LOG-NORMAL STATE PREPARATION

Jan Tułowiecki
j.tulowiecki@beit.tech

May 24, 2022

1 L2 Error computation

Let's start by computing the *error* for fixed set of values p and x. Let's write $N = 2^n$ and notice that:

$$\begin{aligned}
\mathrm{err}^2 &= \sum_{i=0}^{N-1} \int_{x\in[x_i,x_{i+1}]} \left(\frac{p_i}{x_{i+1}-x_i} - f(x)\right)^2 \mathrm{d}x + \int_{x\in[0,x_0]\cup[x_N,\infty)} f^2(x)\mathrm{d}x \\
&= \sum_{i=0}^{N-1} \int_{x\in[x_i,x_{i+1}]} \left(\left(\frac{p_i}{x_{i+1}-x_i}\right)^2 - \frac{2p_i f(x)}{x_{i+1}-x_i}\right) \mathrm{d}x + \int_{x\in[0,\infty)} f^2(x)\mathrm{d}x \\
&= \sum_{i=0}^{N-1} \frac{p_i^2}{x_{i+1}-x_i} - \sum_{i=0}^{N-1} \frac{2p_i}{x_{i+1}-x_i} \int_{x\in[x_i,x_{i+1}]} f(x)\mathrm{d}x + \int_{x\in[0,\infty)} f^2(x)\mathrm{d}x \qquad (1) \\
&= \sum_{i=0}^{N-1} \frac{p_i^2}{x_{i+1}-x_i} - \sum_{i=0}^{N-1} \frac{2p_i}{x_{i+1}-x_i} (\mathrm{CDF}(x_{i+1}) - \mathrm{CDF}(x_i)) + \frac{5\exp(0.0025)}{\sqrt{\pi}} \\
&= \sum_{i=0}^{N-1} \frac{p_i(p_i - 2\mathrm{CDF}(x_{i+1}) + 2\mathrm{CDF}(x_i))}{x_{i+1}-x_i} + \frac{5\exp(0.0025)}{\sqrt{\pi}}.
\end{aligned}$$

This formula for err^2 allows for computation of the error with very good precision and will be used to evaluate the solution. The CDF function is the cumulative distribution function of the log normal distribution with $\mu = 0, \sigma = 0.1$. Python code may look like this:

```
def l2_error(pmf, x_grid):
    pmf = np.array(pmf)
    x_grid = np.array(x_grid)
    assert all(pmf >= 0)
    assert np.isclose(sum(pmf), 1)
    assert all(x_grid >= 0)
    assert all(np.diff(x_grid) > 0)
    assert len(pmf) + 1 == len(x_grid)

    ans = 5 * (math.pi ** -0.5) * math.exp(0.0025)
    for i in range(1, len(x_grid)):
        d = (x_grid[i] - x_grid[i - 1])
        ans += pmf[i - 1] ** 2 / d
        ans -= 2 * (pmf[i - 1] / d) * (lognorm.cdf(x_grid[i], 0.1)
                                       - lognorm.cdf(x_grid[i - 1], 0.1))
    return ans ** 0.5
```

2 Discretization selection

Let's assume that we are given probabilities array p and we fixed $x_0 = L, x_N = R$ for some preselected constants L and R, $0 < L < R$. We may think about this as searching for the best fit of assumed distribution p into interval $[L, R]$.

Note, that the distributions will be close to each other, if their cumulative distribution functions will be close. On the other hand, we have that the total area covered by the log normal distribution over the interval $[L, R]$ is $\mathrm{CDF}(R) - \mathrm{CDF}(L)$ which is always less than one. This in particular means that fitting distribution p precisely is impossible.

