Question 1
(a)
This mechanism is differentially private. This is because a randomized response is produced through the coin flip and based on the outcome of the coin flip, the respondents will answer yes/no to the potentially incriminating question. In this case, the choice governed by the coin flip is between responding honestly and responding “yes” (which either may be the truth or a lie). Privacy comes from the uncertainty of how to interpret a reported value. For example, if the individual answers “yes”, we won’t know where this “yes” came from as there are two possibilities. Additionally, we don’t know if the “yes” is the truth or not.

Let’s denote the
Truth = Yes by P,
Response = Yes by A
Coin = Heads by H
Coin = Tails by T

First, let’s get the probability of Yes given that the individual is telling the truth. This would be either when the individual flipped a tail or a head. In these two cases, the answer to the question is yes while the truth is also yes.
P(A | P) = P(T) + P(H) = ½ + ½ = 1.
Now, let’s get the probability of Yes given that the individual is not telling the truth. This would be only when the individual flipped a head. In this case, the answer to the question is yes, while the truth is no.
P(A | -P) = P(H) = ½.
Now, let’s represent P(A | P) in terms of P(A | -P) to get our ratio: P(A | P) = 2 P(A | -P).
The ratio is 2, so from the differential privacy equation we have:
$e^{epsilon}$ = 2
epsilon = ln(2)
So this version of a randomized response is ln(2)-differentially private

(b)
Let’s denote the
Truth = Yes by P,
Response = Yes by A
D_1 = 1,2,3,4,5,6
D_2 = 1,2,3,4,5,6

First, let’s get the probability of Yes given that the individual is telling the truth. This would be either when the first roll is 1,2,3 or the first roll is 4,5,6 and the second roll is 1,2. In these two cases, the answer to the question is yes while the truth is also yes.
P(A | P) = P(D_1=1,2,3) + P(D_1=4,5,6, D_2=1,2) = ½ + ½ x ⅓ = ⅔.
Now, let’s get the probability of Yes given that the individual is not telling the truth. This would be only when the first roll is 4,5,6 and the second roll is 1,2. In this case, the answer to the question is yes, while the truth is no.

P(A | -P) = P(D_1=4,5,6, D_2=1,2) = ½ x ⅓ = ⅙.
Now, let’s represent P(A | P) in terms of P(A | -P) to get our ratio: P(A | P) = 4 P(A | -P).
The ratio is 2, so from the differential privacy equation we have:
$e^{epsilon}$ = 4
epsilon = ln(4)

Question 2

- Mean difference: $m1 = p(y+|\ gM) - p(y+|\ gF \vee gX)$
- Disparate impact: $m2 = \ p(y+|\ gF \vee gX) \ / \ p(y+|\ gM)$
- Elift ratio: $m3 = \ p(y+|\ gM \wedge\ shigh) \ / \ p(y+|\ shigh)$

(a)
To compute the mean difference (m1) and the disparate impact (m2) while preserving differential privacy, we need to design the queries and allocate the privacy budget accordingly. The queries needed are the following:

Q1: y^+, g^M under ε1 = 0.5
- SELECT COUNT(*) FROM D WHERE admission = 'yes' AND gender = ‘M’;
- This query counts the number of positive outcomes (admissions) in males.

Q2: y^+, g^F/g^X under ε2 = 0.5
- SELECT COUNT(*) FROM D WHERE admission = 'yes' AND (gender = 'F' OR gender = 'X');
- This query counts the number of positive outcomes (admissions) in either female or non-binary.

Q3: g^M under ε3 = 0.5
- SELECT COUNT(*) FROM D WHERE gender = ‘M’;
- This query counts the number of males.

Q4: g^F/g^X under ε4 = 0.5
- SELECT COUNT(*) FROM D WHERE gender = 'F' OR gender = 'X';
- This query counts the number of either female or non-binary.

