{0+10}{\sqrt{N}}\right)\right)+0.5G\left(\frac{0-10}{\sqrt{N}}\right)=0.5-0.5G\left(\frac{10}{\sqrt{N}}\right)+0.5G\left(\frac{-10}{\sqrt{N}}\right)$$

ت) شبیه سازی خطا برای Eg=100 و p=0.5 بر حسب مقادیر مختلف N

مشاهده می کنیم مقدار نظری و شبیه سازی با هم تطابق دارد.



ث) خطا های خواسته شده به صورت زیر است.

مشاهده می شود که کدینگ تکرای برای مقادیر گفته شده سبب کاهش خطا می شود می دانیم جمع دو متغییر گاووسی مستقل گاویس با پارامتر جمع واریانس ها و میانگین هاست بنابرلین در حالت ده بار تکرار در مجموع این ده بار دو توزیع شرطی یکی گاووسی با میانگین 20 و واریانس 10 و دیگری با میانگین -20 و واریانس 10 می شود و این سبب می شود که دو توزیع به خوبی از هم جدا شوند و خطا کاهش بیابد.

```
square error for reapeted coding
     0

square error for simple coding
        2270
```