The best fit I could think of is the linear fit. We map the interval $[0, 1]$ into $[\mathrm{CDF}(L), \mathrm{CDF}(R)]$ via linear mapping. Note, that the corresponding function in *scipy* is named *ppf* (Percent Point Function). Thus we obtain the following fit of the corresponding CDFs:

$$x_{k+1} = \mathrm{ICDF}(\mathrm{CDF}(L) + (\mathrm{CDF}(R) - \mathrm{CDF}(L)) \times \sum_{i=0}^{k} p_i), \tag{2}$$

where ICDF is the Inverse Cumulative Distribution Function of the log normal distribution with $\mu = 0, \sigma = 0.1$ and the empty sum for $k = -1$ is considered to be equal to 0.

This discretization selection is close to the optimal and later we will see that it's sufficent for solving the problem. In code, this may be implemented as follows:

```
def ikses_from_pmf(pmf, L, R):
    l_val = lognorm.cdf(L, 0.1)
    r_val = lognorm.cdf(R, 0.1)
    s = 0.0
    ikses = []
    for p in pmf:
        ikses.append(lognorm.ppf(l_val + (r_val - l_val) * s, 0.1))
        s += p
    ikses.append(R)
    return ikses
```

3 Probabilities selection

We will first try to solve this problem using only 1 qubit gates. Note, that any single qubit unitary gate may be decomposed via so-called Euler Angles Decomposition in many ways. We are however interested in ZYZ basis.

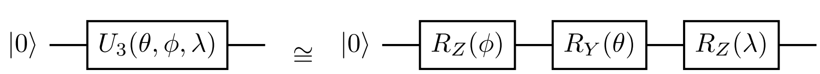

Figure 1: Decomposition of any unitary single qubit gate into Euler Angles in ZYZ base, up to the global phase.

Note, that the first R_Z gate effects only the global phase and thus it won't impact the resulting distribution. Similarly, the last gate R_Z introduces only global and local phase, neither of which affect the outcome of the measurement following the gate. It means that we can only consider circuits consisting of a single R_Y gate applied to each wire (but with perhaps different θs).

Finally, we note that $R_Y(\theta) \cdot |0\rangle = \cos(\theta/2)\,|0\rangle + \sin(\theta/2)\,|1\rangle$, which in turn corresponds to $p = \cos^2(\theta/2)$ and $1 - p = \sin^2(\theta/2)$, which covers the whole interval $[0, 1]$. In other words, given an array P of n numbers, we can compute the distribution p as follows:

$$p_{i=b_0 b_1 \dots b_{n-1}} = \prod_{k=0}^{n-1} \begin{cases} P[k], & \text{if } b_k = 0, \\ 1 - P[k], & \text{otherwise.} \end{cases} \tag{3}$$

where b_i is the binary representation of the index i of our target probability function. In code, obtaining the list of probabilities from the array of probabilities corresponding to measuring zero, may look as follows:

```
def pmf_zeros(zeros):
    pmf = []
    tval = [(t, 1 - t) for t in zeros]
    for t in product([0, 1], repeat=len(zeros)):
        v = 1.0
        for s, x in zip(tval, t):
            v *= s[x]
        pmf.append(v)
    return pmf
```

To compute θs from array *zeros*, we simply use the formula $\theta = 2 \arccos \sqrt{P}$. Remember to apply this formula to the qubits in the correct order (corresponding to bit ordering)!

Now we create an ansatz function that will compute the err value based on the array zeros and values L and R as follows:

```
def ansatz(zeros, L, R):
    pmf = pmf_zeros(zeros)
    ikses = ikses_from_pmf(pmf, L, R)
    return l2_error(pmf, ikses)
```

Now we only need to compute optimal values of the array zeros and variables L and R. To do this, we implement a very simple annealing solution, which runs for about 12 000 iterations per minute. Results presented in this paper were obtained after a minute of computation.

4 The result

Since using all $n = 10$ qubits to span the basis of our target *PMF* will provide the best resolution, we begin with this case. Within a minute, we obtain a solution with the error of `0.003472` which is well within the expected limit. Since reducing the number of qubits should decrease the resolution by a factor of about 2, we expect to be able to succesfully solve the problem using only $n = 9$ qubits and 0 CX gates.

