実験環境

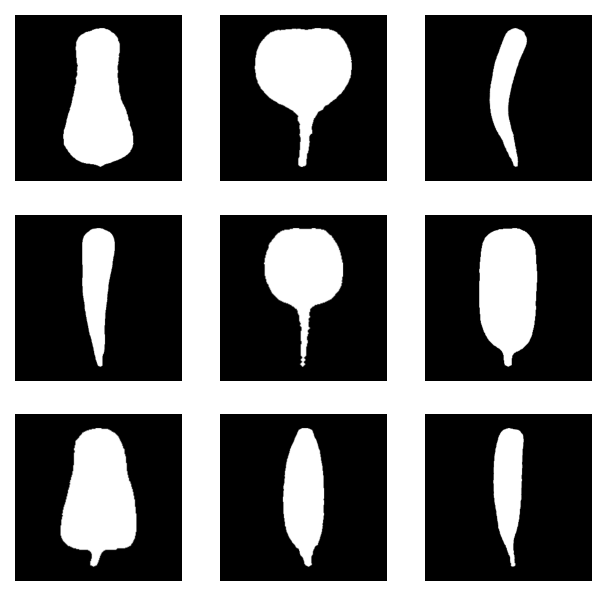
AMD Ryzen 7 PRO 4750GE 3.10 GHz (8core)

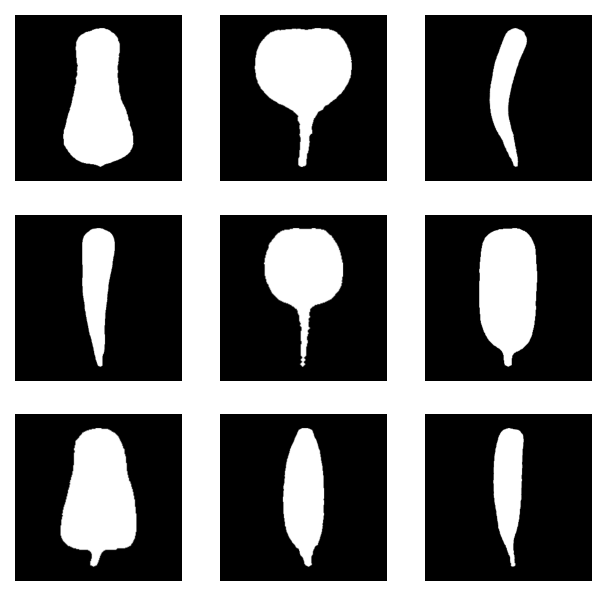
RAM 16GB

Anaconda no GPU

実験対象画像

img 2 lr=0.001 - 0.005





実験１

fit0 err\_th=1.0 で近似するのに必要な次数　（5次から順にテスト）

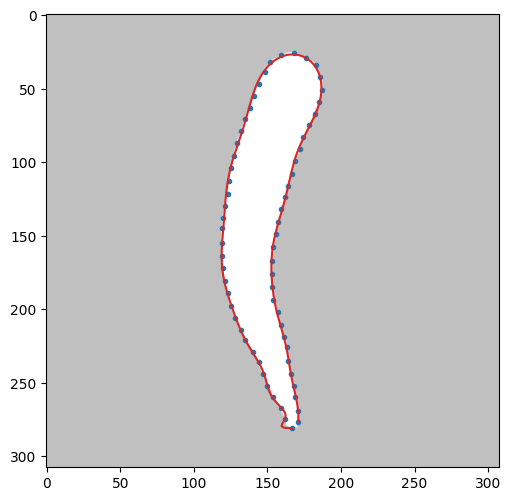
| img | Nsamples=65 | fitting 時間 (s) | 誤差 |
| --- | --- | --- | --- |
| 0 | 17 | 2.0 | 0.7927955 |
| 1 | 18 | 1.5 | 0.9622668 |
| 2 | 20 | 1.7 | 0.9017465 |
| 3 | 16 | 2.2 | 0.9010169 |
| 4 | 20 | 1.4 | 0.9816888 |
| 5 | 15 | 2.0  1.1 | 0.7400273 |
| 6 | 17 | 1.44 | 0.7759406 |
| 7 | 19 | 1.83 | 0.9686808 |
| 8 | 18 | 1.61 | 0.8743404 |

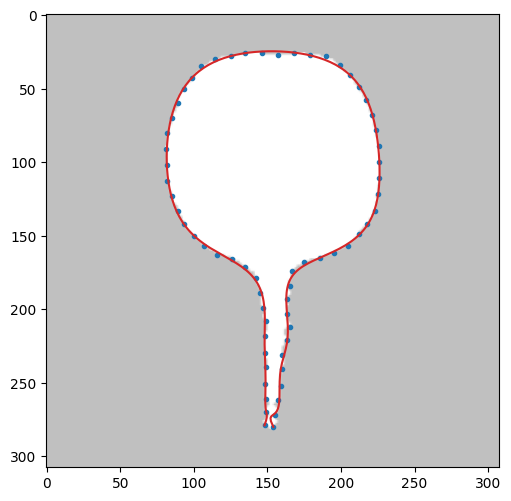
FIT0結果まとめ

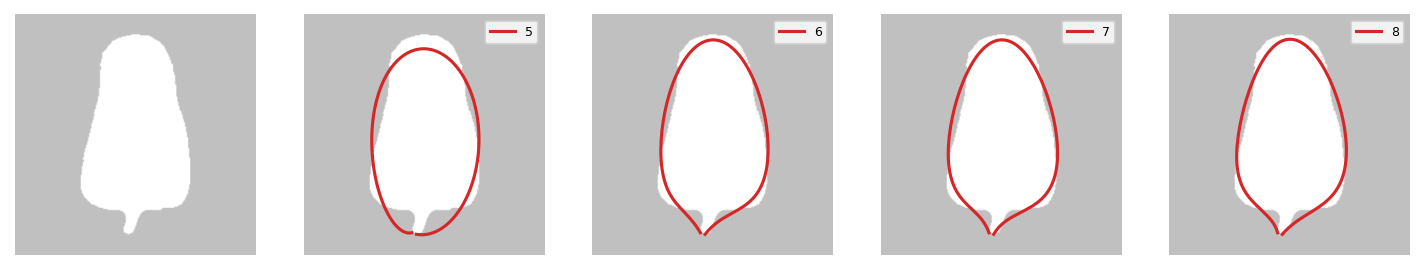
|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

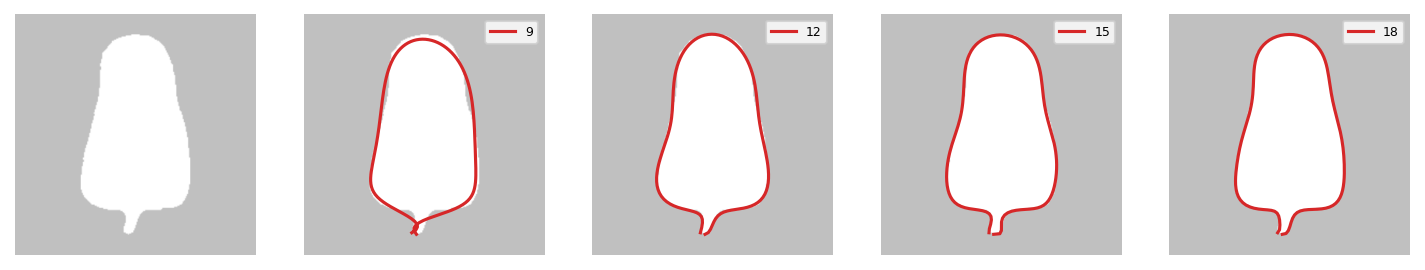
* 次数を高くすることで誤差は小さくできるが、滑らかさが欠けてくる
* 誤差は、表本点と近似曲線上の対応点との間の距離の総和である。対応点が標本点の位置とはずれていても、輪郭線上に位置している場合、見た目では近似曲線は表本点に十分フィットしているように見えてしまう。

次数を大きくしすぎるとオーバフィッティングが発生する。

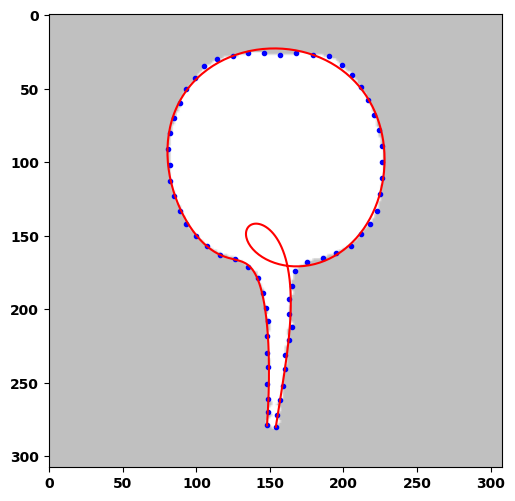
（23次）

2（20次）

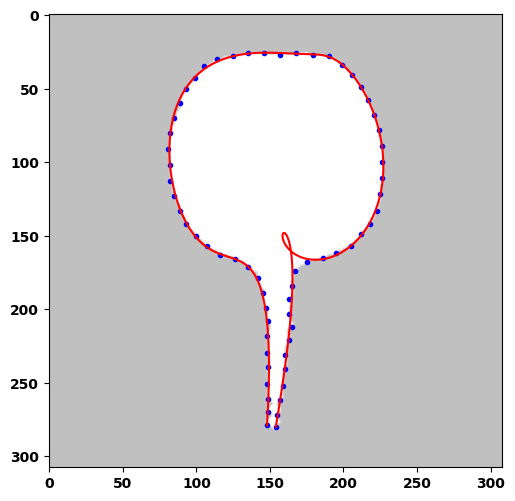
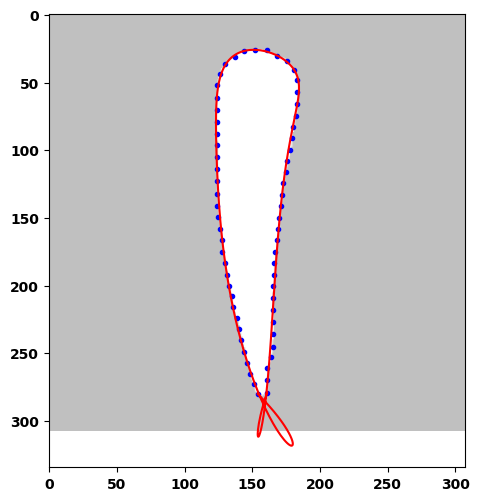


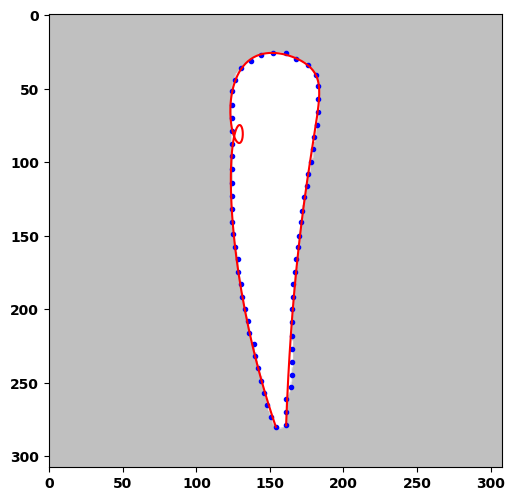


Overfittings



lrP=10000 11次 lrP=5000

fit1T mode 0 img 4 N=11

fit1T mode0 img3 N=8

実験２ fit1

maxTry 1000, err\_th 0.65 時間はLenovo m75q-2

| img | Nsamples=65 | fitting 時間 | 誤差 |
| --- | --- | --- | --- |
| 0 | 7 |  | 0.649787515574525  (収束 385steps,5min 55s)  0.99885782  212.43s |
| 1 | 10 |  | 0.6497152620575508  (14min 13s, 543steps)  https://wandb.ai/aquapathos/checkN%201/runs/31phs3id?workspace=user-aquapathos |
| 2 | 10  11 |  | 0.99868265  156.54s  0.6499734098904164  8min 36s  278steps |
| 3 | 8  10 |  | 0.996736454  738.09s  0.6493236779905556  4min 46s 171steps |
| 4 | 10  11 |  | 0.872408545  4090.92s  0.6495148644593582  (34min 53s  1133stop) |
| 5 | 8  9 |  | 0.90076686  3235.26s  0.6498842589648255 14min 37s 657steps |
| 6 | 11  12 |  | 0.662050  2932.64s  0.6485182889145864  6min 46s 185steps |
| 7 | 6  7  8 |  | 0.7081388434868623  1552.8125s  0.6897  9.578125s  0.649885997075606  3min 25s 184steps |
| 8 | 8  9 |  | 0.6931425  3244.76s 3000steps  0.6499690168879233  24min 48s 1122steps  0.99991572  625.859s |

FIT1 err\_th 0.65 結果まとめ

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

実験２ fit1T mode 0 (パラメータと制御点の同時決定）この実験は別ファイルへ移動しました

lr = 0.001 (0.0001 から0.002 ぐらいの範囲で安定)

lrP = 30000 (10000では小さすぎてOFが発生、30000 - 100000)

| img | Nsamples=65 | fitting 時間 | 誤差 |
| --- | --- | --- | --- |
| 0 | 7 |  | 0.6470270292  1min 16s  479steps |
| 1 | 12 |  | 12: 0.6424112632  788s 3000steps |
| 2 | 9  10  11 |  | 0.6746863144808077, 531.421875s, 3001steps  0.49202134915855045, 381.359375s, 1856steps  0.4949681271  628.3s  2658steps  12次だとOF |
| 3 | 7 |  | 0.4990601529  241,6s  1753steps  8次だとOF |
| 4 | **12** |  | 11次だとOF  0.6401555434, etime: 314.57ss  1153steps |
| 5 | 10  11 |  | 10次  0.4983034288  1497steps  331.7s  根の先部分がフィットせず  0.4845270269  155.6s 614steps |
| 6 | 12 |  | 0.6499550358,  2min 10s  447steps |
| 7 | 8 |  | 0.6367832774,  1min 5s  339steps |
| 8 | 9 |  | 0.6468048464,  3min 40s 1090steps |

FIT1T mode 0 err\_th 0.65 結果まとめ

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

実験２ fit1T mode 1

1. lr=0.001 err\_th=0.65
2. lr=0.0015 err\_th=0.5

(１) は fit1T mode 0 の実験結果との比較のためにlrの値はmode0に合わせて0.001とした。mode 1はもっと高い値でも収束できる。mode 0 ではlr=0.0015とすると、0.65に到達できない個体がいくつかでてくる。mode 1 ではlr=0.001ではerrorを0.5まで下げることができない個体がいくつか生じたが、 lr=0.0015 ならすべて平均誤差0.5未満とできた。

| img | 近似に必要な次数 | 0.5 | 0.65 | 誤差 |
| --- | --- | --- | --- | --- |
| 0 | 0.65 -> 7  7次の限界はminerror: 0.5137262125  0.5 -> 8 |  |  |  |
| 1 | 10 |  |  |  |
| 2 | 11 |  |  |  |
| 3 | 9  9次の限界はminerror: 0.5537344484  10 | 10 |  |  |
| 4 | 11 |  |  |  |
| 5 | 9 |  |  |  |
| 6 | 11 | 12 |  |  |
| 7 | 7 0.51098057 |  |  |  |
| 8 | 9 |  |  |  |

