Assignment 0

1 Problem on Probability

1.1 (a) assuming X is a continous random variable with CDF F_x

Let $Y = F_X(X)$

$$F_Y(y) = P(Y \leq y)$$
$$= P(F_X(X) \leq y)$$
$$= P(X \leq F_X^{-1}(y))$$
$$= F_X(F_X^{-1}(y))$$
$$= y$$

since Y is Uniform(0,1)

$$\boxed{\text{hence } X = F_X^{-1}(y)}$$

1.1 (b) X is of exponential distribution

$$F_X(x) = 1 - e^{-\lambda x}$$
$$y = 1 - e^{-\lambda x}$$
$$x = -\frac{1}{\lambda} \ln(1-y)$$

since y is Uniform (0,1) so is $1-y$

$$\boxed{x = -\frac{1}{\lambda} \ln y}$$

2. Vectorization

(a) $$d(x,y) = \sum_{i=1}^{d} (x_i - y_i)^2$$

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{bmatrix}, \quad Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_d \end{bmatrix}$$

$$\cancel{(X-Y)^T (X-Y) = \begin{bmatrix} x_1-y_1 \\ x_2-y_2 \\ \vdots \\ x_d-y_d \end{bmatrix}}$$

$$(X-Y)^T (X-Y) = \begin{bmatrix} (x_1-y_1) & (x_2-y_2) & \cdots & (x_d-y_d) \end{bmatrix} \begin{bmatrix} x_1-y_1 \\ x_2-y_2 \\ \vdots \\ x_d-y_d \end{bmatrix}$$

$$= \sum_{i=1}^{d} (x_i - y_i)^2 = d(x,y)$$

(b)

3. Probability and simulation

(a) Say x are the min flips required

Case 1 $\rightarrow$ Tail appears on first flip – $(x+1)$ flips

Case 2 $\rightarrow$ Head then tail – $(x+2)$ flips

Case 3 $\rightarrow$ Head then head – 2 flips

$$x = \frac{1}{4}(x+1) + \frac{3}{4} \times \frac{1}{4}(x+2) + 2 \times \frac{3}{4} \times \frac{3}{4}$$

$$9x = 28$$

$$\boxed{x = \frac{28}{9} \approx 3.1}$$ minimum 3 flips are required

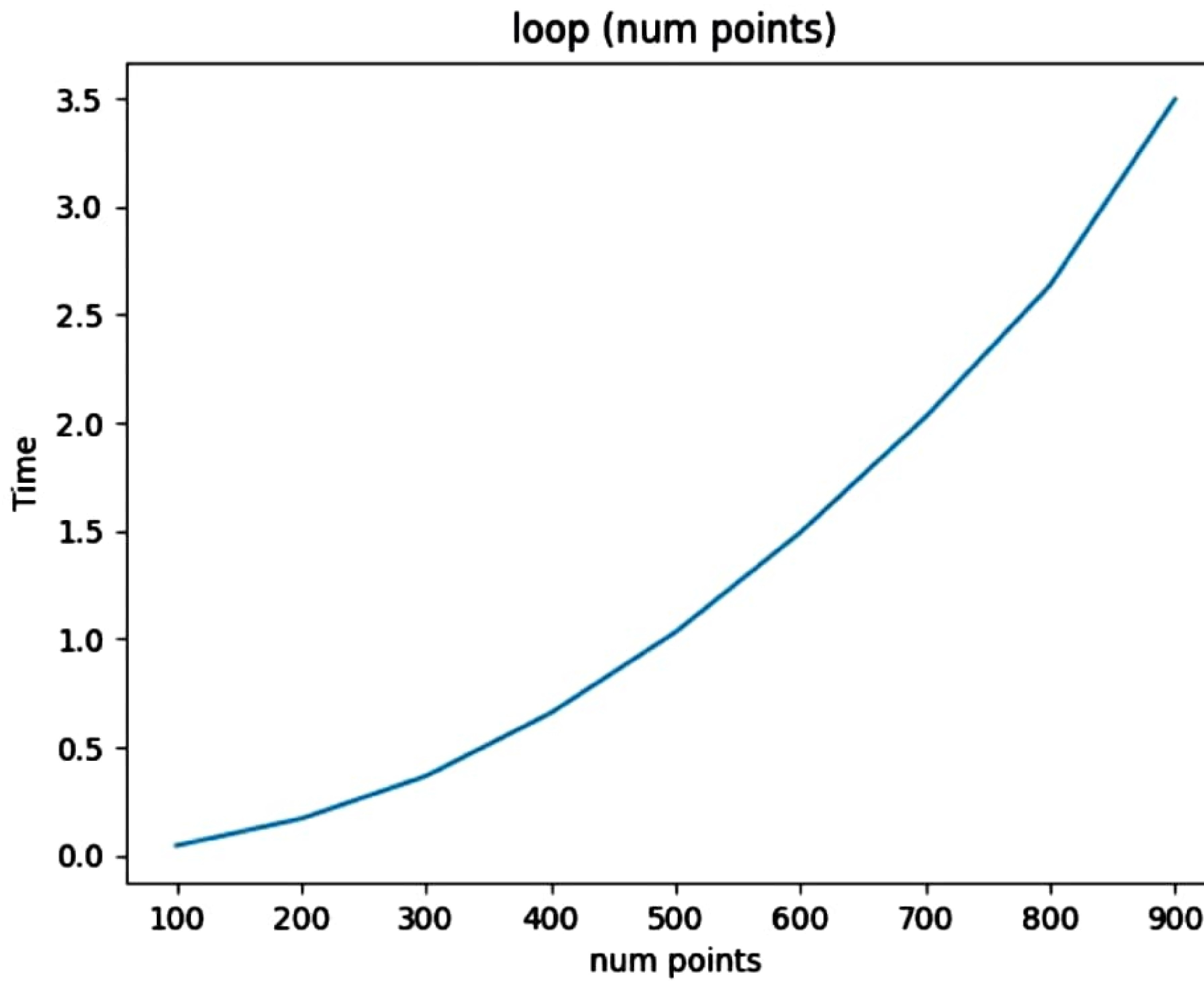
2 (d) after plotting time vs sumpoints and time vs dimension,

