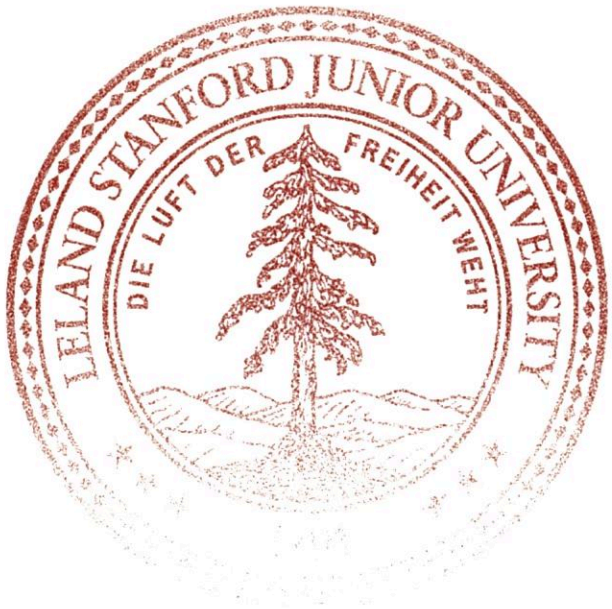
Chris Piech
CS109

Spring Quarter
June 13th, 2017

CS109 Final Exam

This is a closed calculator/computer exam. You are, however, allowed to use notes in the exam. The last page of the exam is a Standard Normal Table, in case you need it. You have 3 hours (180 minutes) to take the exam. The exam is 180 points, meant to roughly correspond to one point per minute of the exam. You may want to use the point allocation for each problem as an indicator for pacing yourself on the exam.

In the event of an incorrect answer, any explanation you provide of how you obtained your answer can potentially allow us to give you partial credit for a problem. For example, describe the distributions and parameter values you used, where appropriate. It is fine for your answers to include summations, products, factorials, exponentials, and combinations, unless the question specifically asks for a numeric quantity or closed form. Where numeric answers are required, the use of fractions is fine.



I acknowledge and accept the letter and spirit of the honor code. I pledge to write more neatly than I have in my entire life:

Signature: Xiaoqi Zhou

Family Name (print): ZHOU

Given Name (print): XIAOQI

1 Random Encounter Redux [14 points]

You walk into a gathering at Stanford with R number of Stanford students. What is the probability that you know more than 5 people at the gathering?

Let P be the number of students at Stanford and let F be the number of Stanford students that you know. Assume that each Stanford student is equally likely to be at the gathering.

Combinations of the students I know

C_n^F

Combinations of the $R-n$ students I don't know

C_{R-n}^{P-F}

Combinations of the R students with n students I know

$C_n^F \cdot C_{R-n}^{P-F}$

Total possible combination

C_R^P

$$p' = P\{\text{less or equal than 5 students I know}\} = \sum_{n=1}^{5} \frac{C_n^F \cdot C_{R-n}^{(P-F)}}{C_R^P}$$

$$P\{\text{students I know} > 5\} = 1 - p' = 1 - \sum_{n=1}^{5} \frac{C_n^F C_{R-n}^{P-F}}{C_R^P}$$

2 Letters of Recommendation [16 points]

To get a job or internship next summer, you submit recommendation letters from two professors. Unfortunately, you can never be completely certain what your recommendation letters say.

You estimate that if you had two good letters, the probability that you would get the job is 0.75. If you only have one good letter, the probability is 0.2 and if you have no good letters the probability is 0.05.

You believe that the two letters are independent, the probability that the first letter is good = 0.8 and the probability that the second letter is good is 0.5.

a. What is the probability of getting two good letters?

$$P\{2 \text{ good letters}\} = 0.8 \times 0.5 = 0.4$$

b. What is the probability of getting exactly one good letter?