For interested people I present the result for $n = 10$ qubits:

```
zeros = [0.8304947422393207, 0.14476028760855256, 0.2931536038035233,
    0.635408640818282, 0.5323765838116589, 0.5114214251004989, 0.5045610894978408,
    0.5016512863818093, 0.5016000418771398, 0.4996325545654904]
L = 0.6862051153652076
R = 1.444663632383746
```

We ran the solution again for one minute for $n = 9$ and $n = 8$ and obtain errors of `0.00661` and `0.012782` correspondingly. As expected, we are able to solve the problem using only $n = 9$ qubits! The solution for $n = 9$ has parameters as follows:

```
zeros = [0.1424109391649266, 0.8127469076841247, 0.6744165384990002,
    0.37325863512903934, 0.48130747504142957, 0.49208104614493486, 0.4966600222281997,
    0.49683060653262484, 0.49841581401947566]
L = 0.6999362421054021
R = 1.4162329072874003
```

The exact values of the discretization points are described in the appendix B. **All the qubits should be measured for the log-normal distribution!**

Appendix A Ploting the result

Due to the structure of the solution (tensor product of R_Y gates), we observe a fractal-like pattern generating the values of our probability mass function. The result is presented in Fig. 2. Note that the most probability is accumulated in the centre of the distribution, to better approximate the peak of the log-normal distribution. This way we can approximate the parts with greater slope with higher resolution and in result, reduce final L2 error.

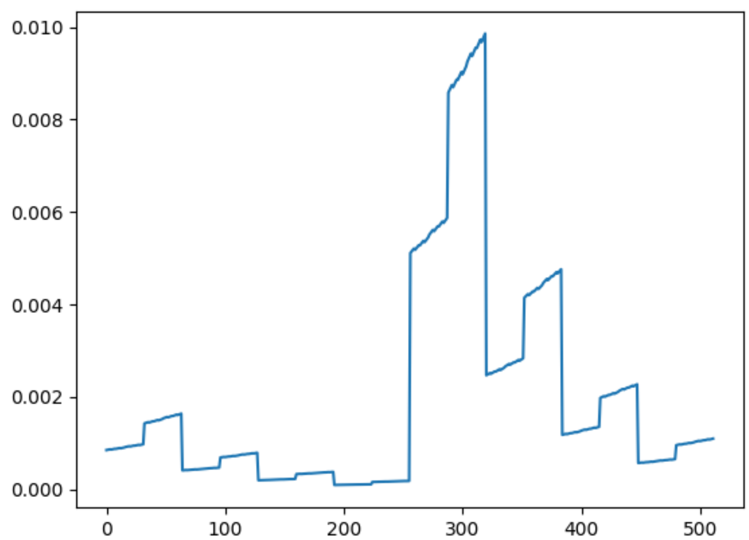

Figure 2: Values of probability mass function in the solution.

The discretization domain is then computed and presented in Fig. 3. We see that slopes around 0 and around 512 are much greater than around the middle of the plot. This again allows for a greater resolution for the crucial part of the distribution.

Finally, we combine both PMF and the discretization domain to see how beautifully they combine together. The red curve in the Fig. 4 is the probability mass function of the actual log-normal distribution with $\mu = 0$ and $\sigma = 0.1$. The blue curve is our result.