ε = max(ε1, ε2) + max(ε3, ε4) = 1

m1 is computed as Q1/Q3 - Q2/Q4
m2 is computed as (Q2/Q4) / (Q1/Q3)

Since we're using the entire privacy budget (ε = 1) on m1 and m2, we can equally allocate 0.5 on each m using sequential composition. Within each m, we use parallel composition since the two queries accessed within each m are disjoint (one is male while the other is female).

(b)

To compute the mean difference (m1), the disparate impact (m2), and the Elift ratio (m3) while preserving differential privacy, we need to design the queries and allocate the privacy budget accordingly. The queries needed are the following:

Q1: y^+, g^M under ε1 = 0.33

- SELECT COUNT(*) FROM D WHERE admission = 'yes' AND gender = ‘M’;
- This query counts the number of positive outcomes (admissions) in males.

Q2: y^+, g^F/g^X under ε2 = 0.33

- SELECT COUNT(*) FROM D WHERE admission = 'yes' AND (gender = 'F' OR gender = 'X');
- This query counts the number of positive outcomes (admissions) in either female or non-binary.

Q3: g^M under ε3 = 0.33

- SELECT COUNT(*) FROM D WHERE gender = ‘M’;
- This query counts the number of males.

Q4: g^F/g^X under ε4 = 0.33

- SELECT COUNT(*) FROM D WHERE gender = 'F' OR gender = 'X';
- This query counts the number of either female or non-binary.

Q5: y^+, g^M, s^{high} under ε5 = 0.17

- SELECT COUNT(*) FROM D WHERE admission = 'yes' AND gender = 'M' AND SAT = 'high';
- This query counts the number of positive outcomes (admissions) in males with a high SAT.

Q6: y^+, s^{high} under ε6 = 0.17

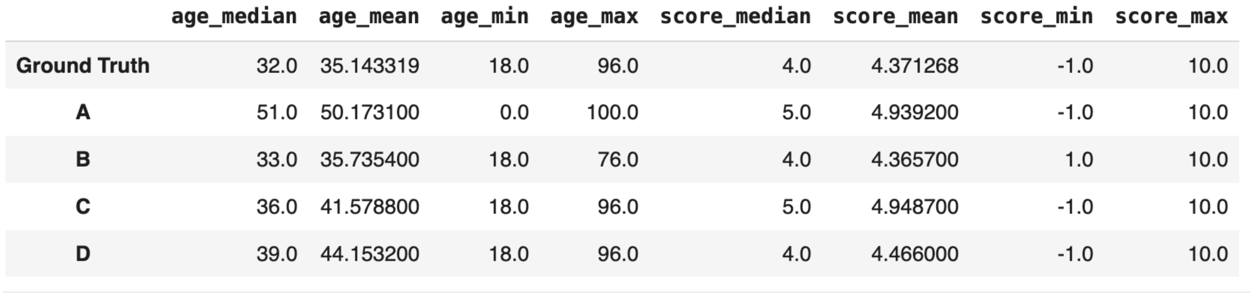
- SELECT COUNT(*) FROM D WHERE admission = 'yes' AND GPA = 'high';
- This query counts the number of positive outcomes (admissions) with a high SAT.

ε = max(ε1, ε2) + max(ε3, ε4) + ε5 + ε6 = 0.33 + 0.33 + 0.17 + 0.17 = 1

m1 is computed as Q1/Q3 - Q2/Q4
m2 is computed as (Q2/Q4) / (Q1/Q3)
m3 is computed as Q5/Q6

Since we're using the entire privacy budget (ε = 1) on m1, m2, and m3, we can equally allocate 0.33 on each m using sequential composition. Within m1 and m2, we use parallel composition since the two queries accessed within each m are disjoint (one is male while the other is female). Within m3, we use sequential composition because the two queries used (Q5 and Q6) are not disjoint. They overlap. Another reason of such allocation is that in both computations of m1 and m2 Q1234 are used twice as many times as Q56 are in m3. Therefore, it makes sense to assign ε1234 double the value of ε56.