$P\{\text{only get first letter}\} = 0.8 - 0.8 \times 0.5 = 0.4$

$P\{\text{only get second letter}\} = 0.5 - 0.8 \times 0.5 = 0.1$

$P\{\text{only get one letter}\} = 0.4 + 0.1 = 0.5$

c. What is the probability of getting the job?

$P\{\text{getting the job}\} = 0.4 \times 0.75 + 0.5 \times 0.2 + 0.1 \times 0.05$

$= 0.3 + 0.1 + 0.005$

$= 0.405$

d. You got the job. What is the probability that you had two good letters?

$A = \{\text{get a good job}\}$

$B = \{\text{had 2 good letters}\}$

$P(B|A) = \dfrac{P(A|B)P(B)}{P(A)} = \dfrac{0.75 \times 0.4}{0.405} \approx 0.75$

3 Hindenbug [16 points]

You are testing software and discover that your program has a non-deterministic bug that causes catastrophic failure. Your program was tested for 400 hours and the bug occurred twice.

a. Based on the rate of occurrence that you observed, what is the probability that the bug will occur fewer than five times if the program is used for another 400 hours?

Y is a poisson ~~random~~ random variable.

$E[Y] = \lambda = 2.$

$P\{Y = i\} = e^{-\lambda} \frac{\lambda^i}{i!}$

$P\{Y \leq 5\} = \sum_{i=0}^{4} e^{-2} \frac{2^i}{i!}$

b. Each user uses your program to complete a three hour long task. If the hindenbug manifests the user will immediately stop using your program. What is the probability that the bug manifests for a given user?

$\lambda' = \frac{3}{400} \lambda = \frac{3}{200}$

$P(Y' > 0) = 1 - P(Y' = 0) = 1 - \frac{2^0}{0!} e^{-\frac{3}{200}}$

$= 1 - e^{-\frac{3}{200}}$

c. Your program is used by one million users. Use a normal approximation to estimate the probability that more than 10000 users experience the bug. Let p be the solution to part (b).

$X \sim N(1\times10^6 p, 1\times10^6 p)$

$P(X > 1\times10^5) = 1 - \Phi\left(\frac{1\times10^5 - 1\times10^6 p}{\sqrt{1\times10^6 p}}\right)$

4 NBA Finals Week [18 points]

Recall that a team's ability can be modeled by an *Elo score*, which predicts that if teams A and B have respective Elo scores E_A and E_B, then the probability that A wins a game against B, all else equal, is

$$P(A \text{ wins}) = \frac{1}{1+9^{\left(-\frac{E_A - E_B}{400}\right)}}$$

a. Suppose that team A has an Elo rating which is 200 less than the Elo rating for team B. What is the probability that team A wins a game?

$$P(A\text{wins}) = \frac{1}{1+9^{\left(-\frac{-200}{400}\right)}} = \frac{1}{1+9^{\left(\frac{1}{2}\right)}} = \frac{1}{1+3} = \frac{1}{4}$$

b. Suppose the Elo scores of the two teams in the finals are drawn **independently** from a normal distribution with mean $\mu = 1600$ and variance $\sigma^2 = \frac{200^2}{2}$. What is the probability density function for the difference (D) between their Elo ratings? $D = E_A - E_B$.

$$D \sim N(\mu_1 - \mu_2, \text{Var}_1 + \text{Var}_2)$$
$$D \sim N(0, 200^2)$$
$$f(x) = \frac{1}{\sqrt{2\pi} \cdot 200} e^{\frac{-x^2}{2 \times 200^2}}$$

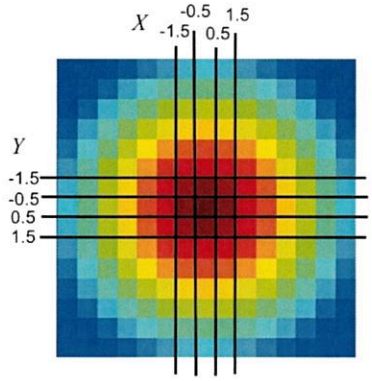
c. The difference between the elo scores of two teams is given by the probability density function from part (b). Write an expression for the probability that team A wins. It is ok to have an integral in your answer.

