Chris Piech
CS109

Fall Quarter
Dec 8th, 2018

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. The penultimate page has some common random variables. 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, integrals 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.

LELAND STANFORD JUNIOR UNIVERSITY
DIE LUFT DER FREIHEIT WEHT

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: ____

Family Name (print): ____________________________________

Given Name (print): ____________________________________

Email (preferably your gradescope email): ____________________________

1 Ultimate Frisbee Flip [X points]

To start game of ultimate frisbee both teams flip a frisbee. Each frisbee either lands "heads" (facing down) or "tails" (facing up). Your team gets to start if the two frisbees are facing the same direction (either the frisbees are both heads or both tails). Your opponent gets to start if they are facing in different directions.

You don't know the value of p, the probability a frisbee lands heads. You do assume that both frisbees have the same value for p and that each flip is independent.

a. What is the probability that your team starts? Express your answer in terms of p.

Answer. Let S be the event that you start.

$$\begin{aligned} P(S) &= P(\text{Both heads}) + P(\text{Both tails}) && \text{Mutually exclusive events} \\ &= p^2 + (1-p)^2 && \text{Independent events} \end{aligned}$$

b. What is the probability that your opponent starts? Express your answer in terms of p.

Answer.

$$\begin{aligned} P(S^C) &= P(HT) + P(TH) && \text{Mutually exclusive events} \\ &= p(1-p) + p(1-p) && \text{Independent events} \\ &= 2p(1-p) && \text{Maths} \end{aligned}$$

c. If we assume that $p = 0.5$, what is the probability that your opponent starts? Give a number.

Answer.

$$\begin{aligned} P(S^C) &= 2p(1-p) && \text{Answer to part (b)} \\ &= 2(0.5)(0.5) = 0.5 && \text{Assuming that } p = 0.5 \end{aligned}$$

d. Let X be a random variable which represents your uncertainty about p. Your prior belief is that $X \sim \text{Beta}(a = 2, b = 2)$. To get a better estimate of X you flip a frisbee 10 times and it comes up heads 6 times. What is your updated belief?

Answer.

$$X \sim \text{Beta}(a = 8, b = 6)$$

e. Using this updated belief about p, give an expression for the probability that you start the game.

Answer.

$$
\begin{aligned}
P(\mathrm{S}) &= \int_{x=0}^{1} P(S, X = x) && \text{Marginalization} \\
&= \int_{x=0}^{1} P(S|X = x) f(X = x) && \text{Chain rule} \\
&= \frac{1}{B(8,6)} \int_{x=0}^{1} P(S|X = x) \cdot x^7 (1-x)^5 && \text{Beta CDF} \\
&= \frac{1}{B(8,6)} \int_{x=0}^{1} \left[x^2 + (1-x)^2\right] \cdot x^7 (1-x)^5 && \text{From part (a)} \\
&= \frac{1}{B(8,6)} \int_{x=0}^{1} x^9 (1-x)^5 + x^7 (1-x)^7 && \text{Algebra. The integral is nasty :)} \\
&= 0.543 && \text{Incase you are curious}
\end{aligned}
$$

2 Phone Compass [X points]

Keeping track of a phone's compass direction is an interesting problem. To represent our uncertainty about the compass angle, we are going to need a random variable *type* made for angle values. There is a special "Circle Normal Distribution" that takes on radian angle values in the range $[0, 2\pi]$.

A Circle Normal has two parameters: μ, the mean angle, and τ, a parameter which changes the distribution's spread. If $X \sim \text{CircleNorm}(\mu, \tau)$, then X has probability density function:

$$f(X = x) = \frac{K(\tau)}{2\pi} e^{\tau \cos(x-\mu)} \qquad \text{Where } K(\tau) \text{ is a normalizing constant that depends on } \tau$$

Cosine review:
cos is a function applied to angles measured in radians. An angle in radians can take on values in the range 0 to 2π, where 2π radians corresponds to 360 degrees. Here are a few identities for cos:

$$\cos(0) = \cos(2\pi) = 1 \qquad \frac{\partial}{\partial x}\cos(x) = -\sin(x)$$

$$\cos(\pi) = -1 \qquad \int \cos(x) = \sin(x)$$

a. Let $X \sim \text{CircleNorm}(\mu = 0, \tau = 0)$. Write a simplified version of the density function for X:

Answer.

$$\begin{aligned} f(X = x) &= \frac{K(\tau)}{2\pi} e^{\tau \cos(x-\mu)} && \text{The CircNorm PDF} \\ &= \frac{K(\tau)}{2\pi} e^{0 \cos(x-0)} && \text{Since } \mu = 0\pi, \tau = 0 \\ &= \frac{K(\tau)}{2\pi} && \text{power of 0} \\ &= \frac{1}{2\pi} && \text{Worth a very small amount of points} \end{aligned}$$

b. Our phone has a belief about the direction X of the compass: $X \sim \text{CircleNorm}(\mu = 0, \tau = 0.5)$. How much more likely is the phone to be facing 0 degrees than π degrees?