FIT1T mode1 err\_th 0.5 結果まとめ

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

FIT1T mode1 err\_th 0.65 結果まとめ

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

fit1T mode 1

| image | err\_th 5.0 | err\_th 2.0 | err\_th 1.0 | err\_th 0.65 | err\_th 0.5 |
| --- | --- | --- | --- | --- | --- |
| 0 |  |  |  |  |  |
|  | 7次106steps  62.10s | 7次128steps  78.03s | 7次166steps 92.46s | 7次198steps  106.23s | 8 次167steps  122.44s |
| 1  10次 |  |  |  |  |  |
| 備考  err\_th≧１なら9 次でもOKだが時間はかかる | 10 次3steps  14.25s | 10 次67steps  78.73s  (9 次630steps  534.00s) | 10 次152steps  165.61s  (9 次762steps  653.00s) | 10 次215steps  229.81s | 10 次342steps  359.14s |
| 2 |  |  |  |  |  |
|  | 11次1steps  12.11s | 11次5steps  17.90s  (10次12steps  23.25s  ９次13steps  21.75s) | 11次22steps  38.11s  (10次45steps  56.10s  ９次58steps  56.77s) | 11次137steps  180.10s | 11次349steps  440.84s |
| 3  0.65 ９次  0.5 10次 |  |  |  |  |  |
| 8次でオーバーフィッティング発生 | 10次4steps  14.60s | 10次21steps  32.80s  9次40steps  43.66s  8次40steps  36.07s  7次149steps  84.20s | 10次40steps  52.06s  9次91steps  85.09s  8次70steps  56.54s  7次357steps  188.64s | 10次57steps  68.92s  9次173steps  155.85s3 | 10次78steps  90.53s |
| 4  11次 |  |  |  |  |  |
|  | 11次6steps  18.10s | 11次37steps  56.60s  10次80steps  92.15s  9次は無理 | 11次433steps  541.95s  10次600steps  629.45s  9次は無理 | 11次571steps  721.47s | 11次662steps  834.50s |
| 5  9次 |  |  |  |  |  |
|  | 9次0steps  10.35s | 9次18steps  25.72s  8次100steps  74.34s  7次は無理 | 9次101steps  92.87s  8次249steps  173.31s  7次は無理 | 9次175steps  158.89s | 9次252steps  226.84029 |
| 6  0.65 11次  0.5 12次 |  |  |  |  |  |
|  | 12次0steps  11.40s | 12次11steps  27.89s  11次18steps  33.47s  10次108steps  121.80s | 12次37steps  65.25s  11次55steps  77.73s  10次1000steps  s | 12次64steps  104.44s  11次288steps  362.14928 | 12次117steps  184.80s |
| 7 |  |  |  |  |  |
|  | 7次4steps  11,17s | 7次7steps  14.07s  6次9steps  12.87s | 7次15steps  17.29s  6次85steps  42.48s | 7次101steps  59.69s | 7次1430steps  730.20s |
| 8 |  |  |  |  |  |
|  | 9次4steps  13.28s | 9次39steps  43.17s  8次45steps  39.28s | 9次245steps  216.04s  8次は無理 | 9次378steps  331.73s | 9次518steps  445.53s |
|  |  |  |  |  |  |

img 1 N=10 lrP の最適化

%%time

# mode0 lr のチューニング OOO

import tensorflow as tf

def objective(trial):

print(datetime.datetime.now())

obj = 'lrP'

lrP = trial.suggest\_loguniform(obj, 0.1,2500) # loguniform

print('%s: %1.10f' % (obj,lrP))

rd.BezierCurve.debugmode=False

rd.BezierCurve.convg\_coe = 1e-5

rd.BezierCurve.swing\_penalty = 300

rd.BezierCurve.smoothness\_coe = 1e-7

err,etime = check1T(1,mode=0,N=10,pat=200,maxTry=200,err\_th=0.65,lr=0.005,lrP=lrP,test="TEST",wandb=wandb)

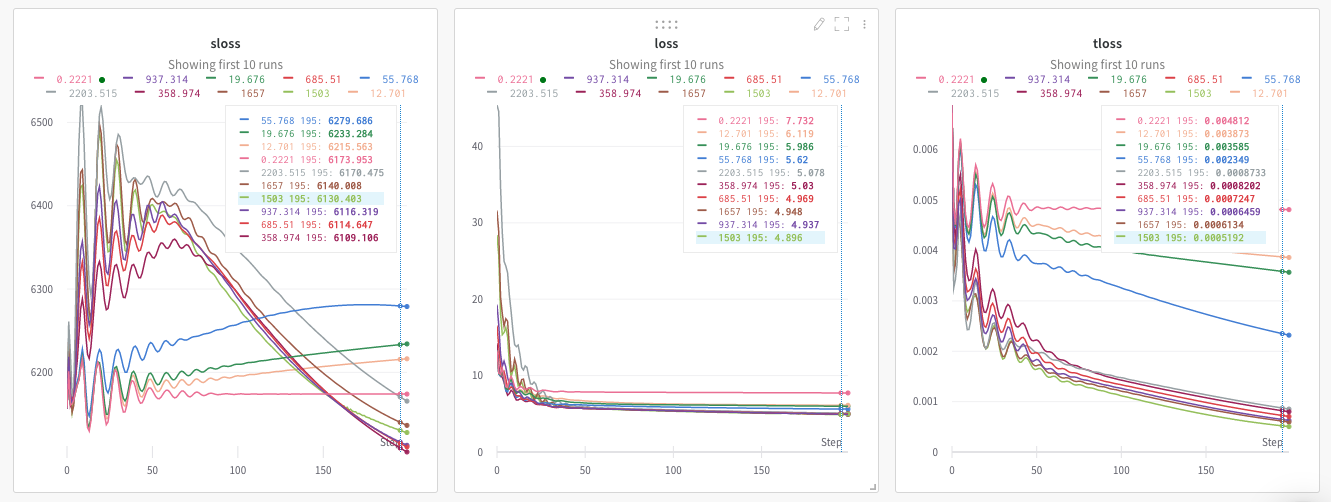
print('%s: %1.10e,minerror: %1.10f, etime: %3.5f' % (obj,lrP, err, etime))

return err

study = optuna.create\_study()

study.optimize(objective, n\_trials=20)

の実験で、次のような結果を得た。



１０００あたりが最適値と思われるので、範囲を 600〜1800に狭めて再実験。maxTry を 300 に増やす。

実験３ fit1T mode 1 でのフィッティング Nsamples=65

Xeon(R) CPU @ 2.30GHz

単独値は err\_th=0.5 pat=100, convg\_coe = 1e-5 での収束値

C は誤差収束による打ち切り、Pはpatientによる打ち切り Eは収束

fit2 はerr\_th = 0.65、fit1Tmode1 は0.5

rd.BezierCurve.mloop\_itt=3

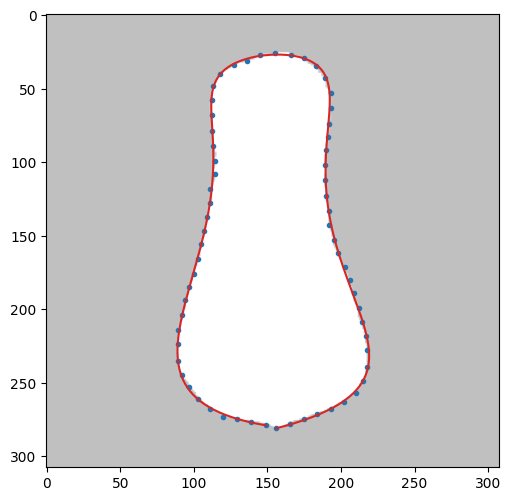
rd.BezierCurve.debugmode=False

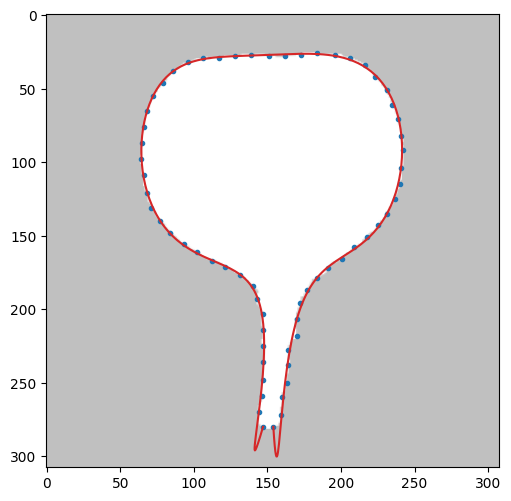
rd.BezierCurve.convg\_coe = 0 # 1e-5

rd.BezierCurve.swing\_penalty = 0 # 300

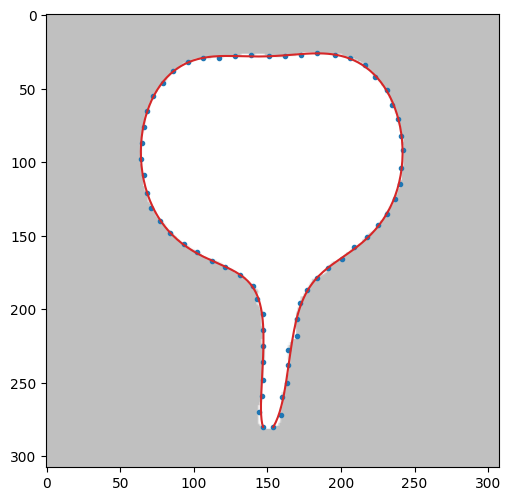
rd.BezierCurve.smoothness\_coe = 0 # 4e-6

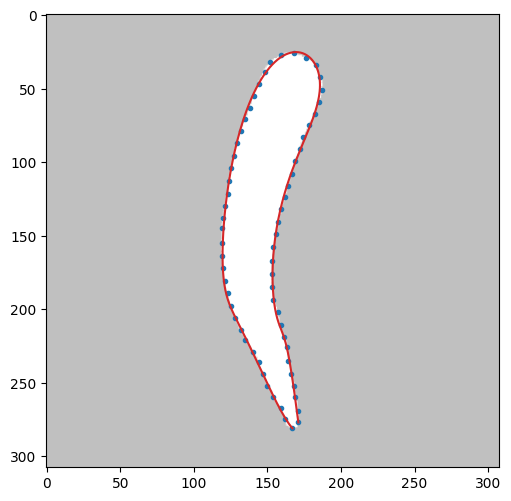
rd.BezierCurve.middle\_coe = 0

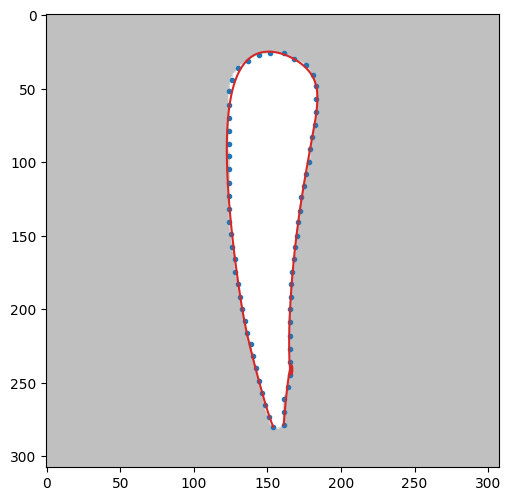
0 0.51371577 7次 fit1T

1 0.6419868 10次 fit2

0.6419868 10次 fit2 terminal\_coe=1.0

1 0.505642533412106 fit1T

2 0.64826476 ９次 fit2

3 0.64529229 9次 fit2 OF

IMG４

%%time Xeon(R) CPU @ 2.30GHz

rd.BezierCurve.convg\_coe = 1e-6

rd.BezierCurve.smoothness\_coe = 4e-6

rd.BezierCurve.swing\_penalty = 380

lrP = 600 #

err,etime = check1T(4,mode=1,N=9,Nsamples=65,pat=200,maxTry=0,err\_th=0.5,lr=0.005,lrP=lrP,test="swing\_penalty",coe=380,wandb=wandb)

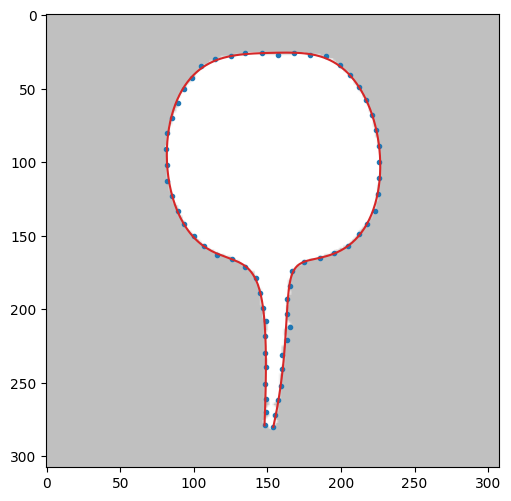
print(err,etime)

Matrix([[154.0\*t\*\*9 + 211.880557994689\*t\*\*8\*(9 - 9\*t) - 2290.30882883669\*t\*\*7\*(1 - t)\*\*2 + 83447.5089539264\*t\*\*6\*(1 - t)\*\*3 - 85503.5701239957\*t\*\*5\*(1 - t)\*\*4 + 99053.3403661315\*t\*\*4\*(1 - t)\*\*5 - 7964.24765352672\*t\*\*3\*(1 - t)\*\*6 - 1823.71636083714\*t\*\*2\*(1 - t)\*\*7 + 1507.34403949259\*t\*(1 - t)\*\*8 + 148.0\*(1 - t)\*\*9], [280.0\*t\*\*9 + 17.0822284711994\*t\*\*8\*(9 - 9\*t) + 5329.97738455751\*t\*\*7\*(1 - t)\*\*2 + 83020.3884966954\*t\*\*6\*(1 - t)\*\*3 - 283809.553377998\*t\*\*5\*(1 - t)\*\*4 + 361997.136949141\*t\*\*4\*(1 - t)\*\*5 - 198951.216079603\*t\*\*3\*(1 - t)\*\*6 + 46930.8475254392\*t\*\*2\*(1 - t)\*\*7 - 1967.12230918608\*t\*(1 - t)\*\*8 + 279.0\*(1 - t)\*\*9]])

0.4948921943040516 905.8992967605591

CPU times: user 14min 55s, sys: 3.96 s, total: 14min 59s

Wall time: 15min 5s



Swing Penalty の効果　N=9 　maxTry 50

penalty: 2.3319728474e+00,minerror: 1.0267258736, etime: 93.80983 OF

penalty: 5.9707960711e+00,minerror: 1.0266114852, etime: 94.33606 OF

penalty: 7.2582358436e+00,minerror: 1.0264897998, etime: 95.88270 OF

penalty: 18.582073103e+00,minerror: 1.0235998813, etime: 94.77232 OF

penalty: 43.448083686e+00,minerror: 1.0166043182, etime: 95.21533 OF

penalty: 83.447298562e+00,minerror: 1.0095841404, etime: 94.93519 OF

penalty: 147.90043777e+00,minerror: 1.0209061013, etime: 94.99020 OF

penalty: 290.91592780e+00,minerror: 1.0294351698, etime: 96.16653 G

penalty: 469.13457083e+00,minerror: 1.0347875175, etime: 96.21131 G

penalty: 747.07901757e+00,minerror: 1.0711307510, etime: 94.21739 G

148未満OF、290以上Good

9次

penalty: 808.69814759e+00,minerror: 1.0804473759, etime: 98.26347

penalty: 1044.2973867e+00,minerror: 1.0836229707, etime: 94.89916

penalty: 1071.5746435e+00,minerror: 1.0824296405, etime: 95.10583

penalty: 1121.3697051e+00,minerror: 1.0801618776, etime: 95.50775

penalty: 1200.9022245e+00,minerror: 1.0845351016, etime: 95.08204

penalty: 1424.7552962e+00,minerror: 1.0862373784, etime: 96.21804

penalty: 1639.0874722e+00,minerror: 1.0675533271, etime: 95.15624

penalty: 2062.9011729e+00,minerror: 1.0553135457, etime: 95.21030

penalty: 4053.4048621e+00,minerror: 1.0718626113, etime: 95.75021

penalty: 5134.0530031e+00,minerror: 1.1517349973, etime: 96.11475

8次

penalty: 302.81031131e+00,minerror: 0.8863438872, etime: 79.17238 OFX

penalty: 502.47864324e+00,minerror: 0.8435173678, etime: 79.86497 OF

penalty: 823.96652814e+00,minerror: 0.8831313287, etime: 76.17550 OF

penalty: 2003.6923418e+00,minerror: 0.9750471748, etime: 79.40048 OF

penalty: 2100.8508367e+00,minerror: 0.9807756079, etime: 77.18000 OF

penalty: 2991.8551364e+00,minerror: 1.3048168518, etime: 77.60901 OFX

penalty: 4925.8407363e+00,minerror: 1.2463756114, etime: 77.20009

penalty: 8418.8139305e+00,minerror: 1.6966924944, etime: 79.03341 OFX

penalty: 17860.166720e+00,minerror: 1.6301024994, etime: 78.45477

penalty: 21437.192969e+00,minerror: 1.8124958493, etime: 76.98046 OF

IMAGE8 N=9 SWING PENALTY の実験

%%time

# swing\_penalty の効果

import tensorflow as tf

def objective(trial):