$$P(A) = \int P(A \mid D = x) P(D = x)$$

$$= \int_{-\infty}^{\infty} \frac{1}{1+\left(9^{\frac{-x}{400}}\right)} \cdot f(x) \cdot dx$$

5 Gaussian Blur [16 points]

In image processing, a Gaussian blur is the result of blurring an image by a Gaussian function. It is a widely used effect in graphics software, typically to reduce image noise.



Gaussian blurring is based on a joint probability distribution of two **independent** random variables: $X \sim N(0,4)$ and $Y \sim N(0,4)$.

a. Write an expression that could be used to calculate $P(X < x, Y < y)$. For full credit your expression should not have integrals.

Since X, Y are independent

$P(X<x, Y<y) = P(X<x)\,P(Y<y)$

$P(X<x) = \Phi(\frac{x}{2})$ $P(Y<y) = \Phi(\frac{y}{2})$ $P(X<x, Y<y) = \Phi(\frac{x}{2})\Phi(\frac{y}{2})$

b. Each pixel is given a weight equal to the probability that X and Y are both within the pixel bounds. The center pixel covers the area where $-0.5 \le x \le 0.5$ and $-0.5 \le y \le 0.5$. What is the weight of the center pixel? Give an expression that has no functions or variables.

$P(-0.5 \le x \le 0.5) = \Phi(\frac{0.5}{2}) - \Phi(\frac{-0.5}{2})$

$= \Phi(\frac{0.5}{2}) - (1 - \Phi(\frac{0.5}{2}))$

$= 2\Phi(\frac{0.5}{2}) - 1$

$= 2 \cdot 0.5987 - 1$

$= 1.1974 - 1$

$= 0.197$

$P(0.5 \le y \le 0.5) = 0.197$

$P(-0.5 \le X \le 0.5, -0.5 \le y \le 0.5) = 0.197^2$

6 Improved Exam Grading [18 points]

You are experimenting with a new training course to prepare TAs for exam grading. You give the new training to 100 graders (group A) and give the old, standard training to another set of 100 graders (group B). All 200 graders are then asked to grade the same assignment.

The data collected by your experiment are the 100 grades given to the assignment by the graders in group A ($A_1 \dots A_{100}$), and the 100 grades given by the graders in group B ($B_1 \dots B_{100}$). You assume that each grade is IID given the grader's group.

You notice that the sample mean of the two groups is about the same. In expectation all graders are accurate. However the sample standard deviation of the grades given by group A was 5 percentage points, whereas the sample standard deviation of grades given by group B was 10 percentage points.

In this question we expect you to write pseudo-code. You will be assessed on the quality of your algorithm, not on programming syntax. Please be as precise as possible. You may use any of the following methods:

Method	Description
size(L)	Returns the number of elements in list L
mean(L)	Returns the arithmetic mean of the values in a list L
join(L_1, L_2)	Returns a list that has all the elements from L_1 and L_2
sum(L)	Returns the sum of all elements in L
sampleReplace(L, n)	Returns a list of n samples, drawn from list L with replacement
sampleNoReplace(L, n)	Returns a list of n samples, drawn from list L without replacement

a. Provide pseudo code for a method **sampleStandard**(S) that calculates the unbiased estimate of standard deviation for a list of IID samples $S = [S_1, S_2, \cdots, S_n]$.