Question 3
(a)
Q1

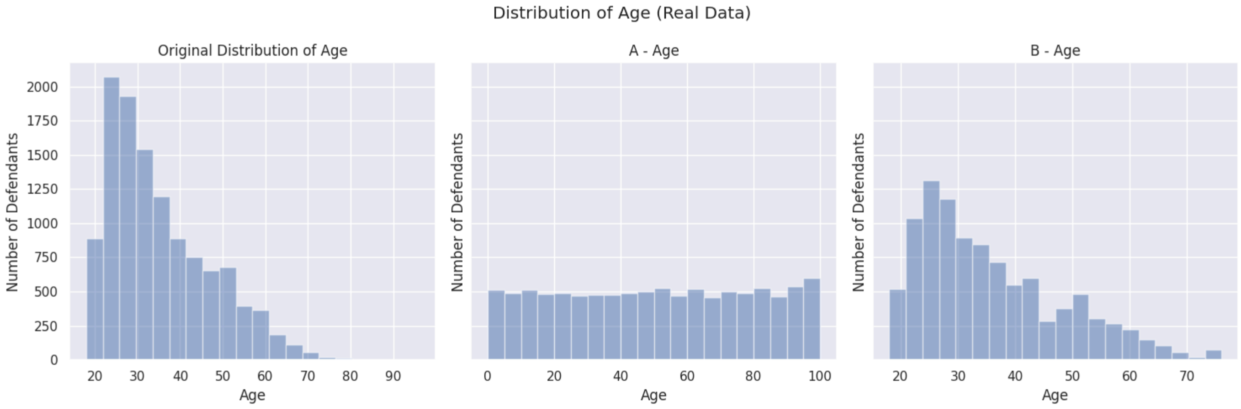


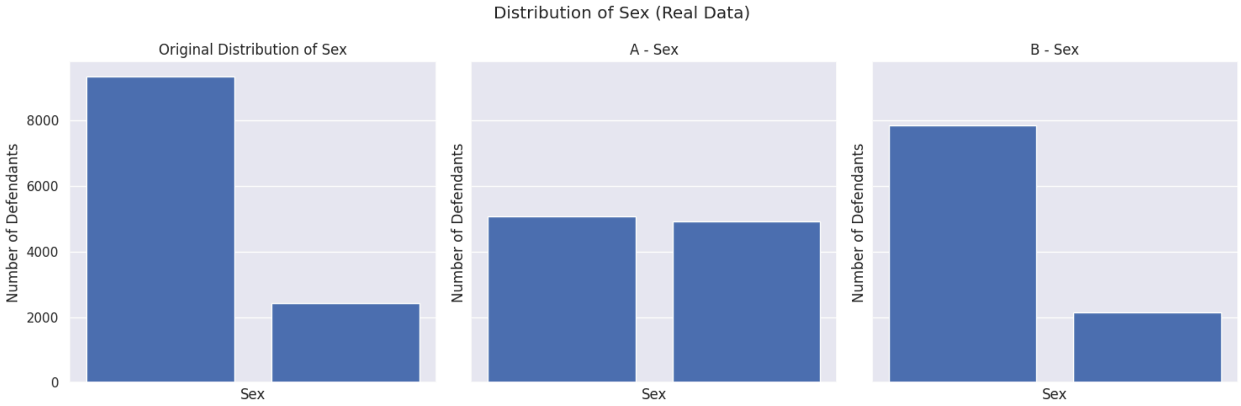
	age_median	age_mean	age_min	age_max	score_median	score_mean	score_min	score_max
Ground Truth	32.0	35.143319	18.0	96.0	4.0	4.371268	-1.0	10.0
A	51.0	50.173100	0.0	100.0	5.0	4.939200	-1.0	10.0
B	33.0	35.735400	18.0	76.0	4.0	4.365700	1.0	10.0
C	36.0	41.578800	18.0	96.0	5.0	4.948700	-1.0	10.0
D	39.0	44.153200	18.0	96.0	4.0	4.466000	-1.0	10.0