print(datetime.datetime.now())

obj = 'swing\_penalty'

coe = trial.suggest\_loguniform(obj, 100,1500) # loguniform

print('%s: %1.10f' % (obj,coe))

rd.BezierCurve.debugmode=False

rd.BezierCurve.convg\_coe = 1e-5

rd.BezierCurve.swing\_penalty = coe

rd.BezierCurve.smoothness\_coe = 7e-04

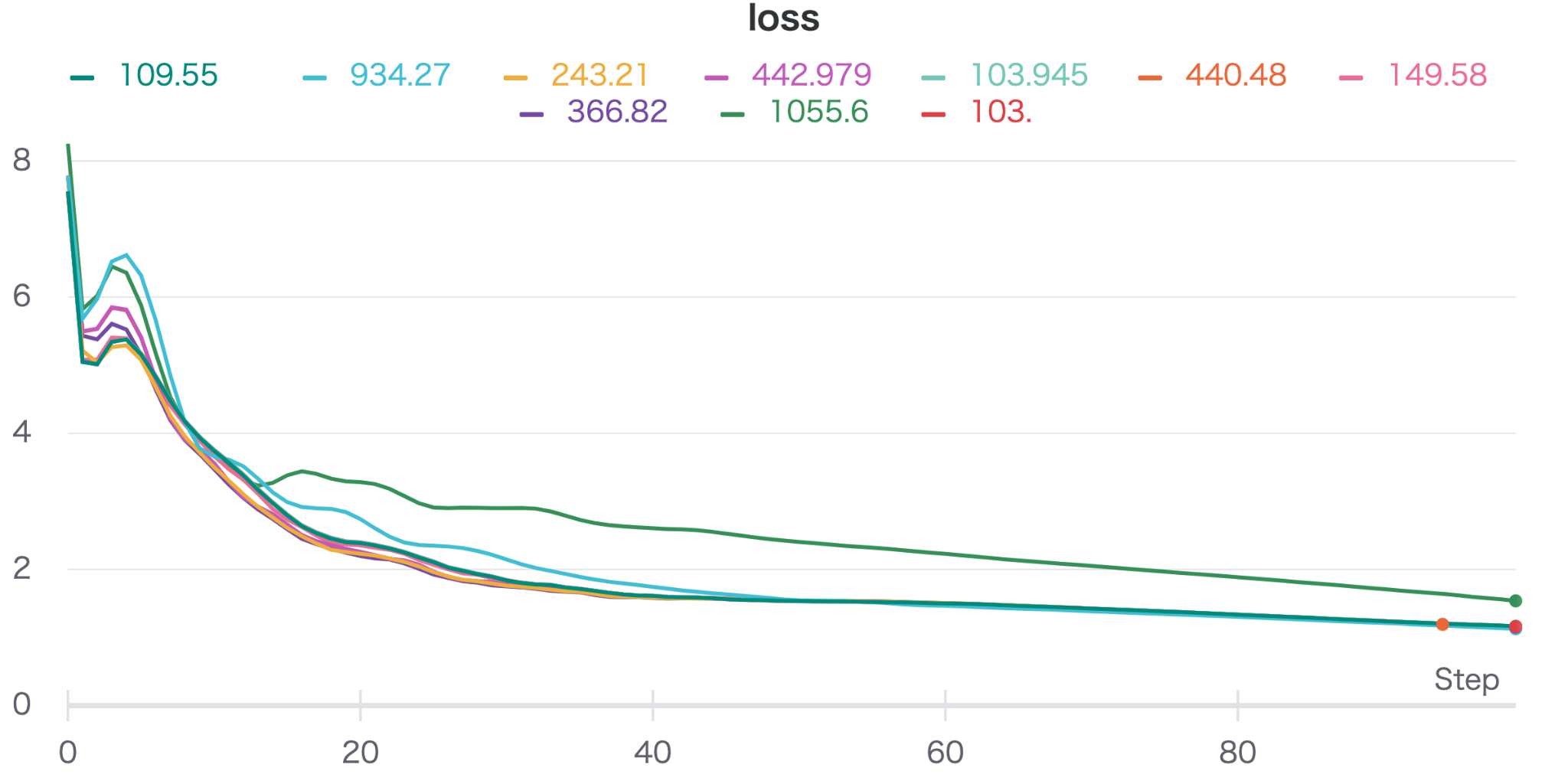
err,etime = check1T(8,N=9,pat=100,maxTry=100,err\_th=0.65,lr=0.005,test=obj,wandb=wandb)

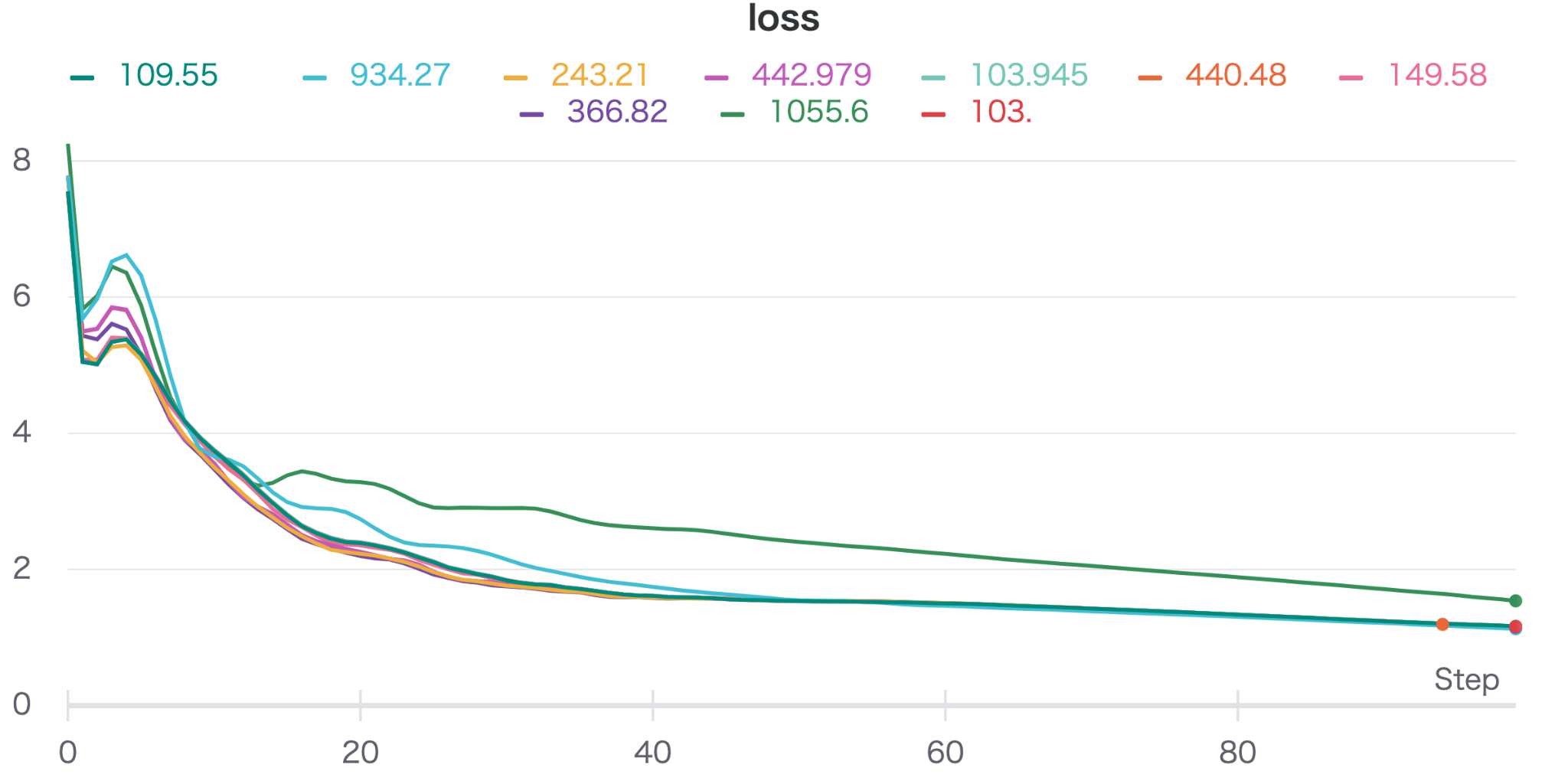
print('%s: %1.10e,minerror: %1.10f, etime: %3.5f' % (obj,coe, err, etime))

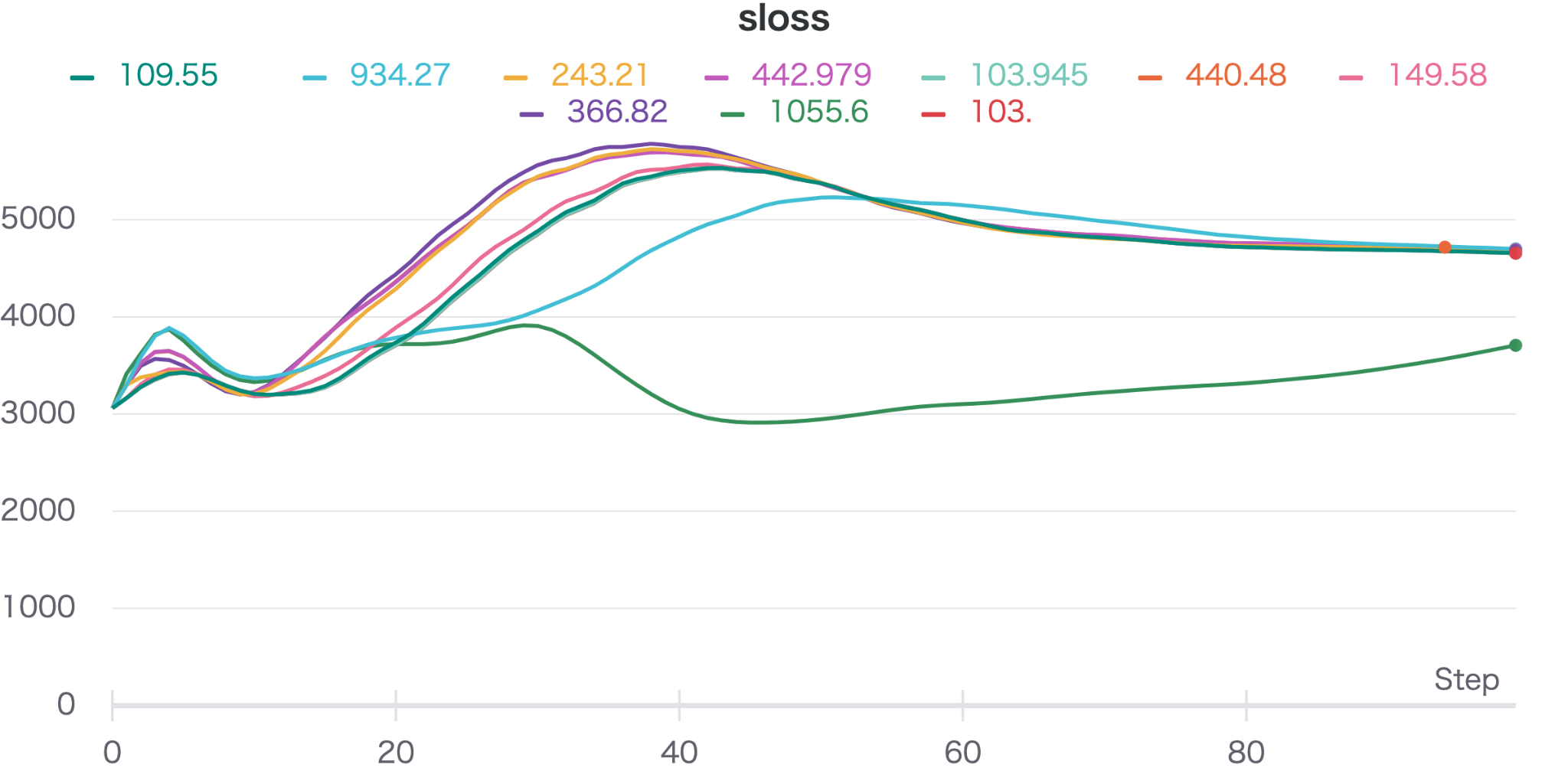
return etime

study = optuna.create\_study()

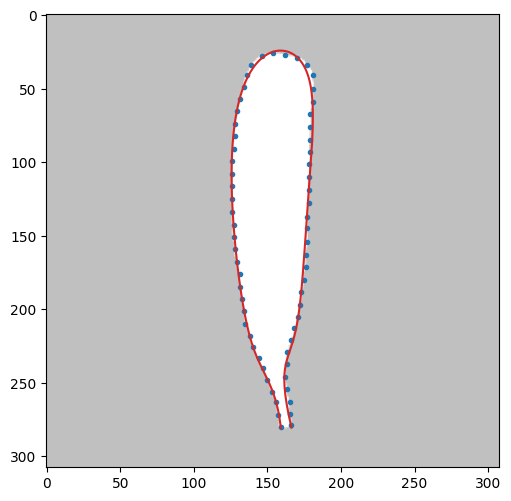
study.optimize(objective, n\_trials=10)



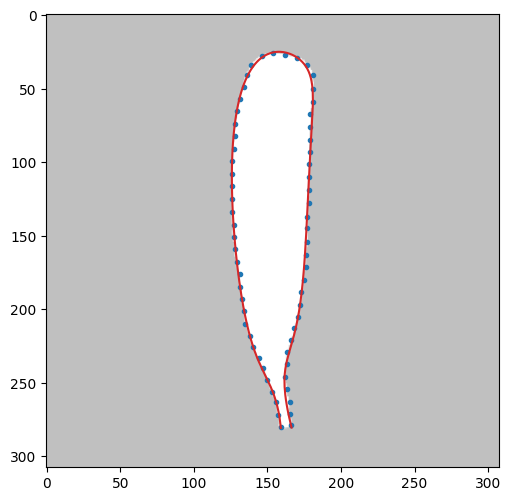




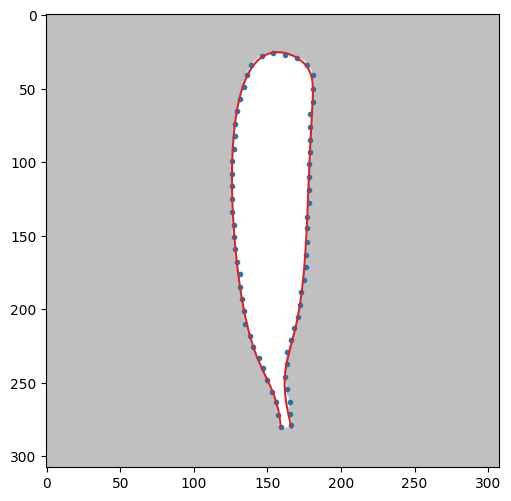
100〜1000の範囲では、目視ではほとんど結果に違いがない



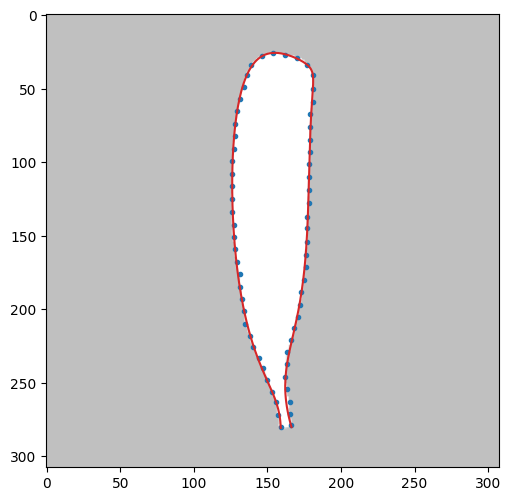
swing\_penalty: 1.0319579693e+02,minerror: 1.1615936158, etime: 184.96781