Answer.

$$\begin{aligned} \frac{f(X = 0)}{f(X = \pi)} &= \frac{\frac{K(\tau)}{2\pi} e^{\tau \cos(x-\mu)}}{\frac{K(\tau)}{2\pi} e^{\tau \cos(x-\mu)}} && \text{CircNorm PDF} \\ &= \frac{\frac{K(0.5)}{2\pi} e^{0.5 \cos(0)}}{\frac{K(0.5)}{2\pi} e^{0.5 \cos(\pi)}} && \text{Substitute params and x} \\ &= \frac{e^{0.5}}{e^{-0.5}} = e && \text{That is so cool} \end{aligned}$$

c. The smartphone “magnetometer” is a noisy sensor. The noise Y is distributed as a Circle Normal with $\mu = 0$ and unknown τ. $Y \sim \text{CircleNorm}(\mu = 0, \tau)$. We are going to estimate the τ parameter of Y using Maximum Likelihood Estimation based on data. We have n IID samples of **magnetometer noise** (Y): $[y_1, y_2, \dots y_n]$.

Compute the derivative of the log likelihood function with respect to τ. Write a few sentences to explain how you could use the derivative in order to chose the most likely value of τ.

Note: you do not need to calculate $K(\tau)$ nor $\frac{\partial K(\tau)}{\partial \tau}$. You may leave either term in your answer.

Answer. :
Likelihood

$$\begin{aligned}
L(\tau) &= \prod_{i=1}^{n} f(Y_i = y_i) && \text{Basic likelihood of IID data} \\
&= \prod_{i=1}^{n} \frac{K(\tau)}{2\pi} \cdot e^{\tau \cos(y_i)} && \text{Substitute in the CircNorm PDF} \\
&= \prod_{i=1}^{n} K(\tau) \cdot e^{\tau \cos(y_i)} && \text{Constants don't matter}
\end{aligned}$$

LogLikelihood

$$\begin{aligned}
LL(\tau) &= \log \left[\prod_{i=1}^{n} K(\tau) \cdot e^{\tau \cos(y_i)} \right] && \text{Take log of likelihood} \\
&= \sum_{i=1}^{n} \log \left[K(\tau) \cdot e^{\tau \cos(y_i)} \right] && \text{Log of prod} \\
&= \sum_{i=1}^{n} \log[K(\tau)] + \sum_{i=1}^{n} \tau \cos(y_i) && \text{Log of exp}
\end{aligned}$$

Derivative of LogLikelihood

$$\begin{aligned}
\frac{\partial}{\partial \tau} LL(\tau) &= \frac{\partial}{\partial \tau} \sum_{i=1}^{n} \log[K(\tau)] + \frac{\partial}{\partial \tau} \sum_{i=1}^{n} \tau \cos(y_i) \\
&= \frac{\partial}{\partial \tau} \sum_{i=1}^{n} \log[K(\tau)] + \sum_{i=1}^{n} \cos(y_i) \\
&= \sum_{i=1}^{n} \frac{1}{K(\tau)} \frac{\partial K(\tau)}{\partial \tau} + \sum_{i=1}^{n} \cos(y_i)
\end{aligned}$$

We can use gradient ascent to chose the value of τ that maximizes LL. Gradient ascent requires us to be able to calculate $\frac{\partial LL(\tau)}{\partial \tau}$.

3 Modelling Disease Spread [X points]

Our ability to fight contageous diseases depends on our ability to model them. One person is exposed to palo-alto-virus. The method below returns the number of individuals who will get infected.

```
# Get number of people infected by one individual
def numInfected():
   # most people are immune to palo-alto-virus
   immune = bernoulli(p = 0.99)
   if immune: return 0

   # people who are not immune, spread the disease far
   spread = 0

   # they make contact with k people (up to 100)
   k = binomial(n = 100, p = 0.25)
   for i in range(k)
     spread += numInfected()

   # total infections should include this individual
   return spread + 1
```

What is the expected return value of numInfected?