Age Comparison:
Mode A (random mode) is the least accurate as its median, mean, min and max values are the furthest away from ground truth values of age. Its minimum value is 0 (compared to 18) and its maximum value is 100 (compared to 96).
Mode B (Independent Attribute Mode) has the closest median (33.0) and mean (35.74) to the ground truth (median: 32.0, mean: 35.14) and its minimum value also matches (18). However, the maximum value of the real data is not preserved. In mode B, it is 76 and in the real data it is 96. This mismatch between max values may be because of how the values are binned (default 20 bins). This grouping can result in loss of precision and detail, which can impact the maximum value that can be represented.
Mode C and D are somewhat accurate as their medians and means are not too far off but also not too close to those of the ground truth. The min and max for mode C and D match. Mode D is slightly less accurate than mode C and the difference may be because of the difference in their k parameter.
There is a substantial difference in the accuracy of method A compared to mode B, C, and D because, in random mode, we draw from uniform distribution. The important parameters when using random mode are seed, minimum and maximum values. But, in the generate_data_A function when calling generate_dataset_in_random_mode, the minimum and maximum values were not specified or set to match the real data's age distribution, so it was learned to be 0 to 100. Since in a uniform distribution, the probability of all possible outcomes are the same and our outcomes range from 0 to 100, it makes sense why the mean and median are around 50 (median: 51.0, mean: 50.173). Additionally, 0 is not a valid age, which also makes the random mode synthetic data less accurate.

Score Comparison:
All the modes have a similar accuracy for score as the mean and median differences from the ground truth stay within 1.0 and the min and max values are the same as ground truth for all but 1 (mode B min). Mode B has the most accurate mean and median. However, its minimum value is different from the ground truth 1, not -1. This could be again because of the default bins being 20, which impact the precision of the minimum and maximum values.