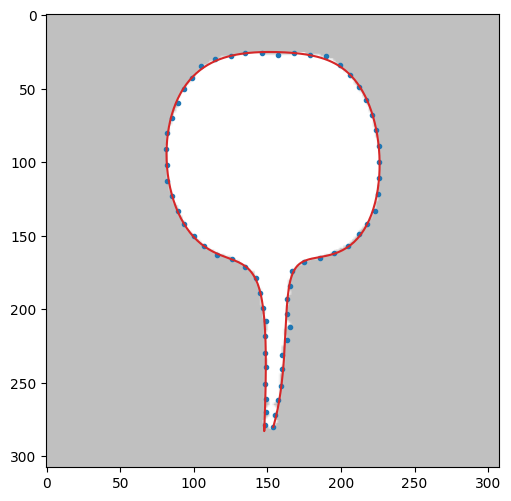
4min 31s 0.65

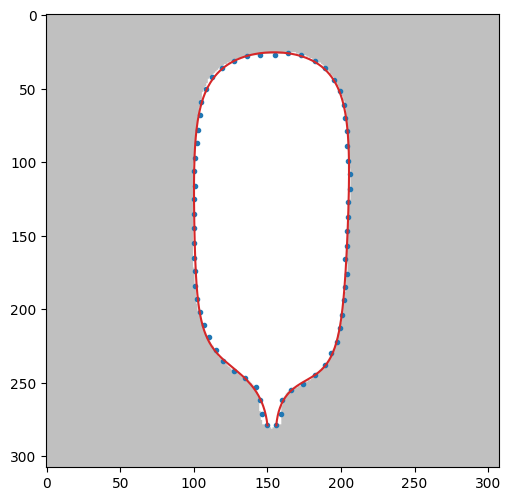


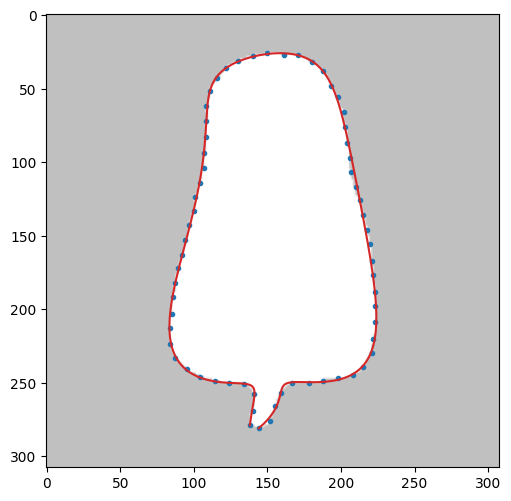
0.5 5min 27s

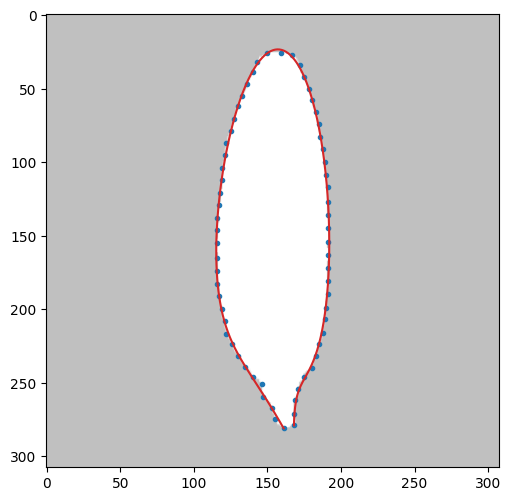


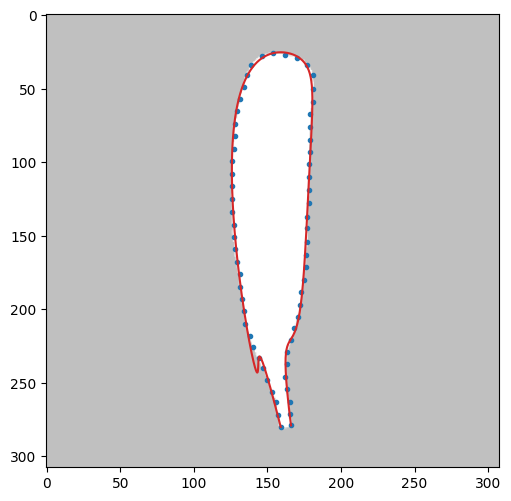
28min 33s error 0.37085849119554914

0.64743554 10次 fit2

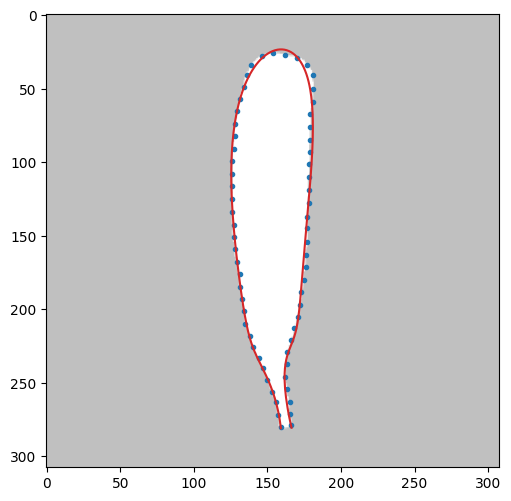
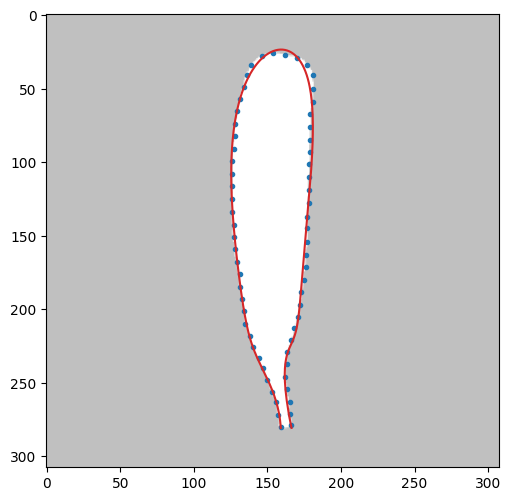
0.6477804 9次 fit2

0.64990233 11次 fit2

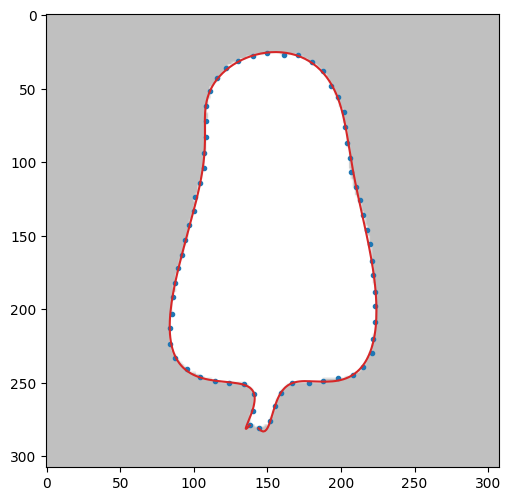
0.64854244 ７次 fit2

0.641830632 ８次 fit2

---



１１次



12次だとオーバフィティングが生じる

img[4] lr に関する実験

%%time

import tensorflow as tf

def objective(trial):

with tf.device("GPU:0"):

lr = trial.suggest\_loguniform('terminal\_coe', 1e-5, 1e-2) # 0.005

print('lr: %1.10f' % lr)

rd.BezierCurve.debugmode=False

rd.BezierCurve.convg\_coe = 1e-5

rd.BezierCurve.terminal\_coe = 0

#rd.BezierCurve.smoothness\_coe = 0

rd.BezierCurve.smoothness\_coe = 0 # 1e-20 # 1e-8 #2e-8

err = check1T(6,N=11,pat=100,maxTry=100,err\_th=0.3,lr=lr,test='lr',wandb=wandb)

print('lr: %1.10f,minerror: %1.10f' % (lr, err))

return err

study = optuna.create\_study()

study.optimize(objective, n\_trials=10)

maxTry = 100

| lr | minerror |  |  |
| --- | --- | --- | --- |
| 0.0028541201 | 0.5918224098 |  | very good |
| 0.0001460131 | 2.6049065605 |  |  |
| 0.0065802148 | 0.5909805245 |  |  |
| 0.0001903743 | 2.2580700613 |  |  |
| 0.0013272848 | 0.6582979731 |  |  |
| 0.0027737964 | 0.5920841729 |  |  |
| 0.0000681090 | 4.1725647264 |  |  |
| 0.0036019238 | 0.5914668892 |  |  |
| 0.0000213591 | 6.6874692097 |  |  |
| 0.0000301828 | 5.9663858970 |  |  |

lr > 0.003 が望ましい

| lr | err\_th 0.5 収束時間秒 | 収束時 err |  |
| --- | --- | --- | --- |
| 0.0344172606 | 884.29521 | 0.8550014486 | ひどく振動  eeeeeeeeeeeeeeeeeeeeee^eeeeeeeeeeeeeeeeee^eeeeeeeeeee^eeeeeeeeeeeee^eeeeeeeeeeee^eeeeeeeeee^eeeee^eee^eee^eeeee^ee^eee^ee^ee^eee^e^ee^ee^e^e..e |
| 0.0151425331 | 983.51022 | 0.5418079229 | eeeeeeeee^ee^^ee^e^^^.e.e |
| 0.0190129042 | 983.79341 | 0.5438670902 | eeeeeeeeeeeeee^eeee^ee^eeeee^e^ee^^ee^ee^.e |
| 0.0118573449 | 983.10552 | 0.5301450948 | eeeeeeee^^.e |
| 0.0205106003 | 985.72771 | 0.5607251716 | eeeeeeeeeeeeee^eeeee^eeeee^eeeee^e^ee^^ee^eee^ee.e |
| 0.0030842670 | 973.48109 | 0.5526705799 | ........^^...... |
| 0.0043153706 | 959.05797 | 0.5457396624 | ........^....... |
| 0.0067004515 | 961.06170 | 0.5434525118 | ^..^...^^^...^^^ |
| 0.0189898118 | 947.93019 | 0.5393312432 | eeeeeeeeeeeeee^eeee^ee^eeeee^e^ee^^ee^ee^.e |
| 0.0109550559 | 955.57069 | 0.5367619834 | eeeeeeeすべて１６分程度 |

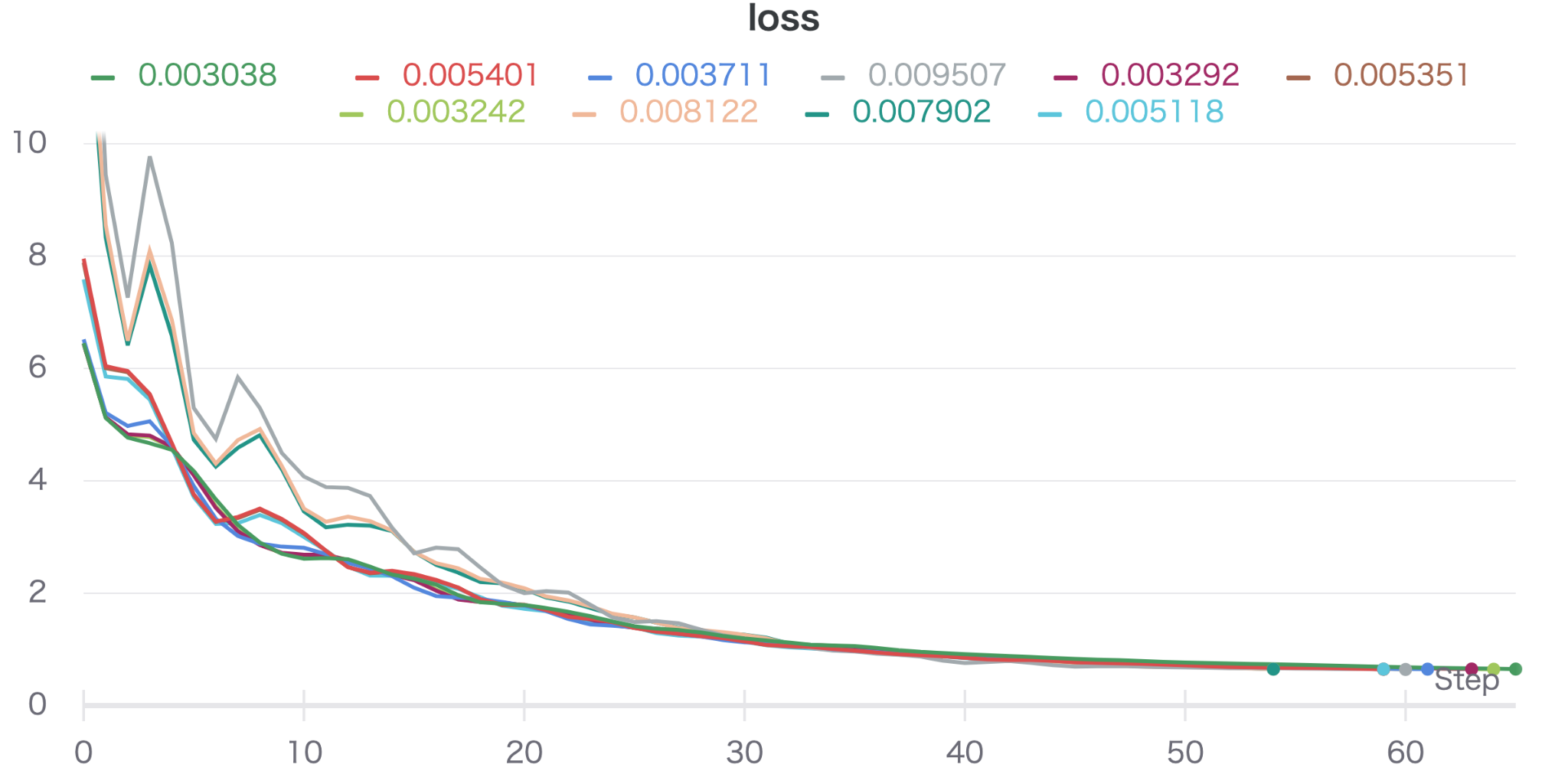
0.02 は大きすぎる 0.03 になると振動、順序関係の矛盾（e)が多く発生

0.006 以下ならばeは生じない

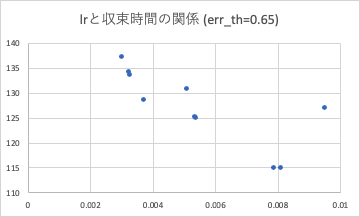
0.003以下は収束が遅くなる　0.003〜0.008 ぐらいが最適

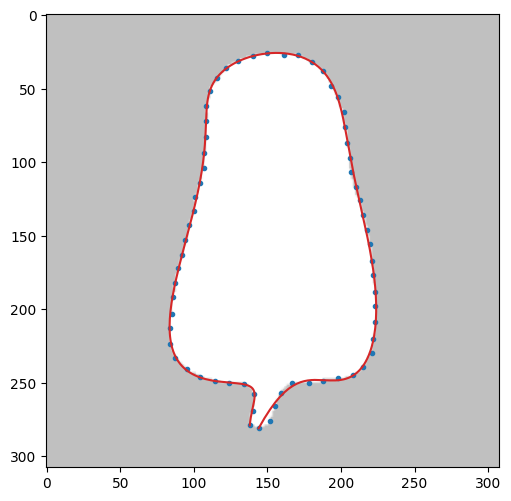
実験(2021.5.6.9am)　err\_th = 0.65 に設定し、lr の違いによる収束時間を比較