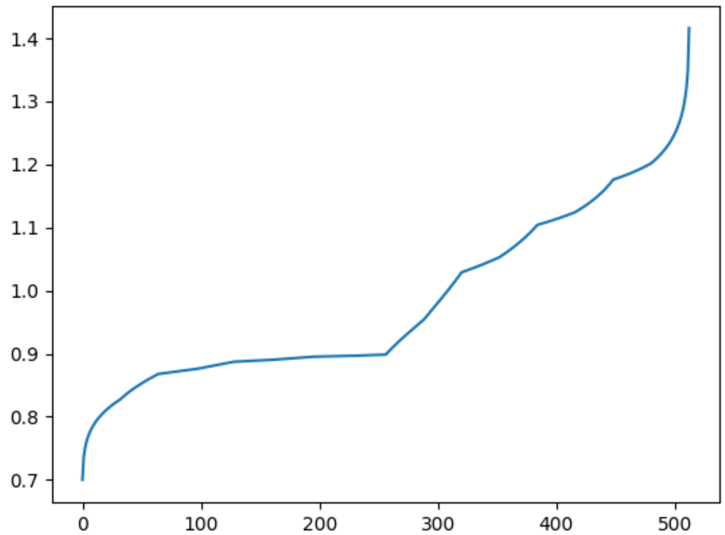

Figure 3: Discretization domain as a sequence of values.

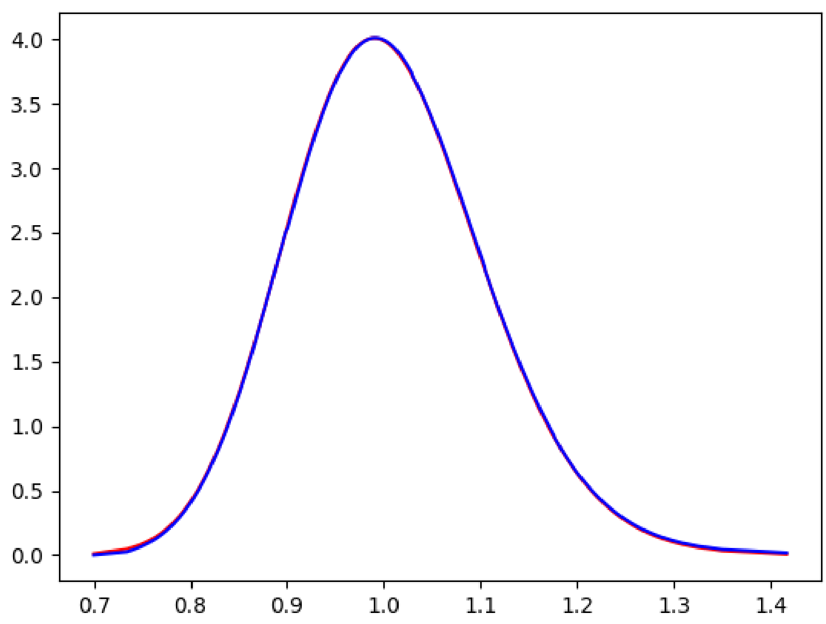

Figure 4: Combining PMF and discretization reveals an almost perfect fit between the solution and the actual distribution.