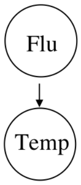
Answer.

$$
\begin{aligned}
E[R] &= \frac{1}{100} \cdot E[R|I] = \frac{1}{100} \cdot (1 + E[S]) \\
E[S] &= \sum_{k=0}^{100} E[S|K=k]P(K=k) \\
&= \sum_{k=0}^{100} \Big(E[R] + \cdots + E[R]\Big) P(K=k) \\
&= \sum_{k=0}^{100} kE[R]P(K=k) \\
&= E[R] \sum_{k=0}^{10} kP(K=k) = E[R] \cdot E[K] = \frac{100E[R]}{4} \\
E[R] &= \frac{1}{100}(1 + \frac{100E[R]}{4}) \\
&= \frac{1}{100} + \frac{1}{4}E[R] \\
\frac{3}{4}E[R] &= \frac{1}{100} \\
E[R] &= \frac{4}{300}
\end{aligned}
$$

4 Cheeky Joint Sampling [X points]

In class we observed that Joint Sampling didn't work well when we asked probability questions that conditioned on rare events. Joint Sampling is especially problematic if you are conditioning on a continuous variable .

To get around our inability to condition on continuous random variables we introduce Cheeky Joint Sampling. Cheeky Joint Sampling is just like regular Joint Sampling, except we consider a continuous random variable assignment to "match" a conditioning event if the values **are within 0.1** of each other. We can explore Cheeky Joint Sampling with this very small Bayesian Network:



P($F = 1$) = 0.01

If $F = 1$ then $T \sim \mathrm{N}(\mu = 101, \sigma^2 = 4)$
If $F = 0$ then $T \sim \mathrm{N}(\mu = 98, \sigma^2 = 4)$

Flu is a Bernoulli random variable with $p = 0.01$ and Temp is a Normal random variable with μ that changes based on whether the person has a flu.

As an example of the cheeky difference: say we are calculating $P(\text{Flu} = 1|\text{Temp} = 100)$ and we generate a sample with Temp = 100.0001. In the Cheeky algorithm that sample would count as matching the conditioned event. In regular Joint Sampling it would not since $100.0001 \neq 100$.

a. Warmup: What is the probability that a person with the flu has a fever greater than 103? You must give a numerical answer for full credit.

Answer.

$$\begin{aligned} P(T > 103) &= 1 - F_T(103) = 1 - \Phi\Big(\frac{x-\mu}{\sigma}\Big) \\ &= 1 - \Phi\Big(\frac{103-101}{2}\Big) = 1 - \Phi(1) = 1 - 0.8413 = 0.1587 \end{aligned}$$

b. For this simple model, we can calculate exact probabilities. What is the probability that someone has a flu given a temperature of 100?

Answer.

$$\begin{aligned} P(F = 1|T = 100) &= \frac{f(T = 100|F = 1)P(F = 1)}{f(T = 100)} \\ &= \frac{f(T = 100|F = 1)P(F = 1)}{f(T = 100|F = 1)P(F = 1) + f(T = 100|F = 0)P(F = 0)} \\ &= \frac{e^{-\frac{(100-101)^2}{8}} 0.01}{e^{-\frac{(100-101)^2}{8}} 0.01 + e^{-\frac{(100-98)^2}{8}} 0.99} \\ &= \frac{e^{-\frac{0.01}{8}}}{e^{-\frac{0.01}{8}} + e^{-\frac{4 \cdot 0.99}{8}}} \end{aligned}$$

c. Write code to calculate the probability that someone has a flu given a fever of 100 using Cheeky Joint Sampling with $N = 10000$ samples.

Answer. :

```
def getCondProb():
  countFlu100 = 0
  count100 = 0
  for i in range(10000):
    [f, t] = sample()
    if t > 99.9 and t < 100.1:
      count101 += 1
      if f == 1:
        countFlu100 += 1
  return countFlu100 / count100

def sample():
  f = bern(0.01)
  if f == 0:
    t = norm(98, 4)
  else:
    t = norm(101, 4)
  return [f, t]
```

d. What is the probability that out of 10,000 iterations of Cheeky Joint Sampling for our model, there are < 20 samples that have a temperature within 0.1 degrees of 100 degrees? Use an approximation so that the probability is computable. You don't need a numerical answer.