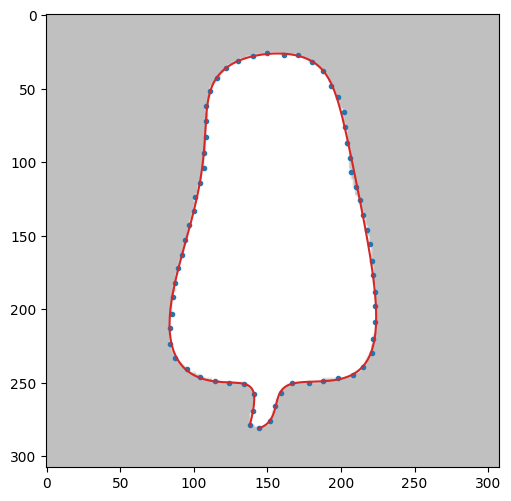
lr = trial.suggest\_loguniform('terminal\_coe', 0.003, 0.010) # 0.005



| lr | 収束時間 (s) | err |
| --- | --- | --- |
| 0.00303826 | 137.14629 | 0.64947406 |
| 0.00324207 | 134.13809 | 0.6462196 |
| 0.00329222 | 133.65216 | 0.64998078 |
| 0.00371147 | 128.57441 | 0.64787287 |
| 0.00511794 | 130.80077 | 0.64866376 |
| 0.00535114 | 125.16364 | 0.64645726 |
| 0.00540122 | 125.01721 | 0.6463978 |
| 0.00790223 | 114.88074 | 0.64840315 |
| 0.00950723 | 127.02249 | 0.6424904 |
| 0.0081218541, | 114.93304 | 0.64873077 |

やはり0.008ぐらいまでにしないといけない

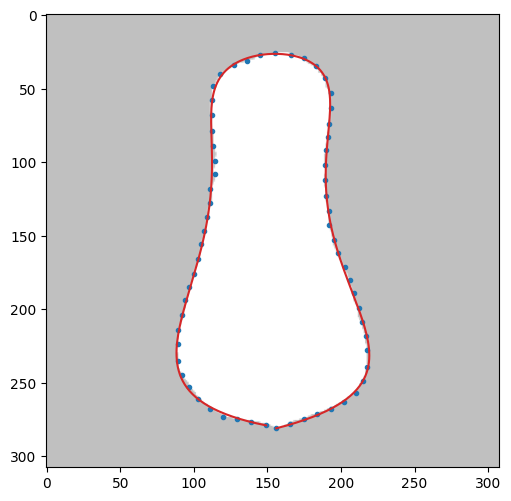
lr=0.0344172606の結果

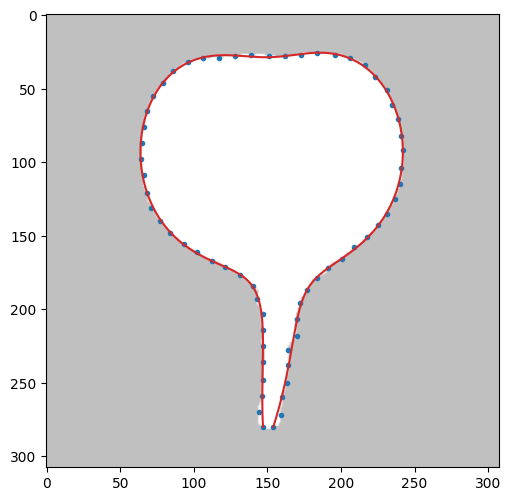
それ以外の結果はほぼ同等

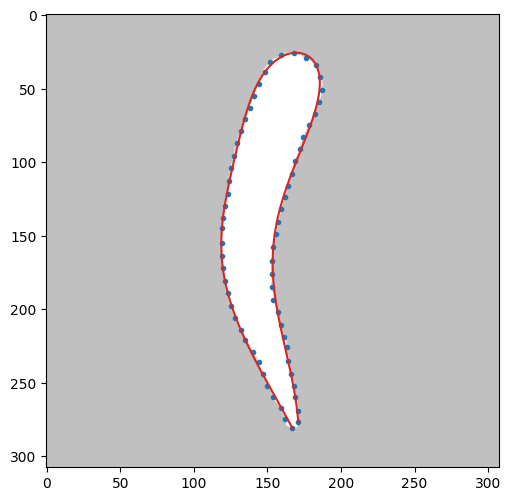
smoothness\_coe の効果　err\_th=0.65,lr=0.005　N=11 での収束時間

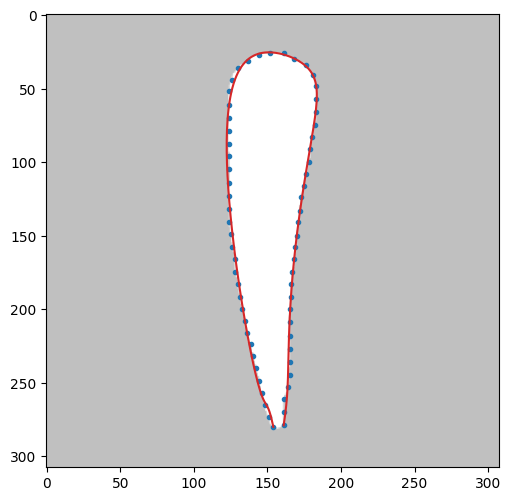
| coe | 収束時間 | フィッティング結果 |
| --- | --- | --- |
| 2.1074e-10 | 127.51835 |  |
| 0.19547e-08 | 116.18933 | good |
| 2.3216e-08 | 377.34347 | good |
| 2.6689e-08 | 374.13979 | OF |
| 0.09556e-08 | 115.85060 | OF |
| 0.12778e-08 | 115.58945 | good |
| 0.039327e-08 | 122.29038 |  |
| 0.200287e-08 | 111.39468 | good |
| 2.4808e-08 | 372.51741 |  |
| 0.05355e-08 | 120.02979 | good |