One of the evident conclusion is the vectorized functions are highly optimised compared to simple for loops

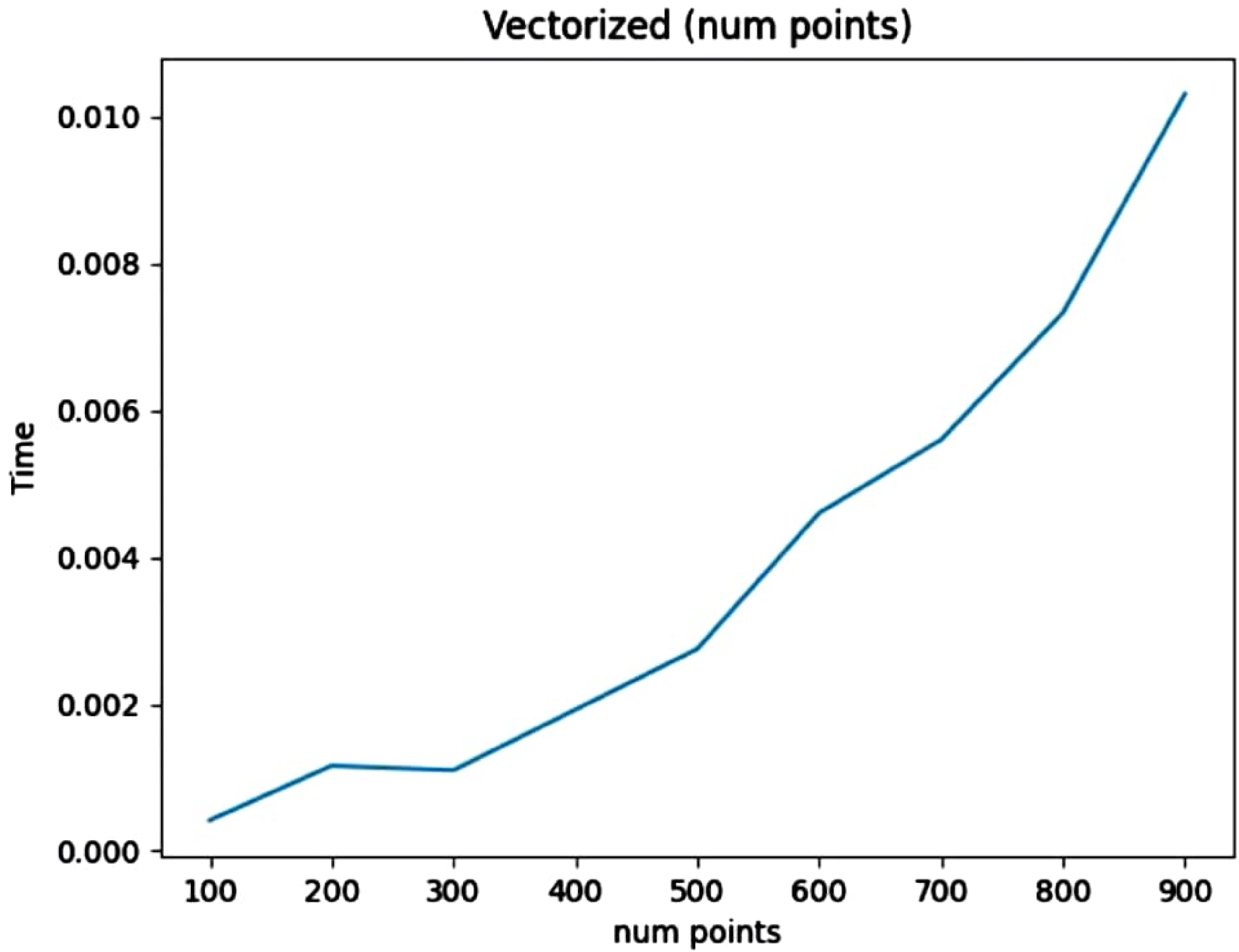
3 (b) for the Simulation, I have designed the program in the following way