Answer. :
Let E be the event that a random sample has a temperature in range:

$$\begin{aligned} P(E) &= P(E|F=1)P(F=1) + P(E|F=0)P(F=0) \\ &= \left[\Phi\left(\frac{100.1-101}{2}\right) - \Phi\left(\frac{99.9-101}{2}\right)\right]0.01 + \left[\Phi\left(\frac{100.1-98}{2}\right) - \Phi\left(\frac{99.9-98}{2}\right)\right]0.99 \end{aligned}$$

Let $p = P(E)$.
Let K be a random variable which represents the number of in range samples.
$K \sim \text{Bin}(n = 10000, p)$
Let X be a normal approximation of K.
$X \sim N(\mu = 10000p, \sigma^2 = 10000p(1-p))$

$$\begin{aligned} P(K < 20) &\approx P(X < 19.5) \\ &= F_X(19.5) = \Phi\left(\frac{19.5 - 10000p}{\sqrt{10000p(1-p)}}\right) \end{aligned}$$

5 Generative Algorithm [X points]

On your homework you trained a Naive Bayes and Logistic Regression algorithm to *classify*. In this problem we are going to explore how we could use a trained model to perform a different task: *generate* a new example. We are going to do so in a two part process. First we will come up with an expression for the joint probability of $\mathbf{X}$ and Y. Second we will generate a sample from that joint.

For all parts of this question you may assume that we have already trained the algorithms. As in your homework, all n features are binary, $X_i \in \{0,1\}$, and so is the label, $Y \in \{0,1\}$.

a. Naive Bayes: You have a trained Naive Bayes classifier just like in your homework. You have learned all of the Naive Bayes parameters which you can access using the following methods:

Method	**Description**
`probXiGivenY(i, x, y)`	Returns the learned value for $P(X_i = x \mid Y = y)$
`probY(y)`	Returns the learned value for $P(Y = y)$

i Write a math expression for the probability of a given feature, label combination $(\mathbf{x}, y)$. In other words, calculate the “joint”: $P(\mathbf{X} = \mathbf{x}, Y = y)$. You may make the Naïve Bayes assumption and you may use any of the learned probabilities.

Answer.

$$\begin{aligned} P(\mathbf{X} = \mathbf{x}, Y = y) &= P(\mathbf{X} = \mathbf{x} \mid Y = y) P(Y = y) && \text{Chain rule} \\ &= P(Y = y) \prod_i P(X_i = x_i \mid Y = y) && \text{Nav̈e Bayes assumption} \end{aligned}$$

ii Write a function `sample` which returns a randomly generated sample of $\mathbf{x}, y$ from the joint probability distribution. Your code can be psuedo-code and you should use a method `bern(p)` that samples from a Bernoulli with probability p:

Answer. :

```
def sample():
  y = bern(probY(1))
  x = []
  for i in range (1, n+1):
    p = probXiGivenY(i, 1, y)
    x[i] = bern(p)
  return x, y
```

b. Logistic Regression: You have a trained Logistic Regression algorithm just like in your homework. Specifically assume you have learned the values for θ (for simplicity, assume there is no intercept term).

i Write a math expression for the probability of a given feature/label combination $(\mathbf{x}, y)$. In other words, calculate the "joint": $P(\mathbf{X}=\mathbf{x}, Y=y)$. You will need to make an assumption about the distribution for the features $\mathbf{X}$. Assume each feature is independent and distributed as a bernoulli $X_i \sim \text{Bern}(p=0.5)$.

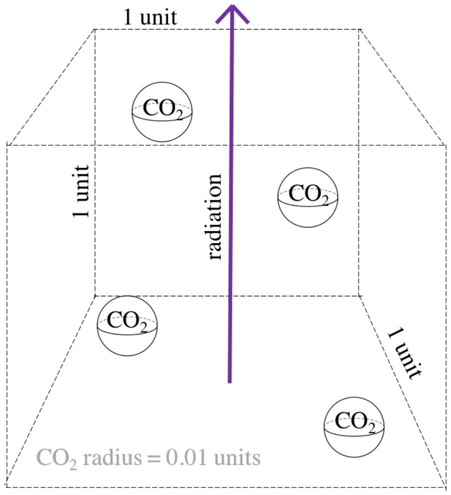
Answer.

$$\begin{aligned}
P(\mathbf{X}=\mathbf{x}, Y=y) &= P(Y=y|\mathbf{X}=\mathbf{x})P(\mathbf{X}=\mathbf{x}) \\
&= P(Y=y|\mathbf{X}=\mathbf{x})\prod_i P(X_i=x_i) \\
&= P(Y=y|\mathbf{X}=\mathbf{x})({}^1\!/\!{}_2)^n \\
&= \sigma(\theta^T x)({}^1\!/\!{}_2)^n
\end{aligned}$$

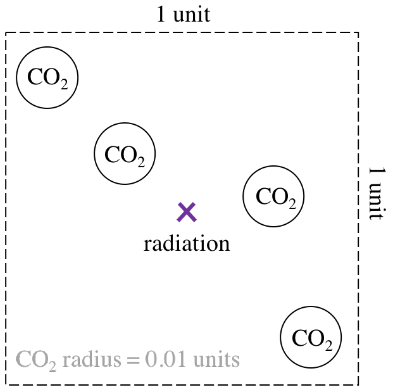
6 Greenhouse Effect Exploration [X Points]

As you probably know, atmospheric CO_2 makes the world warmer because of the *Greenhouse Effect*. Radiation leaving the earth hits CO_2 molecules in the atmosphere.