$$S\_std = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(S_i - \bar{S})^2}$$

```
def sampleStandard(S):
    n = size(S)
    S_mean = mean(S)
        for i in range(0, n):
            S_2[i] = (S[i] - S_mean)**2

    S_V = sum(S_2)
    S_std = (1/(n-1) * S_V)**0.5
    return S_std
```

b. Was the difference in standard deviation significant? Write a method **pValue**(A, B) where A is the list of the grades given by group A and B is a list of the grades given by group B. Under the assumption that all 200 grades are identically distributed, calculate the probability of observing a difference in sample standard deviation greater than or equal to five. You may use **sampleStandard**(S) from part (a).

```
def pValue(A, B)
    AB = join(A, B)
    n = 100000
    n_pick = 0
    for i in range(0, n):
        A_select = SampleReplace(AB, Size(A))
        B_select = SampleReplace(AB, Size(B))
        A_select_std = SampleStandard(A_select)
        B_select_std = SampleStandard(B_select)
        if abs(A_select_std - B_select_std) >= 5:
            n_pick = n_pick + 1
    return(n_pick/n)
```

7 Differential Privacy [22 points]

You have a dataset that consists of 100 IID values: $X_1 \ldots X_{100}$ where $X_i \sim \text{Bern}(p_x)$.

A researcher wants to calculate statistics on your data. Since you are mindful about privacy, you decide that you shouldn't give the raw data to the researcher. Instead for every sample X_i, you give the researcher a value Y_i using the following algorithm.

```
# Maximize accuracy, while preserving privacy.
def calculateYi(Xi):
    obfuscate = random()
    if obfuscate:
        return indicator(random())
    else:
        return Xi
```

Where random is a function that returns true or false with equal probability and indicator is a function that returns 1 if the input is true (and 0 otherwise).

a. What is $E[Y_i]$? Give your answer in terms of p_x.

$P(\text{obfuscate}=1) = 0.5$

$P(Y_i = 1) = P(\text{obfuscate}=1) \times 0.5 = 0.25$

$P(Y_i = 0) = P(\text{obfuscat}=1) \times 0.5 = 0.25$

$P(Y_i = X_i) = P(\text{obfuscat}=0)$

we can treat Y_i ~ $\text{Bern}(0.25 + 0.5p_X)$

$E[Y_i] = 0.25 + 0.5p_X$

b. What is $\text{Var}(Y_i)$? Give your answer in terms of p_x.

$$\text{Var}(Y_i) = p'(1-p')$$

$$= (0.25 + 0.5p_X)(1 - 0.25 - 0.5p_X)$$

$$= (0.25 + 0.5p_X)(0.75 - 0.5p_X)$$

$$= -0.25p_X^2 + 0.25p_X + \frac{3}{16}$$

c. Write the distribution of the sample mean $\bar{Y}$ of the samples $Y_1 \ldots Y_{100}$. Explain why the sample mean follows that distribution. Your distribution parameters should be in terms of p_x and/or values calculated in previous parts.

$E[\bar{Y}] = E[Y_i]$

$Var[\bar{Y}] = \frac{Var[Y_i]}{N} = \frac{Var[Y_i]}{100}$

$\bar{Y} \sim N(E[Y_i], \frac{Var[Y_i]}{100})$

d. Given the sample mean $\bar{Y}$, write an expression for an unbiased estimate of p_x. An unbiased estimate is one where the expectation of your estimate should be equal to the true value.

$\hat{p_x} = (\bar{Y} - 0.25) \times 2$

$\hat{p_x} \sim N(2 \times E[Y_i] - 0.5, \frac{4}{100} Var[Y_i])$

$E[\hat{p_x}] = p_x$

e. Write an expression for the probability that your estimate from the previous part is more than 0.1 greater than, or less than, the true probability p_x.