for each n ~~where n ∈ {10, 100, 1000}~~ where $n \in \{10, 100, 1000, 10000\}$ expected value is calculated 10 times. Finally mean value for each of these is stored as avg expected value error is determined using avg expected value − Theoretical value

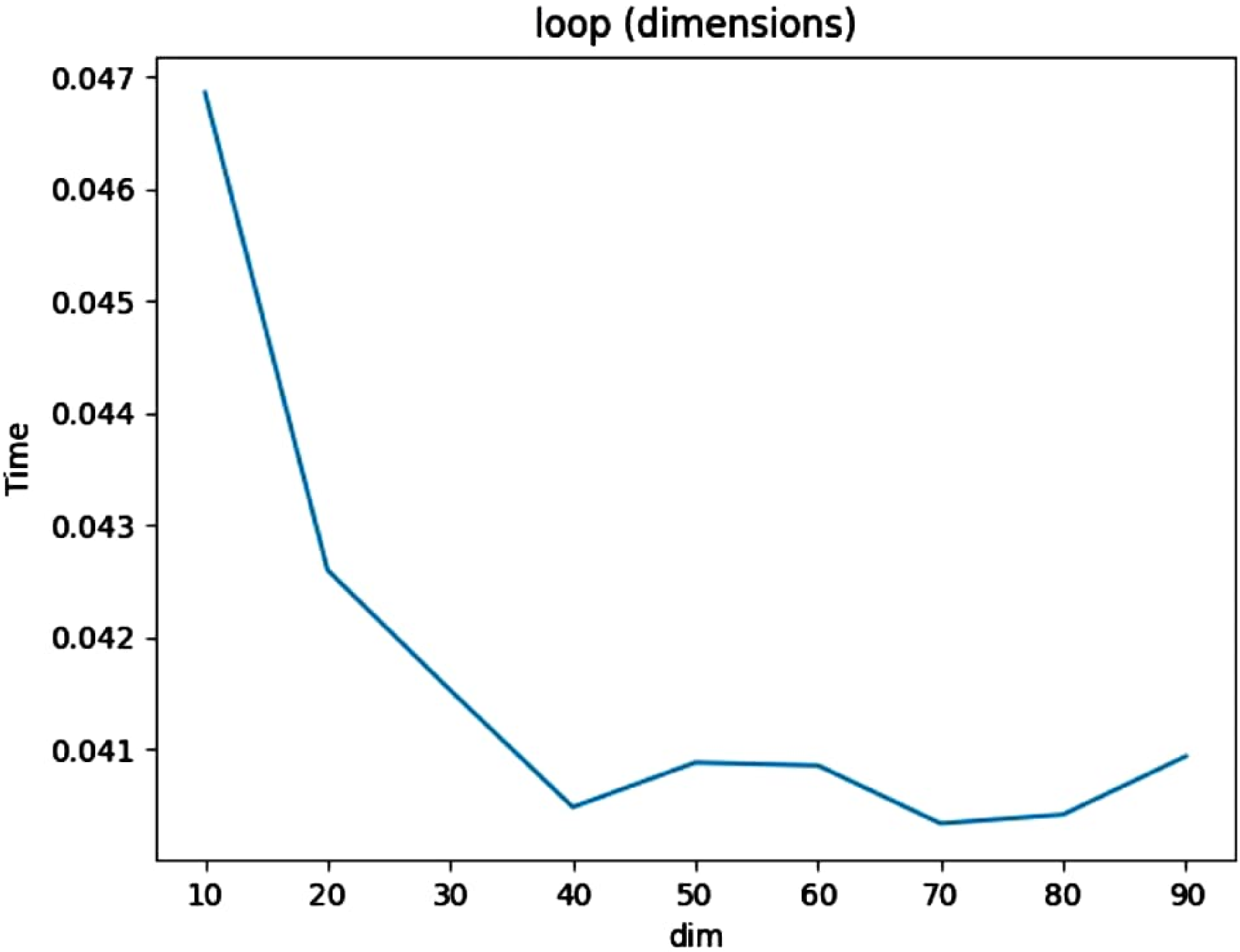
finally the error bars are then plotted