Imagine a ray of radiation goes straight up through the center of a cube of atmosphere. The cube has 1 unit sides and randomly placed CO_2 molecules:



Side View



Top View

We are going to make the following assumptions:

- CO_2 molecules are spheres with radius 0.01 units. From the top view they look like circles.
- The radiation ray always travels straight up.
- The location of each CO_2 molecule is independent of the others. CO_2 spheres may overlap.
- Each CO_2 sphere is equally likely to be anywhere in the cube. In other words, the location of a CO_2 sphere is defined by the X, Y, Z of its center point. X, Y, Z are independent and they are all Uniform($\alpha = 0, \beta = 1$).
- The radiation ray is so fast that we can treat every molecule as stationary.
- Recall that for a circle, area $= \pi r^2$ where r is the circle radius.

Using this simple model answer the following questions:

a. Imagine there is exactly one CO_2 molecule in the cube. What is the probability that it intersects the radiation ray? *Hint:* think from the "Top View" perspective!

Answer. The molecule intersects the ray if its top view profile intersects the radiation point. To do so, the center point of the CO_2 molecule (in the X, Y plane) must be inside a circle with radius 0.01 units centered around the ray point. Given that the X, Y locations are all equally likeliy:

$$\begin{aligned} P(\text{Collision}) &= \frac{|\text{Event Space}|}{|\text{Sample Space}|} \\ &= \frac{\pi(0.01)^2}{1^2} = \pi(0.01)^2 \end{aligned}$$

b. The number of CO_2 molecules in the cube $K \sim \text{Poi}(\lambda = 100)$. What is the standard deviation of the number of CO_2 molecules?

Answer.

$$\text{Std}(X) = \sqrt{\text{Var}(X)} = \sqrt{\lambda} = \sqrt{100} = 10$$

c. Imagine there are exactly 100 CO_2 molecules in the cube. Write an expression for the probability that none of the molecules intersect the radiation. Let p be your answer to part (a).

Answer.

$$P(\text{No Collision}) = (1-p)^{100} = \Big[1 - \pi(0.01)^2\Big]^{100}$$

d. Again imagine there are exactly 100 CO_2 molecules in the cube. The radiation ray stops when it hits a CO_2 molecule. Provide a cumulative density function for random variable X: the length of the radiation ray. Assume that the length of the ray is equal to the Z value (height of the center) of the first molecule the ray hits.

Answer. :
Let T be the number of molecules the ray collides with. $T \sim \text{Bin}(n = 100, p)$
Let Z_i be the z location of the ith colliding molecule.

$$\begin{aligned}
P(X < k|T = i) &= 1 - P(X > k|T = i) \\
&= 1 - \Big[P(Z_i > k)\Big]^i \\
&= 1 - \Big[1 - k\Big]^i \\
P(T = i) &= \binom{100}{i} p^i (1-p)^{100-i} && \text{Binomial PMF} \\
P(X < k) &= \sum_{i=1}^{100} P(X < k, T = i) && \text{Marginalization} \\
&= \sum_{i=1}^{100} P(X < k|T = i)P(T = i) && \text{Chain Rule} \\
&= \sum_{i=1}^{100} \left[1 - (1-k)^i\right] \binom{100}{i} p^i (1-p)^{100-i} && \text{Substitute}
\end{aligned}$$

Answer. (alternate approach): Let $Z_i \sim Uni(0,1)$ be the height of molecule i. Note that the probability of a single molecule blocking the ray before it has reached length x is:

$$q_x = \int_{z=0}^{x} f_Z(z) P(intersects|Z = z) dz = \int_{z=0}^{x} 1 p dz = px$$

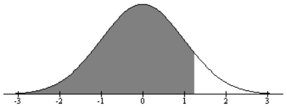
(You can also think of q_x as the probabilty of the center of the molecule being inside a cylinder of height x and radius 0.01.) Note that the probability of a single molecule NOT obstructing the ray before height x is $1 - q_x$. We then have:

$$P(X < x) = 1 - P(X > z) = 1 - (1 - q_x)^{100} = 1 - (1 - px)^{100}$$

This is (despite appearances) the same as the previous answer.

Standard Normal Table

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