$E[\hat{p_x}] = p_x \Rightarrow P(\hat{p_x} > p_x + 0.1) = P(\hat{p_x} < p_x - 0.1)$

$P(\hat{p_x} > p_x + 0.1 \text{ or } \hat{p_x} < p_x - 0.1) = 2(\hat{p_x} < p_x - 0.1)$

$= 2(1 - \Phi(\frac{0.1}{\frac{1}{5} Var(Y_i)})$

when $Var[Y_i) = -\frac{1}{4}p_x^2 + \frac{1}{4}p_x + \frac{1}{16}$

8 Windfarm Modeling [20 points]

In class we saw how climate sensitivity suggests that there is a fierce urgency to developing clean energy solutions. Wind energy presents many opportunities. However, wind is unpredictable and so using and expanding wind energy requires probability theory. The speed of the wind at a windfarm is a random variable that varies as a *Rayliegh Distribution*. A Rayliegh distribution is parameterized by a single scale parameter θ and has the following probability density function.

$$f_X(x) = \begin{cases} \frac{x}{\theta} e^{-x^2/2\theta} & x \geq 0 \\ 0 & else \end{cases}$$

We wish to model the wind speed on a wind farm. To this end we collect N independent measurements of wind speeds $w_1, w_2, \cdots, w_N$. Find a maximum likelihood estimate of θ if we are modeling the wind speed as coming from a Rayleigh distribution.

$$LL(\theta) = \sum_{i=1}^{n} \log f(x_i|\theta)$$

$$= \sum_{i=1}^{n} \log \frac{x_i}{\theta} e^{-x_i^2/2\theta}$$

$$= \sum_{i=1}^{n} (\log x_i - \log\theta - \frac{x_i^2}{2\theta})$$

$$\frac{\delta LL(\theta)}{\delta\theta} = \sum_{i=1}^{n} (-\frac{1}{\theta} + \frac{x_i^2}{2} \cdot (\frac{1}{\theta^2}))$$

$$= \frac{1}{\theta} \sum_{i=1}^{n} (-1 + \frac{1}{\theta} \cdot \frac{x_i^2}{2})$$

$$= \frac{1}{\theta} \cdot (\frac{1}{2\theta} \sum_{i=1}^{N} x_i^2 - N)$$

Let $\frac{\delta LL(\theta)}{\delta(\theta)} = 0$

$$\theta = \frac{1}{2N} \sum_{i=1}^{N} w_i^2$$

So the MLE of θ is $\frac{1}{2N} \sum_{i=1}^{N} w_i^2$

9 Multiclass Bayes [20 points]

In this problem we are going to explore how to write Naive Bayes for multiple output classes. We want to predict a single output variable Y which represents how a user feels about a book. Unlike in your homework, the output variable Y can take on one of the *four* values in the set {Like, Love, Haha, Sad}. We will base our predictions off of three binary feature variables X_1, X_2, and X_3 which are indicators of the user's taste. All values $X_i \in \{0, 1\}$.

We have access to a dataset with 10,000 users. Each user in the dataset has a value for X_1, X_2, X_3 and Y. You can use a special query method **count** that returns the number of users in the dataset with the given *equality* constraints (and only equality constraints). Here are some example usages of **count**:

count$(X_1 = 1, Y = \text{Haha})$	returns the number of users where $X_1 = 1$ and $Y = \text{Haha}$.
count$(Y = \text{Love})$	returns the number of users where $Y = \text{Love}$.
count$(X_1 = 0, X_3 = 0)$	returns the number of users where $X_1 = 0$, and $X_3 = 0$.

You are given a new user with $X_1 = 1$, $X_2 = 1$, $X_3 = 0$. What is the best prediction for how the user will feel about the book (Y)? You may leave your answer in terms of an argmax function. You should explain how you would calculate all probabilities used in your expression. Use **Laplace estimation** when calculating probabilities.

$$\hat{y} = \underset{y}{\text{argmax}} \sum_{i=1}^{3} \log \hat{P}(X = x_i | Y = y) + \log \hat{P}(Y = y)$$

we can determine $\hat{P}(X = x_i | Y = y)$ and $\hat{P}(Y = y)$ using look up table