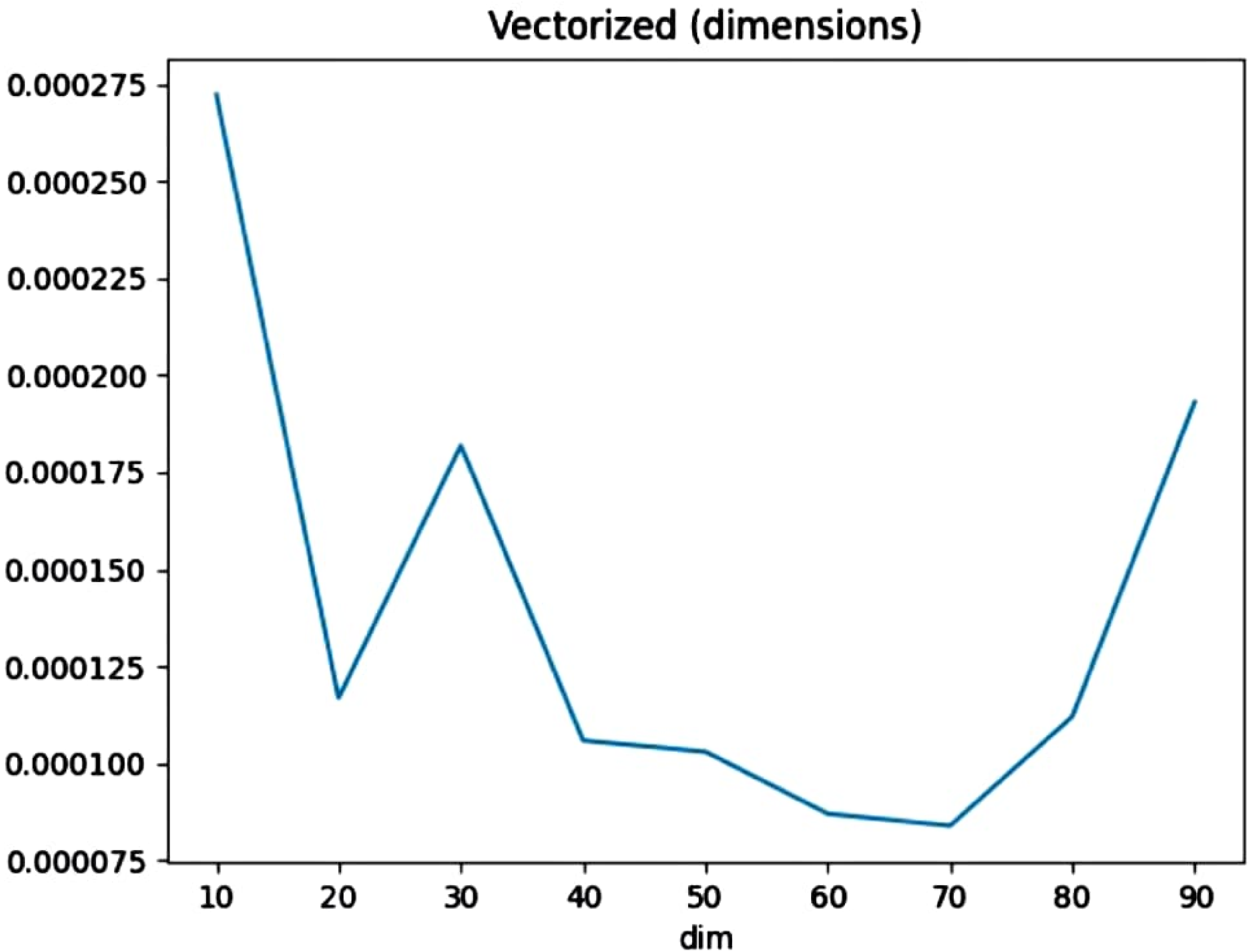


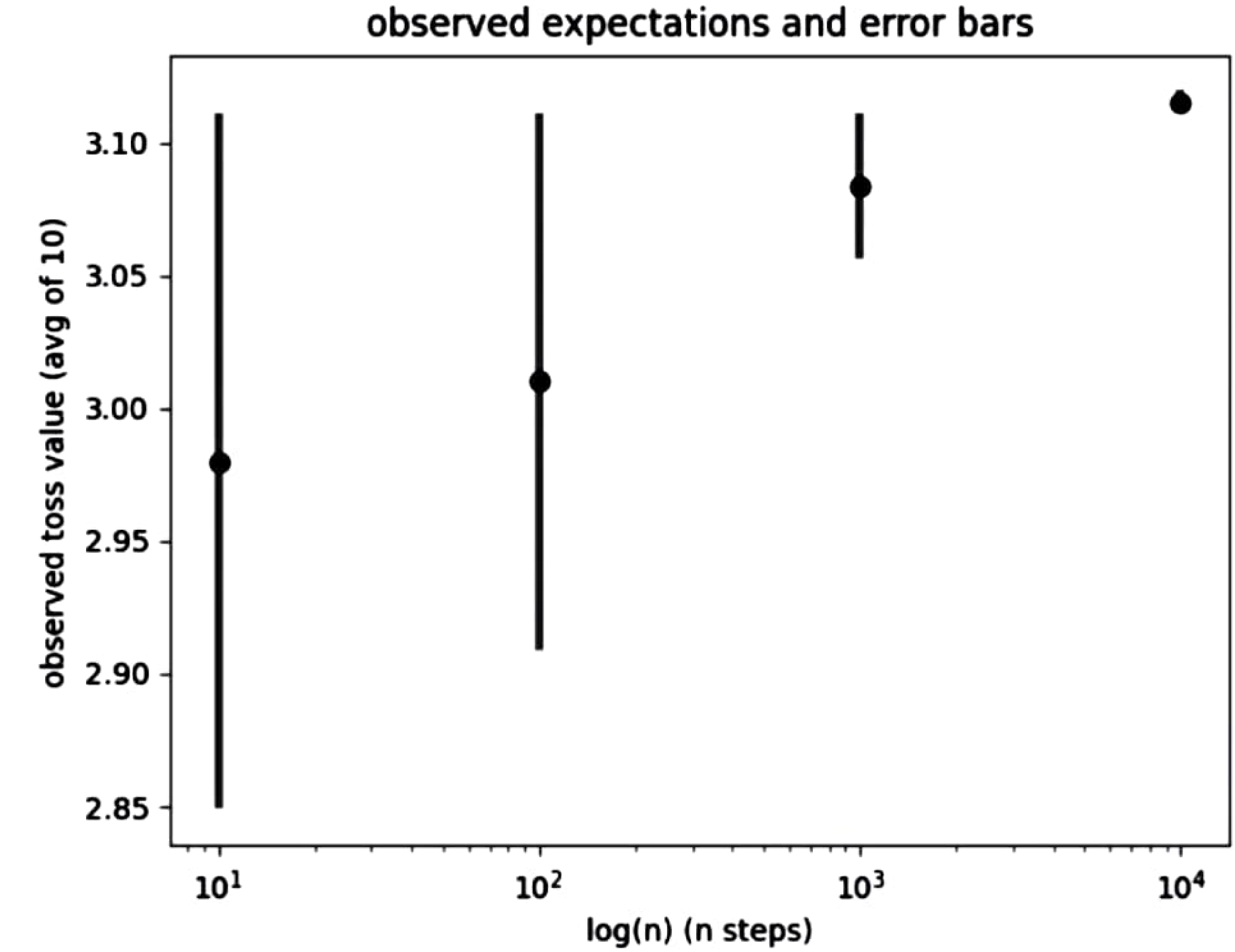
loop (num points)
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
Time
100
200
300
400
500
600
700
800
900
num points



Vectorized (num points)
0.010
0.008
0.006
0.004
0.002
0.000
Time
100
200
300
400
500
600
700
800
900
num points


loop (dimensions)
0.047
0.046
0.045
0.044
0.043
0.042
0.041
Time
10
20
30
40
50
60
70
80
90
dim


Vectorized (dimensions)
0.000275
0.000250
0.000225
0.000200
0.000175
0.000150
0.000125
0.000100
0.000075
10
20
30
40
50
60
70
80
90
dim



observed expectations and error bars
observed toss value (avg of 10)
3.10
3.05
3.00
2.95
2.90
2.85
10^1
10^2
10^3
10^4
log(n) (n steps)