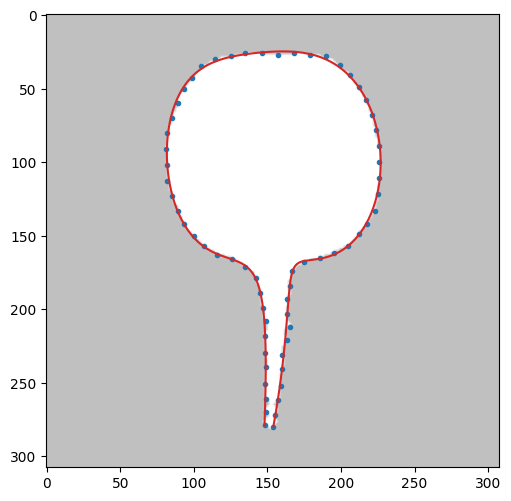
smoothness の効果はほとんど見受けられない

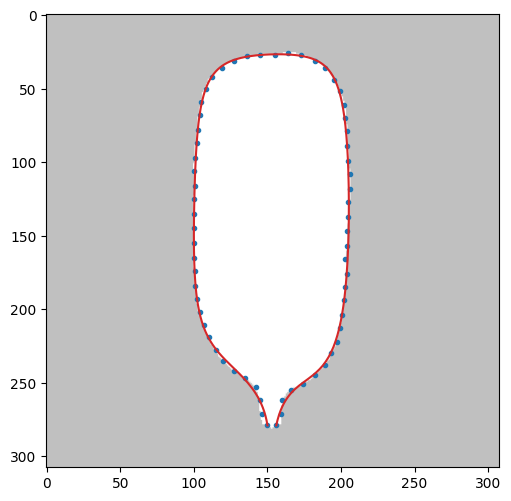
fit1T mode1 7次

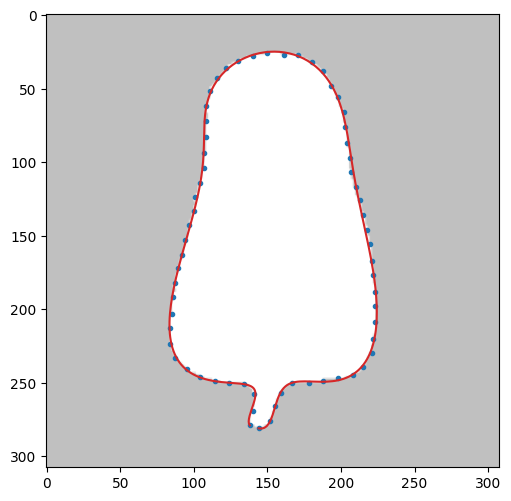
fit1T mode1 10次

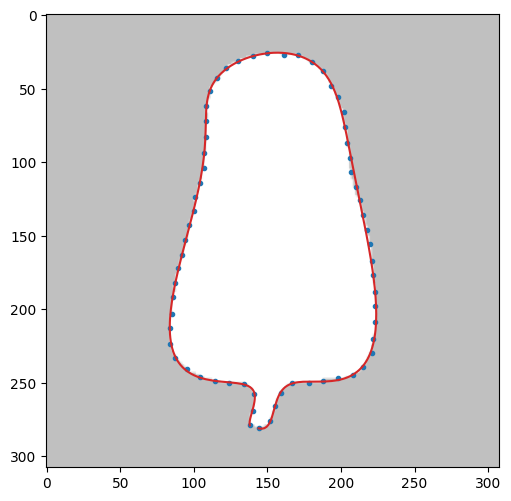
８次

８次

９次

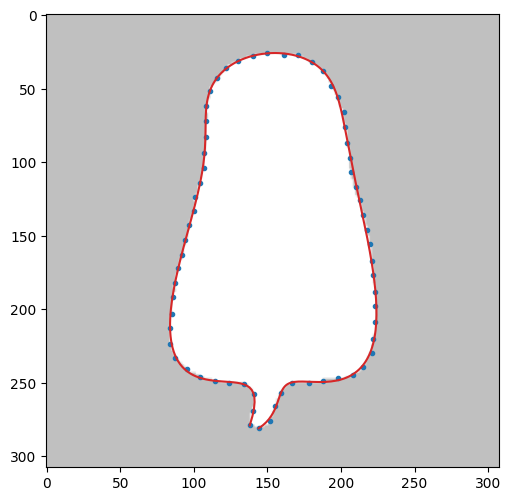
９次

12次

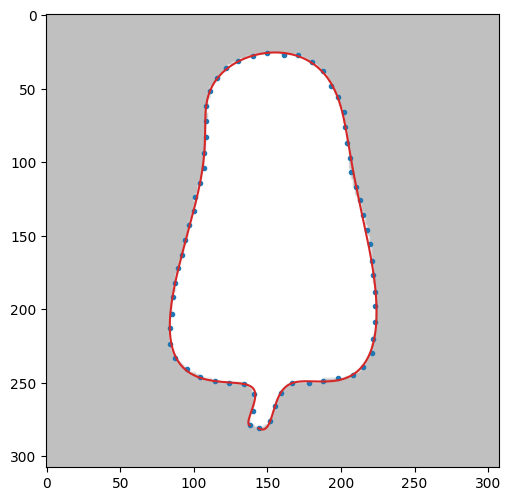
12次

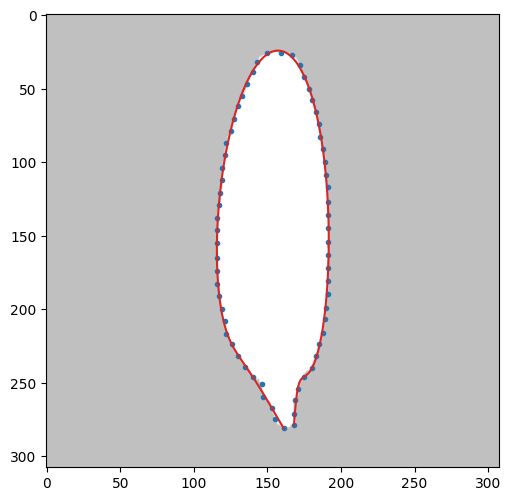
１１次でもOK

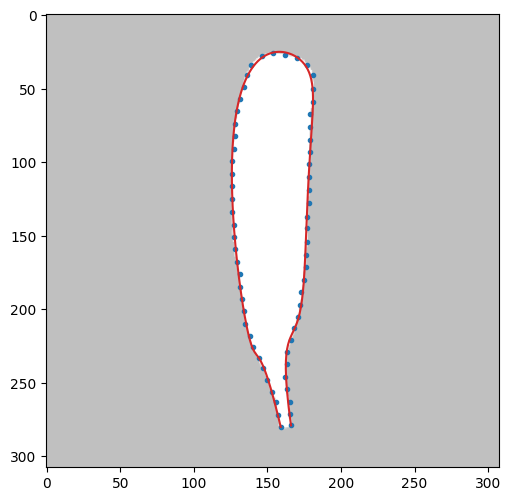
0.4975099

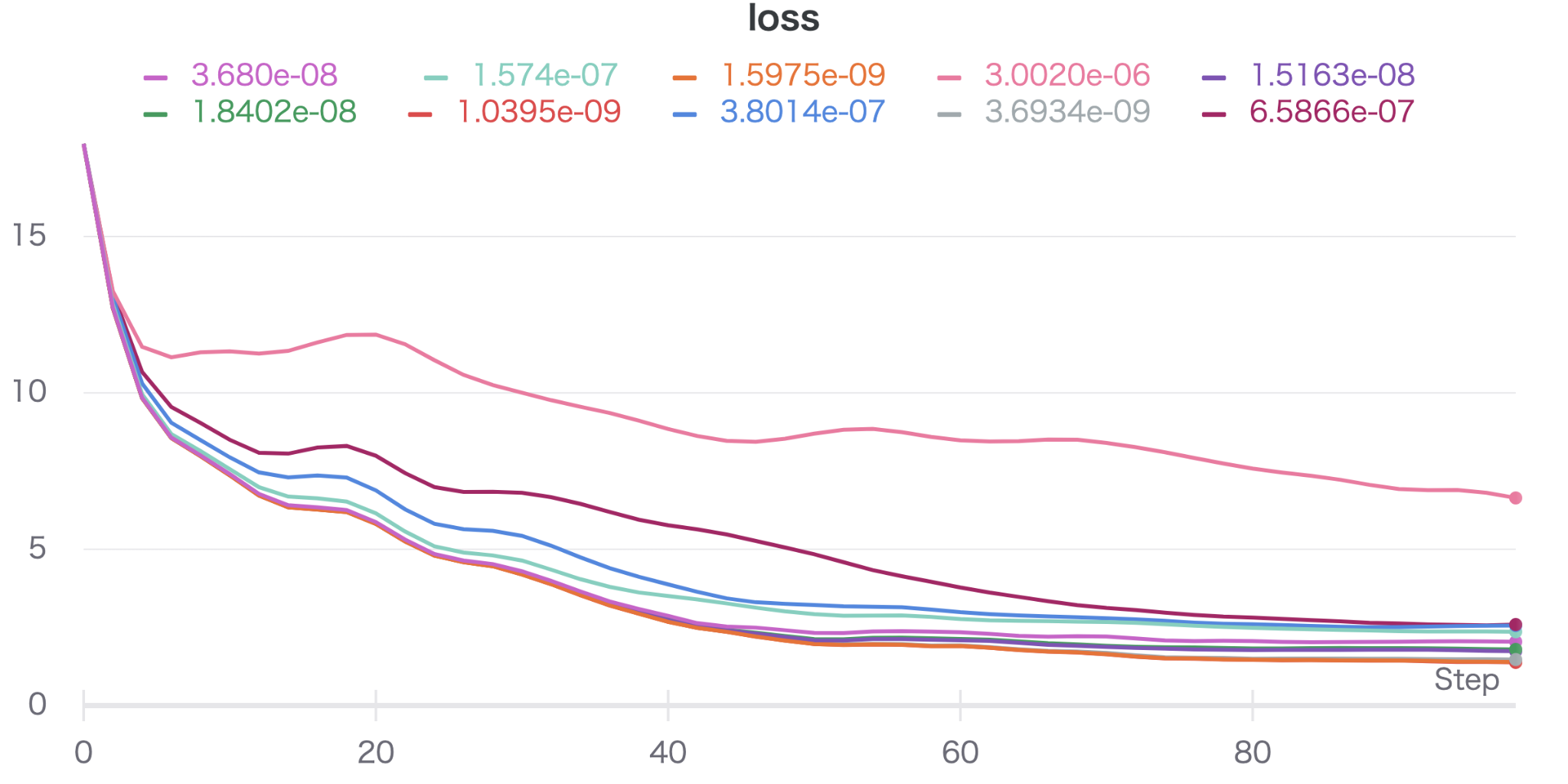
11次

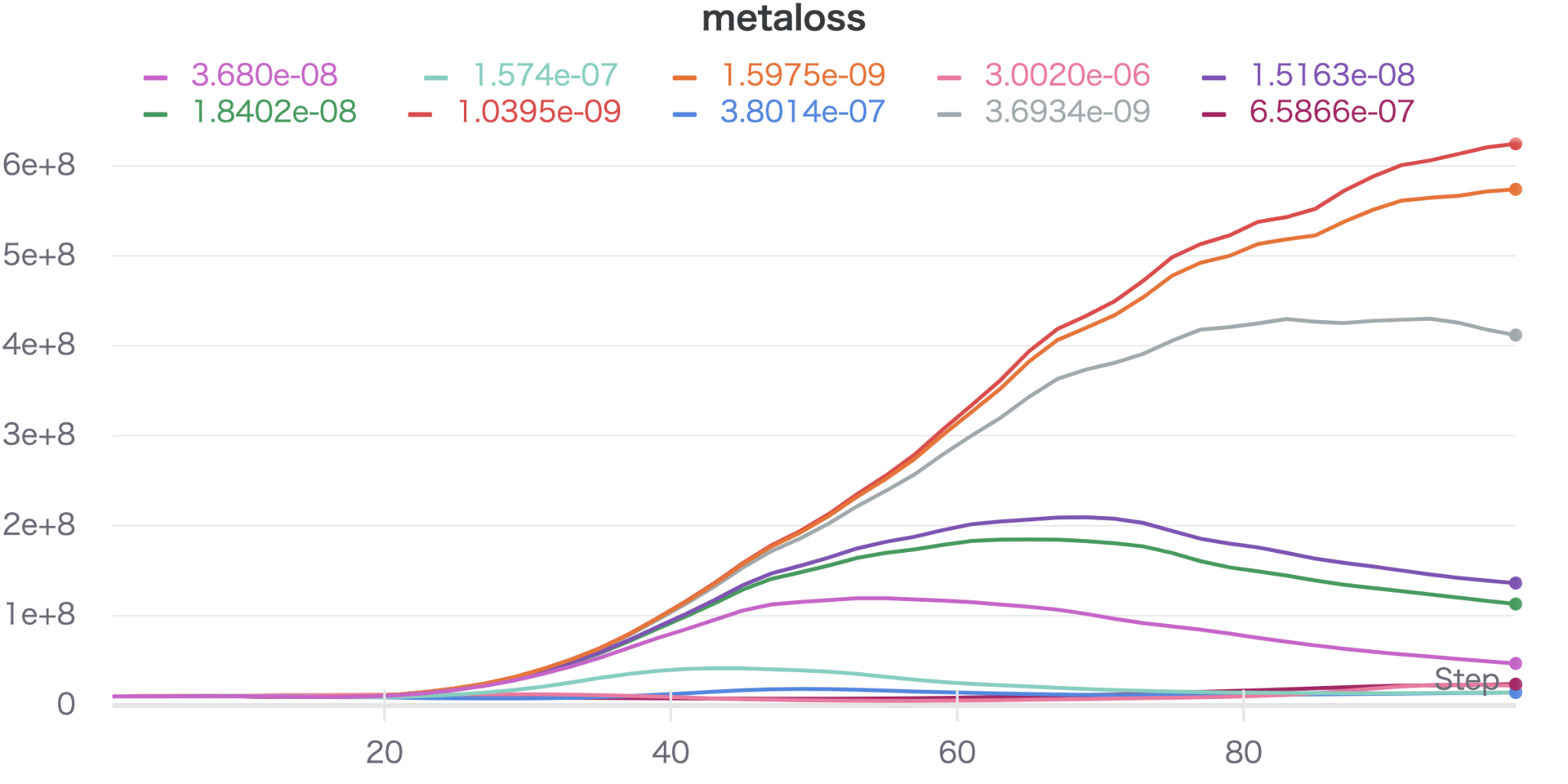
0.5407545

12次 0.49814615 2min 20s

７次 0.4999897

９次





| smoothness\_coe |  |  |  |
| --- | --- | --- | --- |
| 6.586564578243459e-07 | 2.564148965124464 |  | OF |
| 6.586564578243459e-07 | 2.564148965124464 |  | SOF |
| 3.6933601363340025e-09 | 1.4722884774278295 |  | good |
| 3.80141502263093e-07 | 2.5065857424925815 |  | OF |
| 1.8402417804542104e-08 | 1.7967793890147485 |  | good |
| 1.516301232379217e-08 | 1.7376481933506 |  | good |
| 3.0020303771102576e-06 | 6.639886332233251 |  | OF |
| 1.5975354970231803e-09 | 1.3923282063413096 |  | OF |
| 1.574667219483645e-07 | 2.36276069332265 |  | good |
| 3.6807662460385613e-08 | 2.0271223319262655 |  | good |
| 0.0000000994 | 2.2702063054824184 |  | good |
| 0.0000000266 | 1.90739 |  | good |
| 0.0000000089 | 1.64060 |  | good |
| 0.0000000021 | 1.40377 |  | OF |
| 0.0000000071 | 1.58647 |  | SOF |
| 0.0000000020 | 1.40141 |  | OF |
| 0.0000000011 | 1.38363 |  | OF |
| 0.0000000833 | 2.22156 |  | good |
| 0.0000000133 | 1.70966 |  | soso |
| 0.0000000086 | 1.63386 |  | soso |
| 0.0000004831 | 2.72839 |  |  |
| 0.0000003843 | 2.50645 |  | good |
| 0.0000000759 | 2.20979 |  | good |
| 0.0000000167 | 1.76482 |  | good |
| 0.0000007243 | 2.72346 |  | good |
| 0.0000000110 | 1.68172 |  | too bad |
| 0.0000000150 | 1.73451 |  | soso |
| 0.0000001724 | 2.38942 |  | soso |
| 0.0000000229 | 1.85921 |  | soso |
| 0.0000008547 | 2.64289 |  | good |

err\_th 0.5 N=8

|  |  |  |  |
| --- | --- | --- | --- |
| 0.0000000738 | 1.91933 |  | OF |
| 0.0000000537 | 1.32004 |  | OF |
| 0.0000000723 | 1.91397 |  | soso |
| 0.0000000547 | 1.33761 |  | OF |
| 0.0000000397 | 1.16590 |  | soso |
| 0.0000000446 | 1.24143 |  | soso |
| 0.0000000316 | 0.92182 |  | soso |
| 0.0000000419 | 1.20438 |  | soso |
| 0.0000000479 | 1.27821 |  | spsp |

smoothness 1e-8〜2e-8 は概ね良好 3e-8 以上はOF

|  |  |  |  |
| --- | --- | --- | --- |
| 0.0000000200 | 0.51614 |  | good |
| 0.0000000235 | 0.59484 |  | good |
| 0.0000000245 | 0.62884 |  | good |
| 0.0000000118 | 0.49490 |  | good |
| 0.0000000167 | 0.49928 |  | good |
| 0.0000000280 | 0.76567 |  | good |
| 0.0000000107 | 0.49910 |  | good |
| 0.0000000146 | 0.49371 |  | good |
| 0.0000000343 | 1.01984 |  | good |
| 0.0000000291 | 0.81443 |  | good |

| 0.0000000080 | 0.49127 |  | so good |
| --- | --- | --- | --- |
| 0.0000000090 | 0.49870 |  | so so good |
| 0.0000000097 | 0.49929 |  | so good |
| 0.0000000163 | 0.49871 |  | so good |
| 0.0000000061 | 0.49684 |  | so good |

1e-9〜5e-9　　3e-9 以下はあやしくなる　　結論 smoothness 制約は3e-9 以上必要

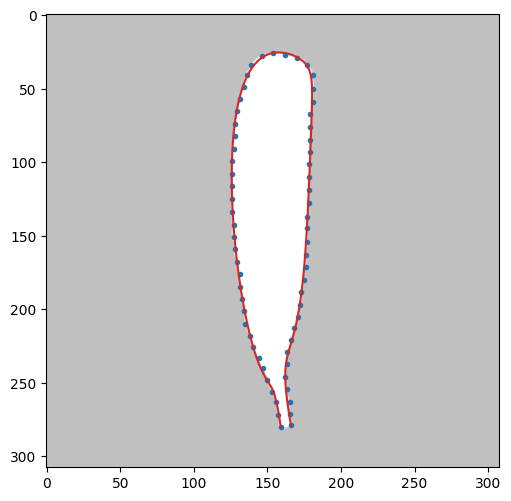
| 2.3700012258740725e-09 | 0.49912648715784214 |  | good |
| --- | --- | --- | --- |
| 2.3599638753782624e-09 | 0.4986945850844941 |  | good 左下が限界かも |
| 2.8255499294770663e-09 | 0.4914617609887752 |  | good 同上 |
| 1.343700656513839e-09 | 0.49341175477656996 |  | SOF　左下 |
| 3.1300484836698056e-09 | 0.49355551081022786 |  | good |

3e-9 以下になると左下があやうい

5e-10, 1e-9 1e-9 以下では平滑化制約が弱すぎてオーバーフィッティングが発生

| 2.3700012258740725e-09 | 0.49912648715784214 |  | OF bad |
| --- | --- | --- | --- |
| 2.3599638753782624e-09 | 0.4986945850844941 |  | OF bad |
| 2.8255499294770663e-09 | 0.4914617609887752 |  | OF bad |
| 1.343700656513839e-09 | 0.49341175477656996 |  | OF bad |
| 3.1300484836698056e-09 | 0.49355551081022786 |  | OF bad |

結論　　No.8 のダイコンは smoothness 2.5e-9 〜 3e-8 の範囲で良好なフィッティング



best one

8次 smoothness 1e-8

terminal coe の実験 smoothness は０

rd.BezierCurve.smoothness\_coe = 0 # 1e-20 # 1e-8 #2e-8

err = check1T(8,N=9,pat=100,maxTry=100,err\_th=0.5,lr=0.005,test='terminal\_coe',wandb=wandb)

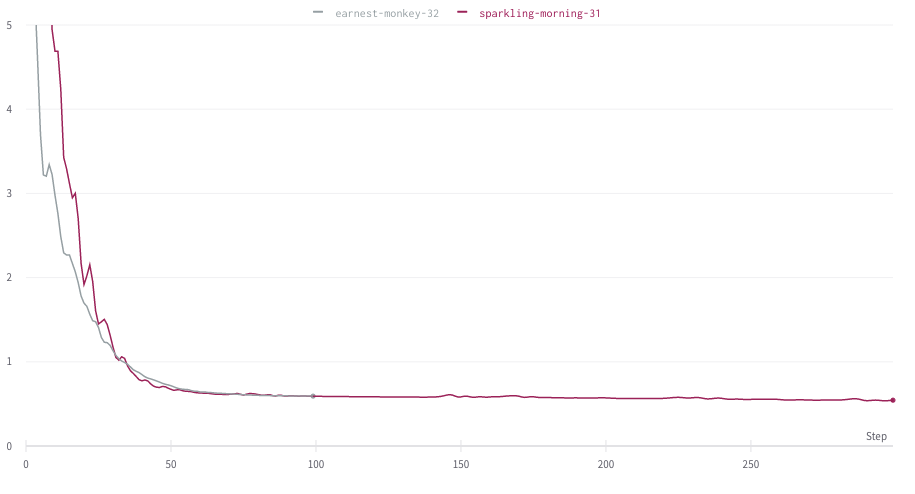
| terminal\_coe | N | error |  |  |
| --- | --- | --- | --- | --- |
| 49.5660407042 | ９ | 2.3881950459691983 |  |  |
| 0.2111312012 |  | 0.7864786674776356 |  | good |
| 0.6974528000 |  | 0.8024464183995887 |  | good |
| 44.7016259163 |  | 2.3812060481895525 |  |  |
| 70.2419058191 |  | 2.38657462493935 |  |  |
| 0.5504384459 |  | 0.7859170942714826 |  | good |
| 6.8862369376 |  | 2.471422893983863 |  |  |
| 4.1562354882 |  | 2.2878190684291426 |  |  |
| 60.5948353648 |  | 2.3845573544144525 |  |  |
| 94.6646906592 |  | 2.3851447896073648 |  |  |

必要条件 terminal coe は4以下であること

1e-3 〜4 以下で実験　maxTry 100

| 0.0257214365 | 9 | 0.7951885116034759 |  | good |
| --- | --- | --- | --- | --- |
| 0.1025228302 |  | 0.7910269047289854 |  | good |
| 0.3161869134 |  | 0.7846896007723405 |  | good |
| 2.9627795323 |  | 2.287294440471791 |  | bad |
| 0.1140047327 |  | 0.7903938027915831 |  | good |
| 0.0048145036 |  | 0.7972822665305982 |  | good |
| 0.3723034151 |  | 0.7855772637296914 |  | good |
| 0.0024944697 |  | 0.7975029778090067 |  | soso |
| 0.0701450567 |  | 0.7927160092506496 |  | soso |
| 0.012581165907359368 | 9 | 0.7967811811159305 |  | soso |

terminal coe は 0.3 以下が望ましい。



0.5までエラーを減少させようと思うと非常に時間がかかるが、0.65で妥協すればかなり早く打ち切れる

img4 lr に関する実験

%%time

import tensorflow as tf

def objective(trial):

with tf.device("GPU:0"):

lr = trial.suggest\_loguniform('terminal\_coe', 1e-5, 1e-2) # 0.005

print('lr: %1.10f' % lr)

rd.BezierCurve.debugmode=False

rd.BezierCurve.convg\_coe = 1e-5

rd.BezierCurve.terminal\_coe = 0

#rd.BezierCurve.smoothness\_coe = 0

rd.BezierCurve.smoothness\_coe = 0 # 1e-20 # 1e-8 #2e-8

err = check1T(6,N=11,pat=100,maxTry=100,err\_th=0.3,lr=lr,test='lr',wandb=wandb)

print('lr: %1.10f,minerror: %1.10f' % (lr, err))

return err

study = optuna.create\_study()

study.optimize(objective, n\_trials=10)

| 0.0028541201,minerror: | 0.5918224098 |  | very good |
| --- | --- | --- | --- |
| 0.0001460131 | 2.6049065605 |  | まだフィットしていない。右下がとくに |
| 0.0065802148 | 0.5909805245 |  | そこそこ |
| 0.0001903743 | 2.2580700613 |  | まだフィットしていない。右下がとくに |
| 0.0013272848 | 0.6582979731 |  | そこそこ |
| 0.0027737964 | 0.5920841729 |  | very good |
| 0.0000681090 | 4.1725647264 |  | まだまだ |
| 0.0036019238 | 0.5914668892 |  | very good |
| 0.0000213591 | 6.6874692097 |  | かなりまだ |
| 0.0000301828 | 5.9663858970 |  | かなりまだ |

収束を早めるには、lr > 0.003 が望ましい

lr は 0.003 〜 0.008

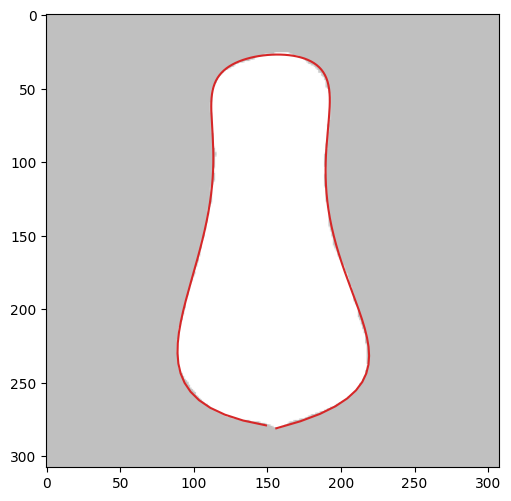
rd.BezierCurve.terminal\_coe は 0.1〜0.3 の範囲だと影響はほとんどなし