Q2



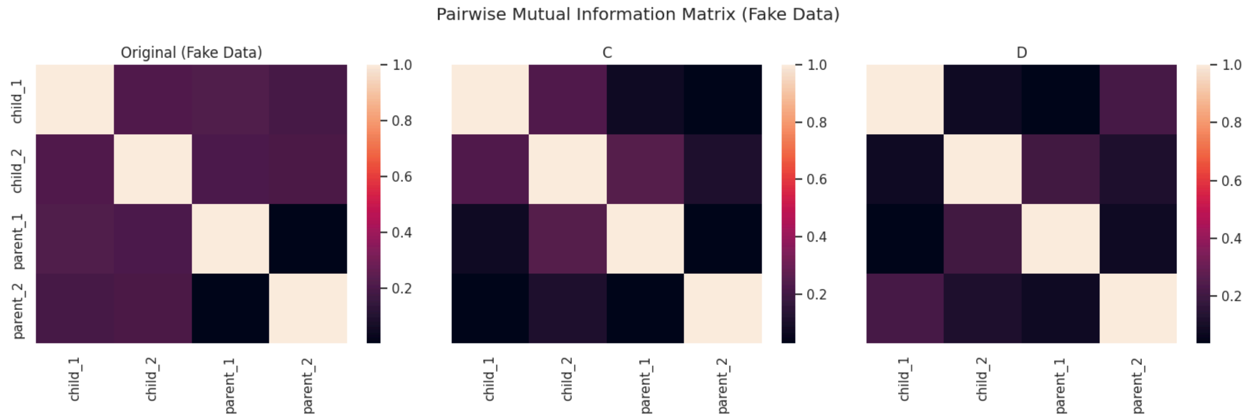


The distribution of age for the real data is skewed right. The distribution of age for the synthetic data generated in random mode (mode A) is very different from the distribution for the real data. The distribution of age, however, for the synthetic data generated in independent attribute mode (mode B) is quite similar to the distribution of age of the real data. It is also skewed right, meaning statistically similar, Look statistically similar but the precise number of people in each bin has changed that’s where the noise comes in. We see the same observations for sex as well. The random mode (mode A) distribution for sex does not match that of the real data while the independent attribute mode (mode B) distribution for sex does match that of real data. In the real data and independent attribute mode data we have almost four times the number of males than females, But in the random mode data we have similar amounts (around 5000) of both males and females. Again in the random mode, we see uniformity. The dataset generated under independent attribute mode preserves the distributions of the real data better than the data set generated under random mode. This is because in random mode we replace the features that we want to protect with random values drawn from uniform distribution while in independent attribute mode the distribution of each feature is learned, and values are generated by sampling from that distribution. The distribution of each feature is learned independently of other features. Therefore, the underlying distribution of each feature is preserved in the synthetic data in independent attribute mode. If the underlying distribution of original data is not uniform, then random mode fails to preserve it.

KS test for age for original vs. A: 0.3735091775112699
KS test for age for original vs. B: 0.026252445351705345
KL test for sex for original vs. A: 0.22319792405369002
KL test for sex for original vs. B: 0.0002494300869420041

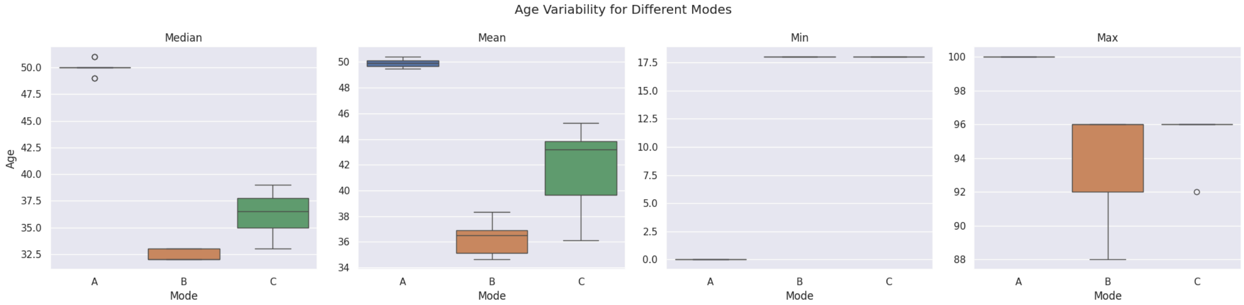
KS statistic for mode A is larger than B, meaning that the the original and mode A are more likely to come from different distributions. KL statistic for mode A is larger than B, meaning that the original and mode A probability distributions are more different from each other.