Appendix B Values of x_i

[0.6999362421054021, 0.7347758788787174, 0.7484682378283596, 0.7575075129872215, 0.7643802640307514, 0.7699212871713811, 0.7746381592894078, 0.7787649875349292, 0.7824472922227133, 0.7857597335065737, 0.7887961442435373, 0.7916051439400213, 0.7942233585507168, 0.7966508003360417, 0.7989414169924636, 0.8011126144472922, 0.803178622181215, 0.8051597581702232, 0.8070566687630698, 0.8088778656802353, 0.8106306086472722, 0.8123014155311941, 0.8139168690450274, 0.8154815571041671, 0.8169995067320102, 0.8184641281627592, 0.8198893583387552, 0.8212779819950513, 0.8226324894698338, 0.8239395950030071, 0.8252176507565692, 0.8264684437347414, 0.8276935936702073, 0.8294276507643693, 0.8311102510887345, 0.8327454305131016, 0.8343367583902959, 0.8358692266361433, 0.8373648647076943, 0.8388261556759504, 0.8402553328582516, 0.841644754012302, 0.8430063840951779, 0.8443418804380576, 0.8456527507705511, 0.846925216881213, 0.848176325995374, 0.8494072206066344, 0.8506189488662251, 0.8518177141551116, 0.8529990186771165, 0.8541637270275019, 0.8553126364278042, 0.8564331094548963, 0.8575396492107358, 0.8586328927573863, 0.859713429942223, 0.8607743973030798, 0.8618239498924829, 0.8628625782159427, 0.8638907374902515, 0.8648968224546278, 0.8658935830753902, 0.8668813989031496, 0.8678606225686871, 0.8681033946297048, 0.8683468123344923, 0.868590878211027, 0.8688355927462018, 0.8690780381428876, 0.8693211381225792, 0.8695648952401833, 0.8698093100122757, 0.8700526695581119, 0.8702966914552219, 0.8705413783014155, 0.8707867306486762, 0.8710298228887512, 0.871273586339295, 0.8715180236280287, 0.8717631353391169, 0.8720100087257786, 0.8722575588939337, 0.8725057885507261, 0.8727546983285877, 0.8730013201833597, 0.8732486279775222, 0.8734966244500558, 0.8737453102673839, 0.8739929425456744, 0.8742412689483585, 0.8744902922627333, 0.8747400131952825, 0.8749874538289582, 0.8752355977007581, 0.8754844476328985, 0.875734004368452, 0.8760971484604365, 0.8764607522596214, 0.8768248225667933, 0.8771893631273564, 0.8775500391839396, 0.8779112117957273, 0.8782728875513742, 0.8786350700084196, 0.8789952286076635, 0.879355912815632, 0.8797171291422226, 0.8800788810730688, 0.8804368655662379, 0.8807954097177628, 0.8811545198775949, 0.881514199392506, 0.8818760434397804, 0.8822384616639493, 0.882601460596897, 0.8829650437359945, 0.8833248854405015, 0.8836853344519411, 0.8840463971698241, 0.8844080769785361, 0.8847678455123764, 0.8851282481809181, 0.8854892913653525, 0.8858509784343498, 0.886209005088161, 0.8865676969363884, 0.8869270602678878, 0.8872870983757606, 0.8873759160782013, 0.8874652002314349, 0.8875549532465056, 0.887645176785717, 0.8877347924311906, 0.8878248782381498, 0.887915436633981, 0.8880064692905915, 0.8880973373044211, 0.8881886810657409, 0.8882805030242873, 0.8883728048637163, 0.8884644833344597, 0.8885566413167435, 0.8886492812769768, 0.8887424049086475, 0.888836428273152, 0.8889309411725232, 0.8890259461094047, 0.8891214447936312, 0.8892162957822656, 0.8893116401378771, 0.8894074803802069, 0.8895038182292069, 0.889599977543302, 0.8896966359990816, 0.8897937961394348, 0.8898914596963969, 0.8899884593623145, 0.8900859620592904, 0.8901839703477082, 0.8902824859700139, 0.8904262501606929, 0.8905706801490245, 0.8907157788997204, 0.8908615481633574, 0.8910062462739234, 0.8911516145907207, 0.8912976561034678, 0.8914443725799883, 