（時間、結果）

大根０



５、６、７次で平均誤差１まで収束につき、7次で十分



check0(0,maxTry=300,coe = 2.5e-8,N=7,err\_th=0.5,pat=100)

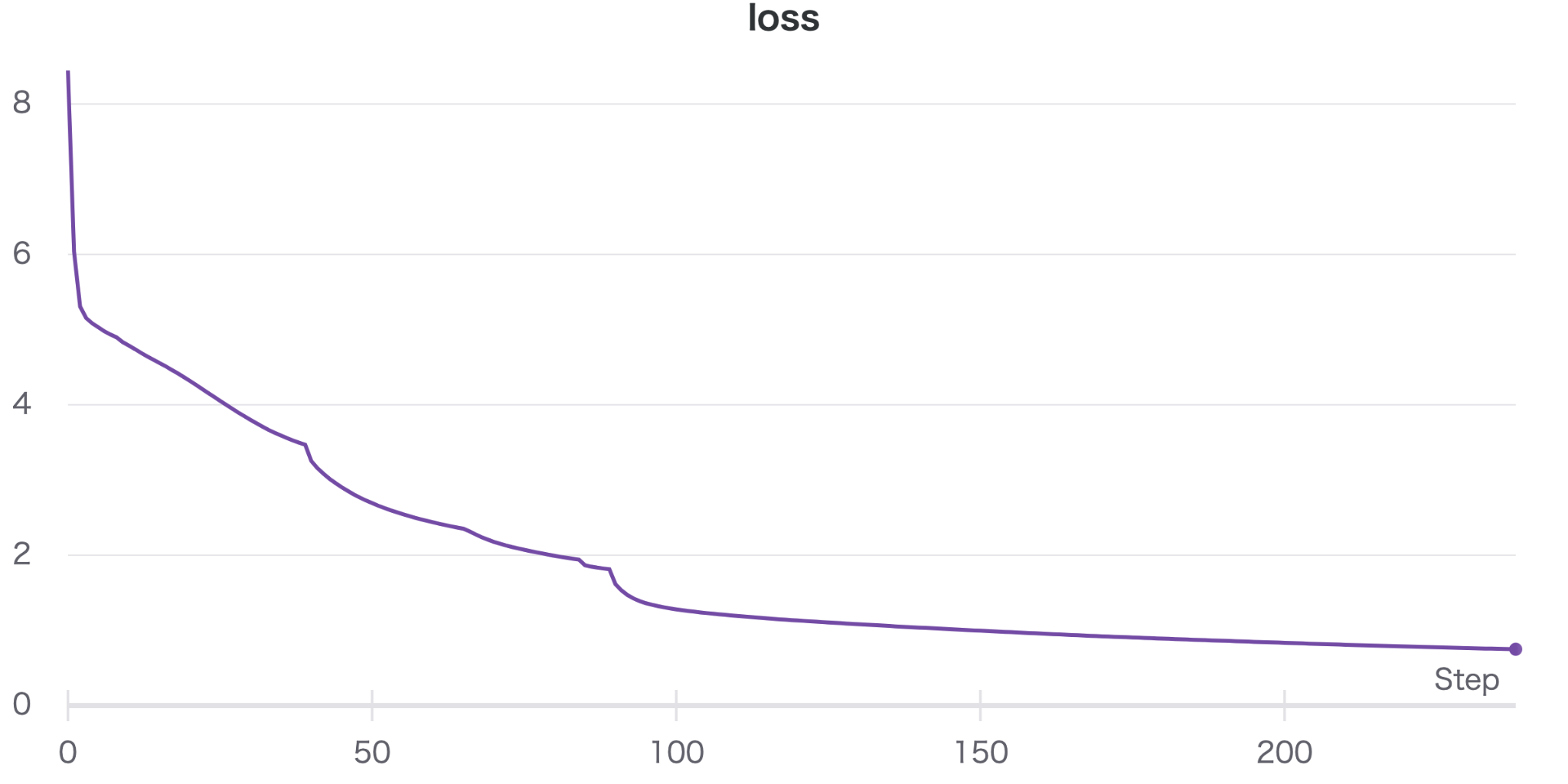
5-7-10 10次

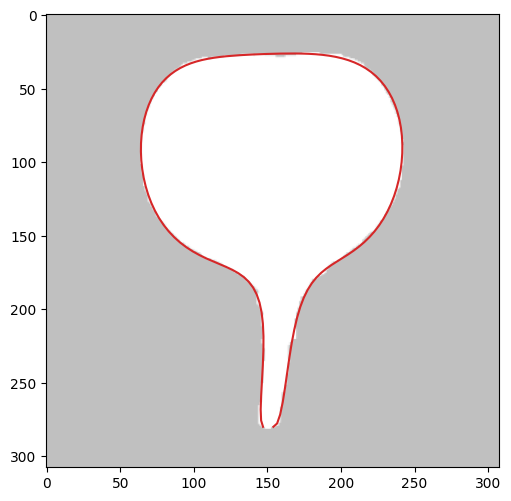
| loss | 0.68498 |
| --- | --- |

CPU times: user 4min 2s, sys: 1.46 s, total: 4min 4s

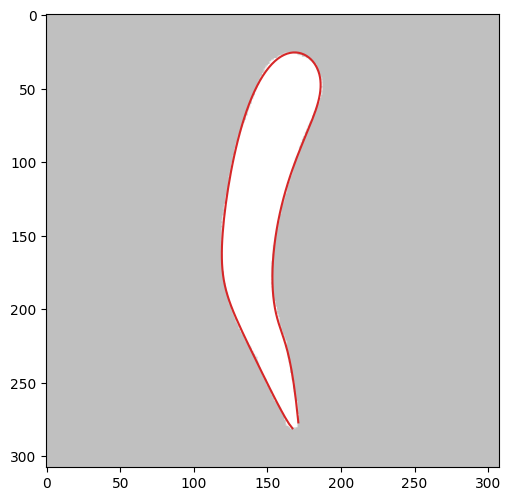
Wall time: 4min 9s

大根２





大根２



check0(2,maxTry=300,coe = 2.5e-8,N=9,err\_th=0.5,pat=100)

0.51995623

def check0(imgnum,mode=0,coe=0,samples=65,N=7,lr=0.02,preTry=100,maxTry=1500,pat=10,err\_th=0.75,threstune=1.0,epsilon=0):

wandb.init(project="fit1T optimize %d" % imgnum)

#調整するパラメータを登録

config = {"eq\_coe": coe}

wandb.config.update(config)

rd.BezierCurve.wandb=wandb

# rd.BezierCurve.eq\_coe = coe

img = imgs[imgnum]

conAll = rd.getCntPairWithImg(img, mode=1) # 全周輪郭を得る

Samples = rd.getSamples(conAll,N=samples,mode='Equidistant')

bez = rd.BezierCurve(N=N,samples=Samples) # インスタンス生成

bestcps, bestfunc, minerror = bez.fit1T(maxTry=maxTry, lr=lr,withErr=True, tpara=[], pat=pat, err\_th=err\_th, threstune=threstune)

rd.drawBez(img,stt=0.0,end=1.0,bezL=bestfunc)

return minerror

1e-8〜6e-8 1e-9台まで下げると　端点部が不自然になる

check0(2,maxTry=200,coe = coe,N=9,err\_th=0.75,pat=100)

rd.BezierCurve.convg\_coe = 0.0001

rd.BezierCurve.terminal\_coe = coe

rd.BezierCurve.eq\_coe = c= 0.0

実験１用プログラム

def fit0andDraw(imgnum,Nsamples=65,N=7,withFig=False):

img = imgs[imgnum]

conAll = rd.getCntPairWithImg(img, mode=1) # 全周輪郭を得る

Samples = rd.getSamples(conAll,N=Nsamples,mode='Equidistant')

bez = rd.BezierCurve(N=N,samples=Samples) # インスタンス生成

cps, func = bez.fit0(tpara=[])

if withFig:

rd.drawBez(img,stt=0.0,end=1.0,bezL=func)

def meanerr(func, ts):

sps = Samples

fx, fy = func

t = symbols('t')

nfx, nfy = lambdify(t, fx, "numpy"), lambdify(t, fy, "numpy")

onps = [[nfx(ts[i]), nfy(ts[i])] for i in range(len(ts))]

return mean([np.sqrt((sps[i][0]-onps[i][0])\*\*2+(sps[i][1]-onps[i][1])\*\*2) for i in range(len(sps))])

return func, meanerr(func,ts=bez.ts)

def check0N(imgnum,Nsamples=65,Nfrom=5,Nto=20,err\_th=0.5):

for N in range(Nfrom,Nto+1):

print(N,end='')

func, err = fit0andDraw(imgnum,Nsamples=Nsamples,N=N,withFig=False)

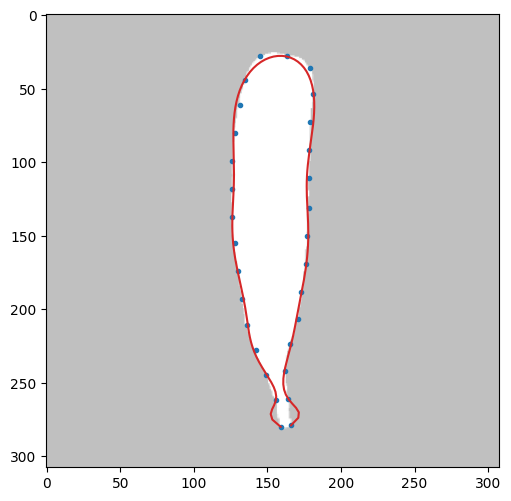
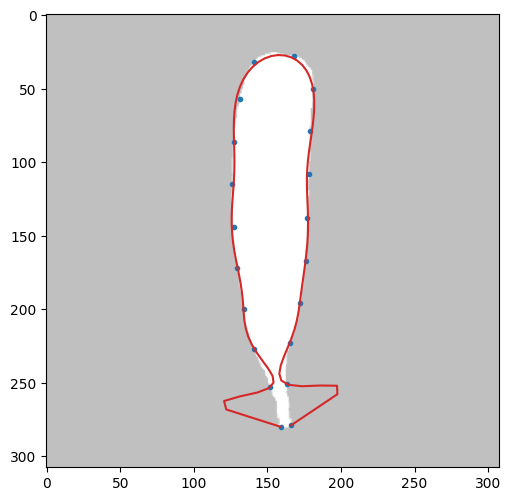
if err < err\_th:

break

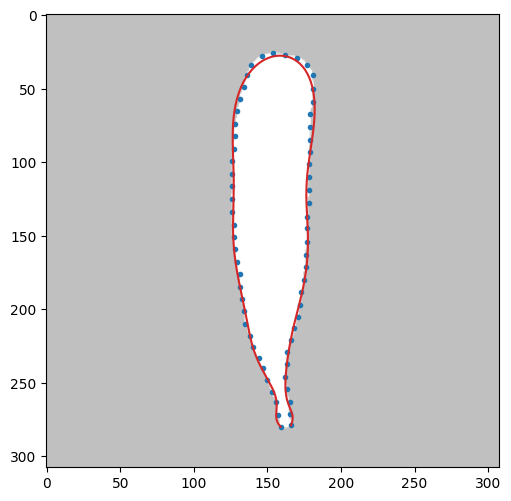
print("\nN= {} err = {}".format(N,err))

rd.drawBez(imgs[imgnum],stt=0.0,end=1.0,bezL=func)

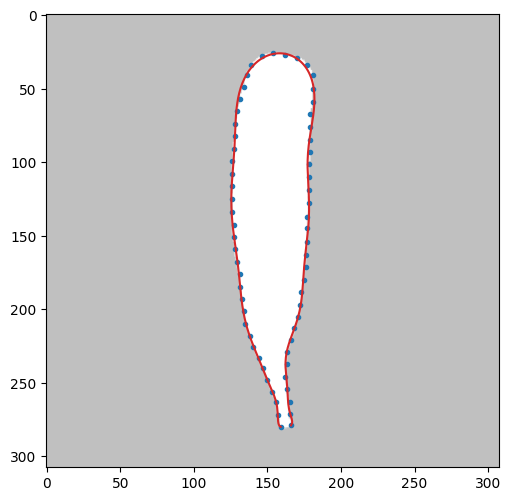
オーバーフィッティング



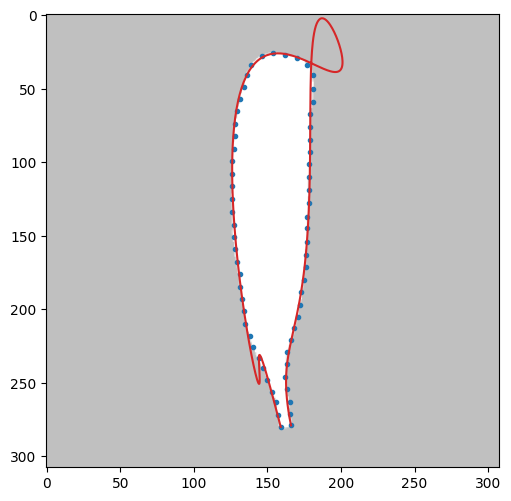
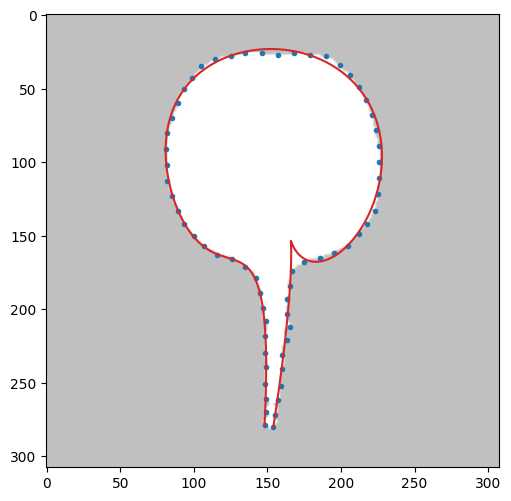
20サンプル、15次 20サンプル　15次



65サンプル 15次　fit0



65サンプル 20次　fit0 平均誤差 0.71

8次

terminal coe の効果

rd.BezierCurve.convg\_coe = 1e-5

rd.BezierCurve.swing\_penalty = 0

rd.BezierCurve.smoothness\_coe = 0

err,etime = check1T(1,mode=1,N=10,pat=200,maxTry=300,err\_th=0.65,lr=0.005,lrP=650,test=obj,coe=0,wandb=wandb)