Q3



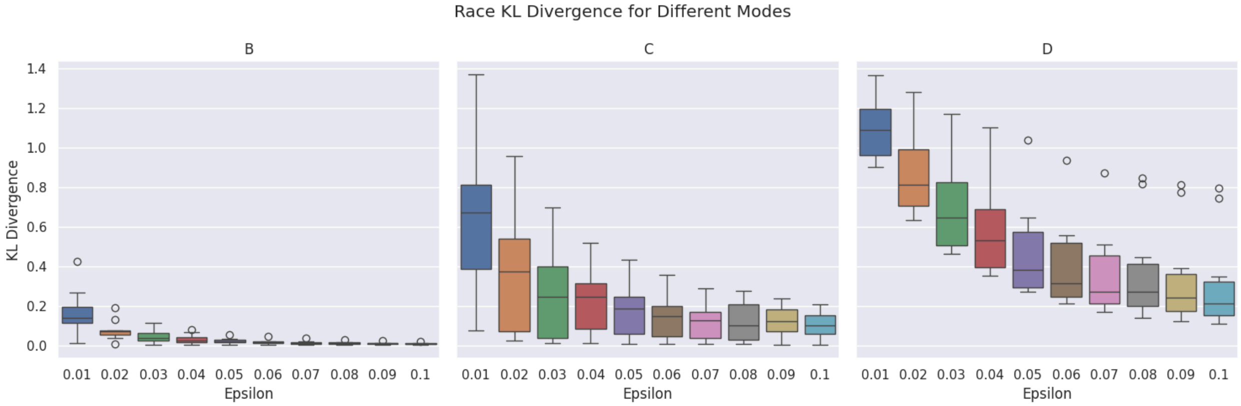
If the normalized mutual information is equal to 0 then two attributes are independent and if the normalized mutual information is equal to 1 then the two attributes are perfectly correlated. Using this definition, from the heatmaps, beige means two attributes perfectly correlated, purple means that two attributes have a medium correlation and black means that two attributes have no correlation. We can see that in both synthetic datasets generated under C and D, about 10/16 (around 62.5%) of the blocks match for both. Since a correlation matrix's diagonals have to be perfectly correlated (beige), to see how well mutual information was preserved we can just focus on the 12 (16-4 diagonals) other blocks. Only 6/12 = 50% of the non-perfectly correlated blocks were preserved for both so they were not able to preserve mutual information that well. More specifically, in the heatmap for hw_fake, the diagonals are perfectly correlated, child_1 has a medium correlation with child_2, each parent has a medium correlation with each child, and parent_1 has no correlation with parent_2. Although both modes C & D preserve about the same amount of information, some of the specific correlations each preserves varies. Both modes preserve the diagonals and there is no correlation between parent_1 & parent_2. Mode C preserves the medium correlation between child_2 & parent_1 and child_1 & child_2. Mode D preserves the medium correlations between child_1 & parent_2 and child_2 & parent_1. Mode C preserves the medium correlations slightly better than Mode D, as its purple blocks are slightly lighter. Overall, Mode C preserves the correlations marginally better, hence may be slightly more accurate, than mode D but both modes do not preserve mutual information that well. The differences in which correlations were preserved may be due to the k hyperparameter. In mode C, we only look at a maximum of one parent, and in Mode D, we look at two parents which may have affected how relationships are learned between attributes.

(b)