0.8915907343229835, 0.8917377728320922, 0.8918854911266768, 0.8920338909902965, 0.8921811993464254, 0.8923291889872998, 0.8924778629584644, 0.8926272230618041, 0.892777935096813, 0.8929293400270139, 0.8930814409389903, 0.8932342396471176, 0.8933859105669515, 0.8935382790415117, 0.8936913481852807, 0.8938451198325893, 0.8939985148047385, 0.8941526142028108, 0.8943074211743008, 0.8944629375721754, 0.8946173053623719, 0.894772382364766, 0.8949281717561093, 0.8950846754105778, 0.8951237951631786, 0.8951631462063968, 0.895202729979525, 0.8952425475952551, 0.8952821230333501, 0.895321932285775, 0.8953619768058311, 0.8954022577163859, 0.8954424924138785, 0.8954829645559166, 0.8955236756157254, 0.895564626730547, 0.8956053283299911, 0.8956462699517337, 0.8956874530832382, 0.8957288788720568, 0.8957707329848797, 0.8958128334119594, 0.8958551816747334, 0.895897778945019, 0.8959401159740107, 0.8959827019708548, 0.8960255384715379, 0.8960686266583611, 0.8961116642259741, 0.8961549545859376, 0.896198499294937, 0.8962422995500732, 0.8962858317219443, 0.896329619395343, 0.8963736641417355, 0.8964179671688405, 0.896482673129448, 0.8965477436438636, 0.8966131808147657, 0.8966789862019998, 0.8967443731494359, 0.8968101281444893, 0.896876253307193, 0.8969427502089992, 0.8970091522318876, 0.897075927426763, 0.897143077938251, 0.8972106053538186, 0.8972777024355454, 0.8973451762445251, 0.8974130289430026, 0.8974812621301996, 0.8975501811979378, 0.8976194860126763, 0.8976891787782452, 0.897759261120033, 0.8978288952643266, 0.8978989187950626, 0.8979693339340334, 0.8980401423185573, 0.8981108470676542, 0.8981819465523427, 0.8982534430197983, 0.8983253381236742, 0.898396772424733, 0.898468605164324, 0.8985408386078769, 0.8986134744211457, 0.9006292233075226, 0.9026155636045202, 0.9045745491456245, 0.9065080566814904, 0.9083952692521855, 0.910261005947858, 0.9121067201307989, 0.9139337529061475, 0.9157307816520964, 0.9175118714090666, 0.9192781177527756, 0.9210305393080285, 0.9227495077967959, 0.9244569397988665, 0.9261536818554857, 0.9278405262456872, 0.9295256170521026, 0.9312021855687287, 0.9328709485756019, 0.9345325806312219, 0.936168106199432, 0.9377980425528971, 0.9394229849910507, 0.9410434969899005, 0.9426488504985108, 0.9442509801131872, 0.9458504067349585, 0.9474476262650771, 0.9490241810053414, 0.9505996461833082, 0.9521744863504075, 0.9537491464251721, 0.9560331677599572, 0.9583126074435195, 0.9605888499863271, 0.9628632294396526, 0.9651100438950434, 0.9673577483842054, 0.9696075912910848, 0.9718607938215622, 0.9741027909717814, 0.9763506131842181, 0.9786054786638392, 0.9808685970159241, 0.9831141378498673, 0.9853703203302533, 0.9876383879882557, 0.9899195935924001, 0.9922253922051241, 0.9945470386175359, 0.996885937953471, 0.9992435262013664, 1.0015929280125417, 1.0039637343354764, 1.0063575389407404, 1.008775992463596, 1.0112036651704592, 1.0136592775360225, 1.0161447496891378, 1.0186620962983435, 1.0211829459720247, 1.023739427342661, 1.026333925729215, 1.0289689756443015, 1.0296323853452378, 1.0303017083169783, 1.0309770429587912, 1.0316584845757706, 1.032337926243753, 1.0330235806064476, 1.0337155543076937, 1.034413951063436, 1.0351139327343108, 1.03582049774178, 1.0365337632507188, 1.0372538437228929, 1.0379721801644417, 1.0386974632396384, 1.0394298207887758, 1.040169378212398, 1.0409195786545045, 1.0416772930190934, 1.0424426664133775, 1.0432158419405342, 