For example, let $N' = 10000 + 8$

Y \ X_1	Laplace Est. of $P(X_1 = x_i \mid Y = y)$	
	0	1
Like	$\frac{\text{count}(X_1=0, Y=\text{like})+1}{N'}$	$\frac{\text{count}(X_1=1, Y=\text{like})+1}{N'}$
Love	$\frac{\text{count}(X_1=0, Y=\text{love})+1}{N'}$	$\frac{\text{count}(X_1=1, Y=\text{love})+1}{N'}$
Haha	$\frac{\text{count}(X_1=0, Y=\text{Haha})+1}{N'}$	$\frac{\text{count}(X_1=1, Y=\text{Haha})+1}{N'}$
Sad	$\frac{\text{count}(X_1=0, Y=\text{Sad})+1}{N'}$	$\frac{\text{count}(X_1=1, Y=\text{Sad})+1}{N'}$

Y	Laplace Est. $P(Y=y)$
Like	$\frac{\text{count}(Y=\text{like})+2}{N'}$
Love	$\frac{\text{count}(Y=\text{love})+2}{N'}$
Haha	$\frac{\text{count}(Y=\text{Haha})+2}{N'}$
Sad	$\frac{\text{count}(Y=\text{Sad})+2}{N'}$

After we get the 4 different estimation based on different Y we can just select the $Y = y_i$ makes the ~~the~~ maximum value.

10 Logistic Vision Test [20 points]

You decide that the vision tests given by eye doctors could have more precise results if we used an approach inspired by logistic regression. In a vision test a user looks at a letter with a particular font size and either correctly guesses the letter or incorrectly guesses the letter.

You assume that the probability that a particular patient is able to guess a letter correctly is:

$$p = \sigma(\theta - f)$$

Where θ is the user's vision score and f is the font size of the letter.

Explain how you could estimate a user's vision score (θ) based on their 20 responses $(f^{(1)}, y^{(1)}) \dots (f^{(20)}, y^{(20)})$, where $y^{(i)}$ is an indicator variable for whether the user correctly identified the ith letter and $f^{(i)}$ is the font size of the ith letter. Solve for any and all partial derivatives required by your answer.

We can use the 20 samples to get estimation of θ

$$LL(\theta) = \sum_{i=1}^{20} y^{(i)} \log \sigma(\theta - f^{(i)}) + (1 - y^{(i)}) \log(1 - \sigma(\theta - f^{(i)}))$$

$$\frac{\partial LL(\theta)}{\partial \theta} = \sum_{i=1}^{20} [y^{(i)} - \sigma(\theta - f^{(i)})]$$

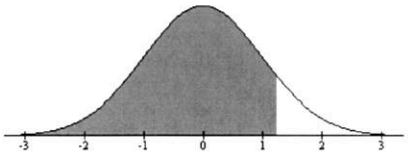
to get the global maxima of $LL(\theta)$, we can repeat the following calculation with a small step η

$$\theta^{new} = \theta^{old} + \eta \cdot \frac{\partial LL(\theta^{old})}{\partial \theta^{old}}$$

we can use $\theta^{(0)} = 0$ as a start point

Standard Normal Table

Note: An entry in the table is the area under the curve to the left of z, $P(Z \leq z) = \Phi(z)$