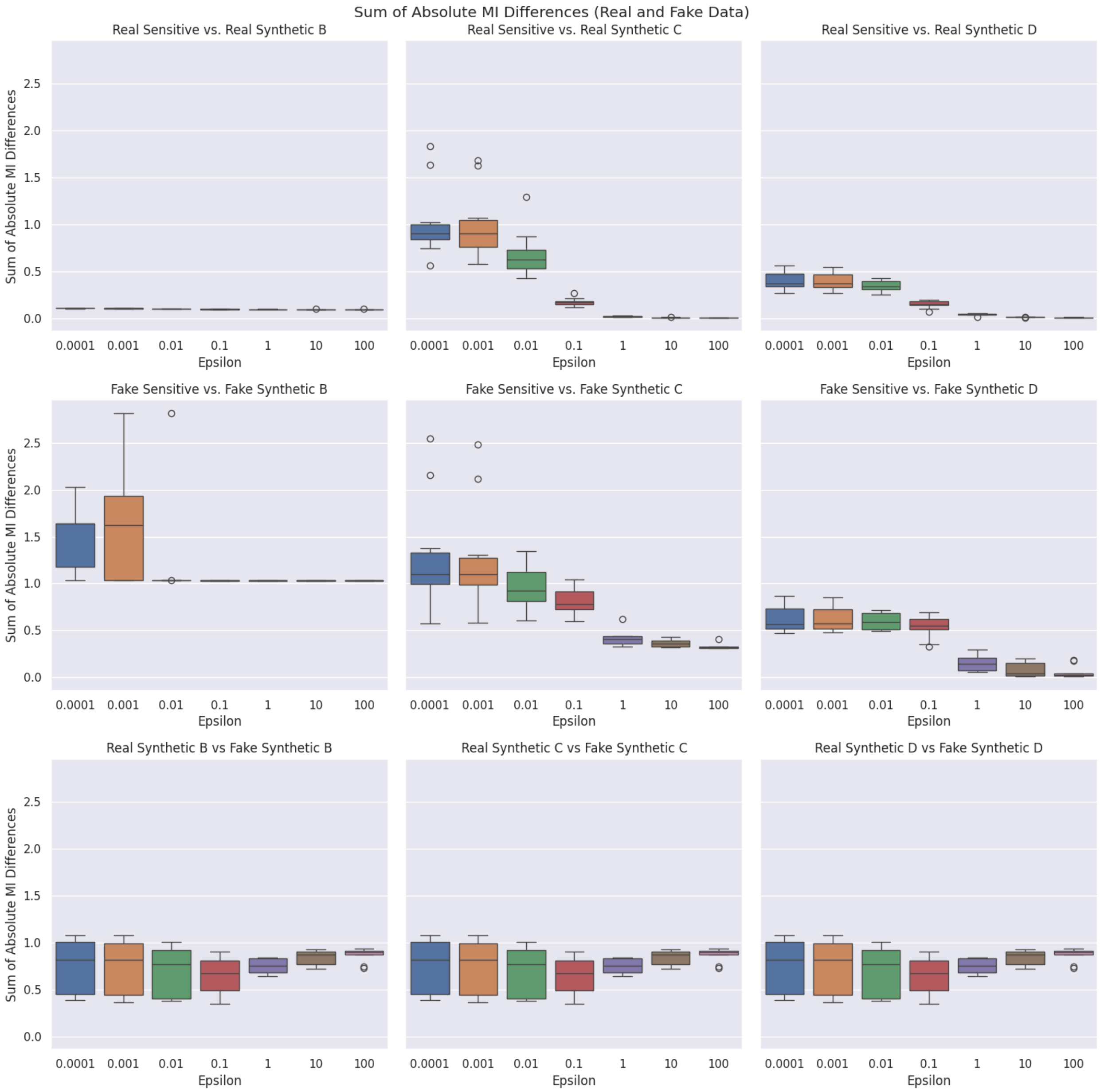


We can see that mode B gives more accurate results as its values for median age, mean age, min age are nearest to the actual values (32, 35, and 18). Mode C is second most accurate and Mode A is the least accurate. Random mode generates random values for a feature from a uniform distribution, which is why it makes sense that it is the least accurate. In independent attribute mode, we learn the distribution of age independently of other features. In correlated attribute mode we learn the distribution of age using correlation with other features (conditional probabilities). Age tends to not be highly correlated with the other features (sex, score, race), so correlated attribute mode perturbs the attributes jointly, it can cause the distribution age to change in a way to diverge from the original distribution. Thus, mode B represents the distribution of age most accurately. In terms of variability, A is still the lowest but C is the highest.

(c)



The general trend for all 3 modes is that as the epsilon value increases the mean KL-divergence decreases. A decrease in the KL-divergence means the real distribution for race and the synthetic data's distribution for race become more similar. The gradually shortening box and whisker plots show the reducing variability as epsilon increases. Therefore, we can learn that a lower epsilon value (privacy budget) means stronger privacy as more noise is added, making the synthetic and real distributions less similar and so, KL divergence in this case bigger.



Similarly, the general trend for all 3 modes is that as the epsilon value increases the sum of absolute mutual information differences decreases. A decrease in the sum of differences means the real dataset and the synthetic dataset become more similar. The gradually shortening box and whisker plots show the reducing variability as epsilon increases. Therefore, we can learn that a lower epsilon value (privacy budget) means stronger privacy as more noise is added, making the synthetic and real datasets less similar and so, sum of differences in this case bigger.