1.0439876415838893, 1.0447674153696516, 1.0455553230429724, 1.0463515227860405, 1.0471505300494732, 1.0479580827732957, 1.0487743595500134, 1.0495995380143084, 1.0504238382168063, 1.0512572622334426, 1.0521000085307544, 1.0529522753215343, 1.054205497372647, 1.05547620152012, 1.056764949793978, 1.0580723216289591, 1.059383053309352, 1.0607132937180548, 1.0620637097123202, 1.0634349925152973, 1.0648180695071419, 1.0662232700973933, 1.0676514061502163, 1.0691033243104693, 1.0705622335056917, 1.0720462803627888, 1.0735564563109194, 1.0750938016342129, 1.0766663996778945, 1.0782686872224292, 1.079901951291824, 1.0815675531704172, 1.0832465597856844, 1.0849602135145624, 1.0867101537292283, 1.0884981277409183, 1.0903131120675915, 1.0921695432094183, 1.0940695925518311, 1.096015595869194, 1.0979861031898162, 1.1000067871735175, 1.102080579149873, 1.1042106653926467, 1.104744559321046, 1.1052847177219434, 1.105831262396471, 1.1063843141301095, 1.1069373098769482, 1.1074969509859576, 1.1080633709582346, 1.1086367025995174, 1.1092130058066396, 1.109796428557662, 1.1103871193939805, 1.1109852266070985, 1.1115836638984353, 1.1121896951789405, 1.1128034846948407, 1.113425196954146, 1.1140578008252062, 1.1146987347470656, 1.1153481881673444, 1.1160063516774295, 1.1166654459685703, 1.117333490577952, 1.1180106974404758, 1.1186972805101347, 1.1193885670735426, 1.1200895805770188, 1.1208005622817507, 1.121521756694925, 1.122244664587771, 1.122978108276396, 1.1237223611265308, 1.124477701205671, 1.1255933244253273, 1.126730548909234, 1.127890192728711, 1.1290731130287148, 1.130265735474331, 1.1314830071453594, 1.132725935173748, 1.1339955811820621, 1.1352839185018604, 1.136600970923147, 1.1379480187328885, 1.139326421131835, 1.1407206745377307, 1.1421485956377793, 1.143611828909836, 1.1451121337356382, 1.1466582950358037, 1.1482458240580449, 1.149876991769758, 1.1515542533837704, 1.1532594857037317, 1.1550153049043115, 1.1568248270237296, 1.1586914585047632, 1.1606052738959023, 1.1625832699828191, 1.164629975924427, 1.1667504145415315, 1.168923579972351, 1.171180481819151, 1.1735280032201345, 1.1759739165972747, 1.1765998821755863, 1.1772356849222503, 1.1778815923470778, 1.1785378775848507, 1.1791968394071581, 1.179866554347696, 1.1805473290702926, 1.1812394778479434, 1.181938373171508, 1.182649181240198, 1.183372260873708, 1.1841079813445419, 1.1848477574798095, 1.1856007073652568, 1.186367249915481, 1.1871478180378234, 1.1879464091783614, 1.1887600764361395, 1.1895893350179978, 1.1904347202731398, 1.1912864561764893, 1.1921551356186488, 1.1930413755665115, 1.1939458202333932, 1.1948626369846362, 1.1957988325353308, 1.1967551659262683, 1.1977324342186018, 1.1987194857985395, 1.1997287576810378, 1.2007611878598887, 1.2018177671728452, 1.2033950102663873, 1.2050240845826243, 1.2067084140033373, 1.208451766309591, 1.2102364713968006, 1.2120875214541074, 1.2140099509139, 1.2160093859525818, 1.2180773151064848, 1.2202345584637182, 1.2224891301723069, 1.224850177933448, 1.2272980670465552, 1.2298721909000483, 1.2325863428259702, 1.2354567333400497, 1.2385164968466666, 1.241776718888614, 1.2452659850767784, 1.249019569098311, 1.2530314941774805, 1.2573997058913444, 1.2621956429183792, 1.2675150789597223, 1.2734464053640662, 1.2802128233044812, 1.2880998945648356, 1.2975741888097811, 1.309319782585657, 1.3251623602819815, 1.349949686992977, 1.4162329072874003]