z	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09
0.0	0.5000	0.5040	0.5080	0.5120	0.5160	0.5199	0.5239	0.5279	0.5319	0.5359
0.1	0.5398	0.5438	0.5478	0.5517	0.5557	0.5596	0.5636	0.5675	0.5714	0.5753
0.2	0.5793	0.5832	0.5871	0.5910	0.5948	0.5987	0.6026	0.6064	0.6103	0.6141
0.3	0.6179	0.6217	0.6255	0.6293	0.6331	0.6368	0.6406	0.6443	0.6480	0.6517
0.4	0.6554	0.6591	0.6628	0.6664	0.6700	0.6736	0.6772	0.6808	0.6844	0.6879
0.5	0.6915	0.6950	0.6985	0.7019	0.7054	0.7088	0.7123	0.7157	0.7190	0.7224
0.6	0.7257	0.7291	0.7324	0.7357	0.7389	0.7422	0.7454	0.7486	0.7517	0.7549
0.7	0.7580	0.7611	0.7642	0.7673	0.7703	0.7734	0.7764	0.7793	0.7823	0.7852
0.8	0.7881	0.7910	0.7939	0.7967	0.7995	0.8023	0.8051	0.8078	0.8106	0.8133
0.9	0.8159	0.8186	0.8212	0.8238	0.8264	0.8289	0.8315	0.8340	0.8365	0.8389
1.0	0.8413	0.8438	0.8461	0.8485	0.8508	0.8531	0.8554	0.8577	0.8599	0.8621
1.1	0.8643	0.8665	0.8686	0.8708	0.8729	0.8749	0.8770	0.8790	0.8810	0.8830
1.2	0.8849	0.8869	0.8888	0.8906	0.8925	0.8943	0.8962	0.8980	0.8997	0.9015
1.3	0.9032	0.9049	0.9066	0.9082	0.9099	0.9115	0.9131	0.9147	0.9162	0.9177
1.4	0.9192	0.9207	0.9222	0.9236	0.9251	0.9265	0.9279	0.9292	0.9306	0.9319
1.5	0.9332	0.9345	0.9357	0.9370	0.9382	0.9394	0.9406	0.9418	0.9429	0.9441
1.6	0.9452	0.9463	0.9474	0.9484	0.9495	0.9505	0.9515	0.9525	0.9535	0.9545
1.7	0.9554	0.9564	0.9573	0.9582	0.9591	0.9599	0.9608	0.9616	0.9625	0.9633
1.8	0.9641	0.9649	0.9656	0.9664	0.9671	0.9678	0.9686	0.9693	0.9699	0.9706
1.9	0.9713	0.9719	0.9726	0.9732	0.9738	0.9744	0.9750	0.9756	0.9761	0.9767
2.0	0.9772	0.9778	0.9783	0.9788	0.9793	0.9798	0.9803	0.9808	0.9812	0.9817
2.1	0.9821	0.9826	0.9830	0.9834	0.9838	0.9842	0.9846	0.9850	0.9854	0.9857
2.2	0.9861	0.9864	0.9868	0.9871	0.9875	0.9878	0.9881	0.9884	0.9887	0.9890
2.3	0.9893	0.9896	0.9898	0.9901	0.9904	0.9906	0.9909	0.9911	0.9913	0.9916
2.4	0.9918	0.9920	0.9922	0.9925	0.9927	0.9929	0.9931	0.9932	0.9934	0.9936
2.5	0.9938	0.9940	0.9941	0.9943	0.9945	0.9946	0.9948	0.9949	0.9951	0.9952
2.6	0.9953	0.9955	0.9956	0.9957	0.9959	0.9960	0.9961	0.9962	0.9963	0.9964
2.7	0.9965	0.9966	0.9967	0.9968	0.9969	0.9970	0.9971	0.9972	0.9973	0.9974
2.8	0.9974	0.9975	0.9976	0.9977	0.9977	0.9978	0.9979	0.9979	0.9980	0.9981
2.9	0.9981	0.9982	0.9982	0.9983	0.9984	0.9984	0.9985	0.9985	0.9986	0.9986
3.0	0.9987	0.9987	0.9987	0.9988	0.9988	0.9989	0.9989	0.9989	0.9990	0.9990
3.1	0.9990	0.9991	0.9991	0.9991	0.9992	0.9992	0.9992	0.9992	0.9993	0.9993
3.2	0.9993	0.9993	0.9994	0.9994	0.9994	0.9994	0.9994	0.9995	0.9995	0.9995
3.3	0.9995	0.9995	0.9995	0.9996	0.9996	0.9996	0.9996	0.9996	0.9996	0.9997
3.4	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9998
3.5	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998