の実行結果は次の通り

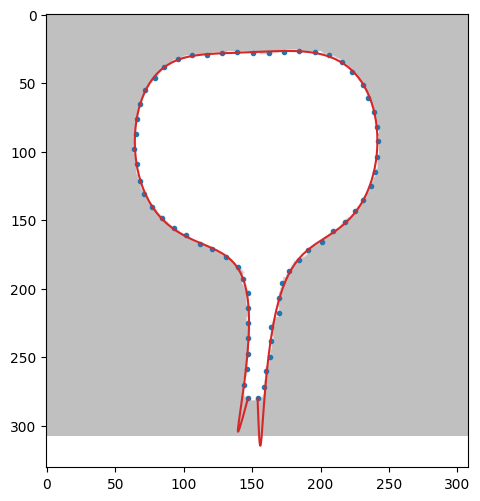
Matrix([[154.0\*t\*\*10 + 166.994117185962\*t\*\*9\*(10 - 10\*t) + 5785.63641900107\*t\*\*8\*(1 - t)\*\*2 + 44409.5360974933\*t\*\*7\*(1 - t)\*\*3 + 157931.234563304\*t\*\*6\*(1 - t)\*\*4 - 270460.416332542\*t\*\*5\*(1 - t)\*\*5 + 329146.28477631\*t\*\*4\*(1 - t)\*\*6 - 148812.800902267\*t\*\*3\*(1 - t)\*\*7 + 28759.8221731705\*t\*\*2\*(1 - t)\*\*8 + 642.025678850341\*t\*(1 - t)\*\*9 + 147.0\*(1 - t)\*\*10], [280.0\*t\*\*10 + 671.371288264853\*t\*\*9\*(10 - 10\*t) - 88926.7645766488\*t\*\*8\*(1 - t)\*\*2 + 655881.272498875\*t\*\*7\*(1 - t)\*\*3 - 1905169.14501086\*t\*\*6\*(1 - t)\*\*4 + 2670170.81890784\*t\*\*5\*(1 - t)\*\*5 - 1863654.41512575\*t\*\*4\*(1 - t)\*\*6 + 626662.983616024\*t\*\*3\*(1 - t)\*\*7 - 80174.5178438124\*t\*\*2\*(1 - t)\*\*8 + 5915.67220987861\*t\*(1 - t)\*\*9 + 280.0\*(1 - t)\*\*10]])

smoothness: 4.7448497528e-07,minerror: 0.6422759742, etime: 184.85415

CPU times: user 3min 2s, sys: 29.4 s, total: 3min 31s

Wall time: 3min 4s

error = 0.6423



終端部でかなりオーバーフィッティングが生じる

rd.BezierCurve.swing\_penalty = 300

rd.BezierCurve.smoothness\_coe = 4e-7

で実行

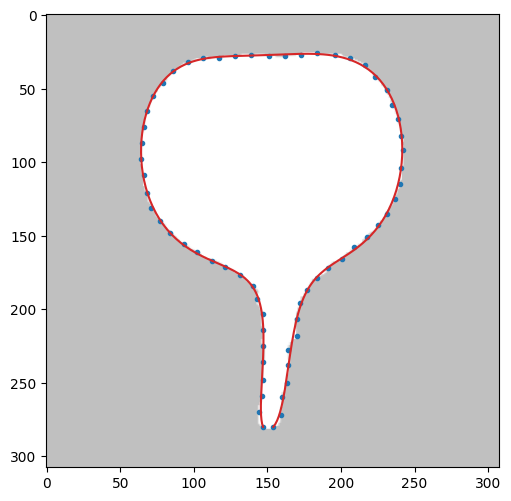
Matrix([[154.0\*t\*\*10 + 200.804450462266\*t\*\*9\*(10 - 10\*t) + 2652.84197363112\*t\*\*8\*(1 - t)\*\*2 + 77532.9836403768\*t\*\*7\*(1 - t)\*\*3 + 35962.1564936634\*t\*\*6\*(1 - t)\*\*4 - 96539.876775727\*t\*\*5\*(1 - t)\*\*5 + 202237.656634998\*t\*\*4\*(1 - t)\*\*6 - 92538.6802572469\*t\*\*3\*(1 - t)\*\*7 + 16745.969464831\*t\*\*2\*(1 - t)\*\*8 + 1228.18584278464\*t\*(1 - t)\*\*9 + 147.0\*(1 - t)\*\*10], [280.0\*t\*\*10 + 227.295645958876\*t\*\*9\*(10 - 10\*t) - 23236.9791223218\*t\*\*8\*(1 - t)\*\*2 + 290994.337213\*t\*\*7\*(1 - t)\*\*3 - 980812.935684398\*t\*\*6\*(1 - t)\*\*4 + 1450402.96578148\*t\*\*5\*(1 - t)\*\*5 - 988234.665120427\*t\*\*4\*(1 - t)\*\*6 + 296963.198831202\*t\*\*3\*(1 - t)\*\*7 - 23113.9641966071\*t\*\*2\*(1 - t)\*\*8 + 2205.79804177853\*t\*(1 - t)\*\*9 + 280.0\*(1 - t)\*\*10]])

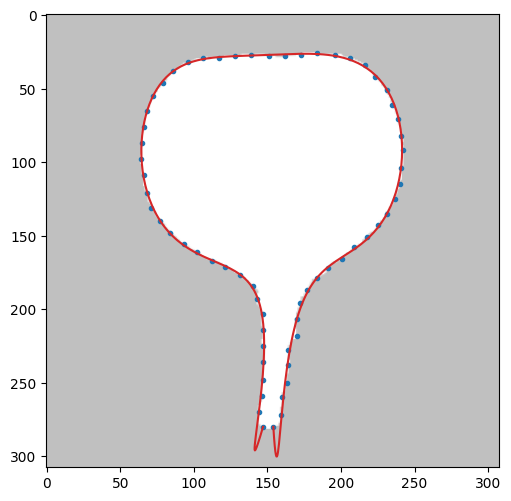
smoothness: 9.1071344083e-07,minerror: 0.6427349873, etime: 244.88003

CPU times: user 4min 2s, sys: 37.2 s, total: 4min 39s

Wall time: 4min 4s

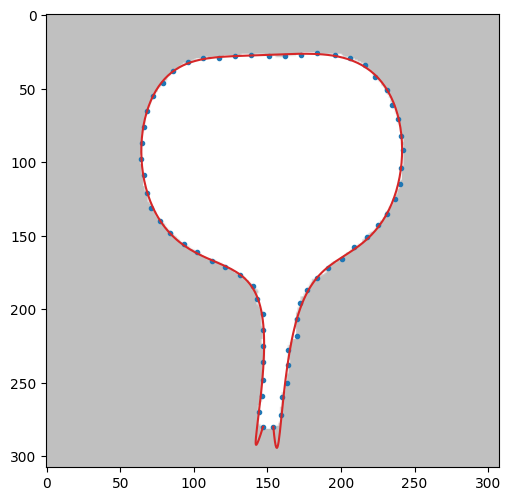
error = 0.6427





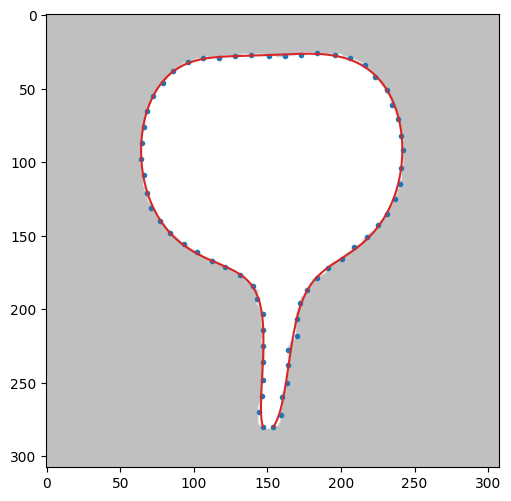
terminal\_coe: 3.5336891028e-01,

minerror: 0.6458442656, etime: 152.32591



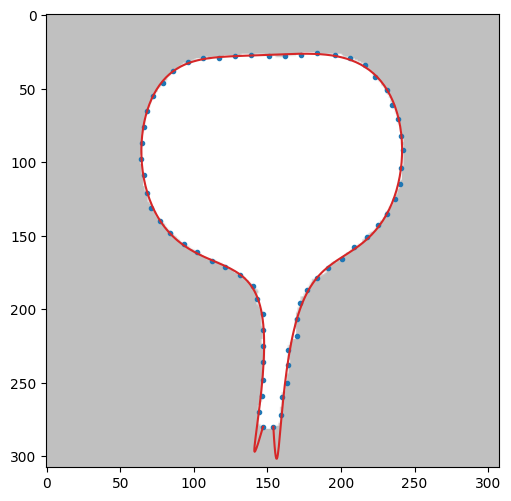
terminal\_coe: 1.5340255679e+00,

minerror: 0.6413759578, etime: 181.08036



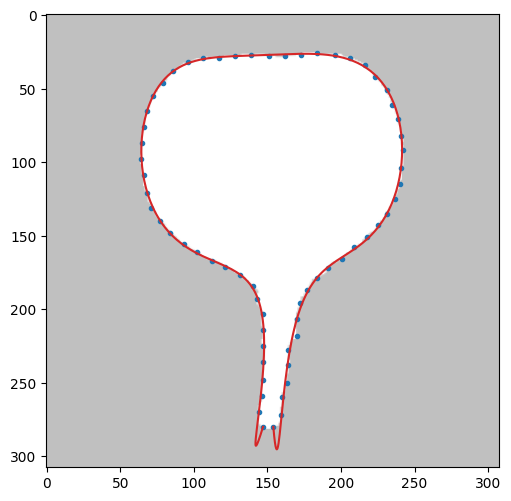
terminal\_coe: 2.1161439478e-01,

minerror: 0.6499198286, etime: 147.45584



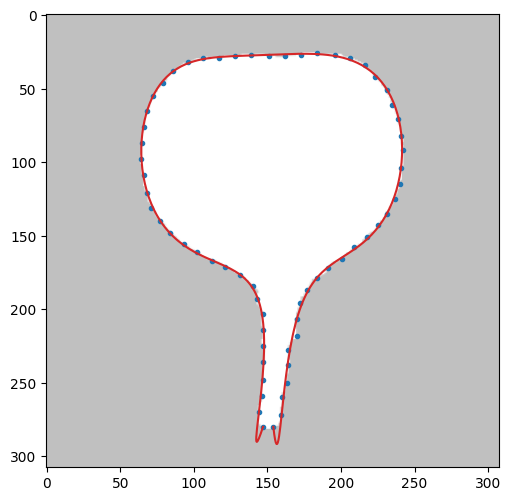
terminal\_coe: 3.3250803257e-01,

minerror: 0.6419142300, etime: 152.78088



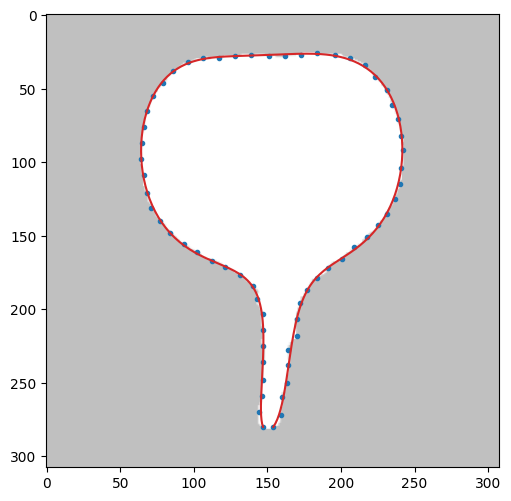
terminal\_coe: 4.3114690645e-01,

minerror: 0.6490682517, etime: 156.65873



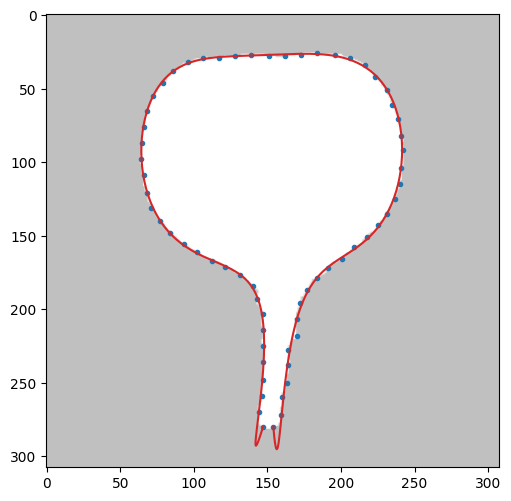
terminal\_coe: 1.4648450442e+00,

minerror: 0.6487819328, etime: 183.06500



terminal\_coe: 3.3443835195e-01,

minerror: 0.6423785105, etime: 153.40722



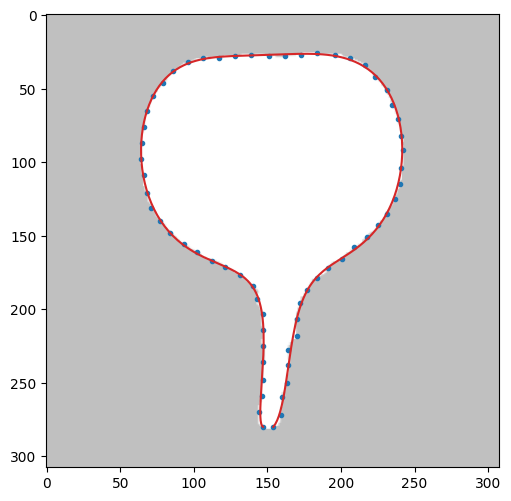
terminal\_coe: 4.8703187498e-01,

minerror: 0.6477167762, etime: 155.77645



terminal\_coe: 1.0001687666e+00,

minerror: 0.6462643480, etime: 177.99563



良かった３つ

terminal\_coe: 1.5340255679e+00,

minerror: 0.6413759578, etime: 181.08036

terminal\_coe: 1.0001687666e+00,

minerror: 0.6462643480, etime: 177.99563

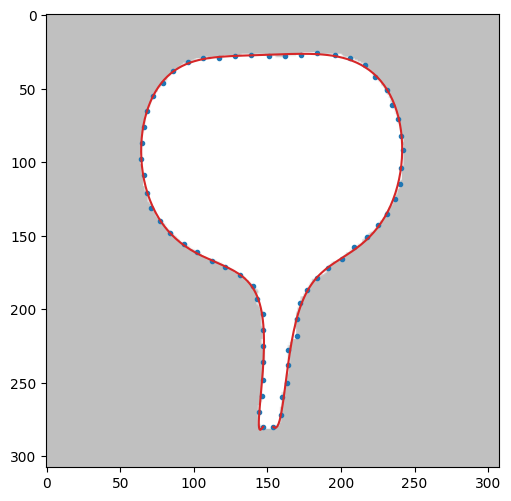
terminal\_coe: 1.4648450442e+00,

minerror: 0.6487819328, etime: 183.06500

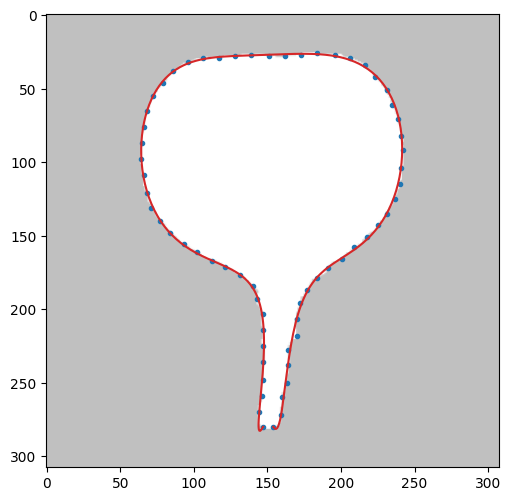
1.0〜1.5 を中心に再実験　0.5〜2.0

結果

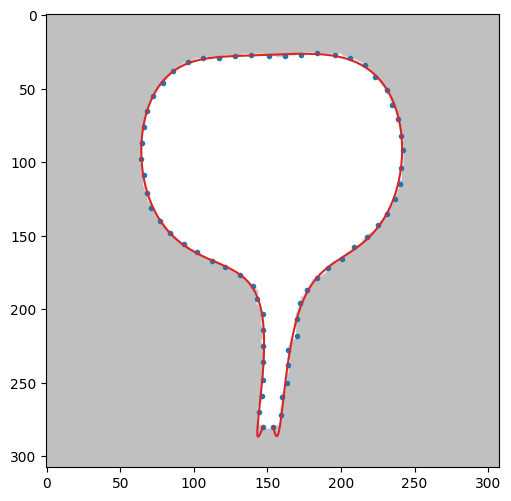
terminal\_coe: 7.4505376499e-01,minerror: 0.6469962478, etime: 179.38195



terminal\_coe: 7.0752965535e-01,minerror: 0.6469344590, etime: 165.57612



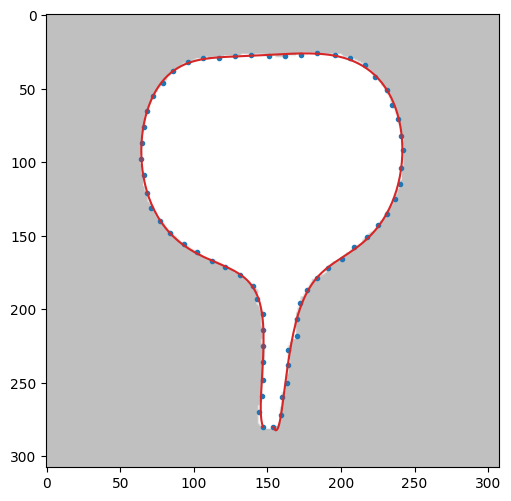
terminal\_coe: 5.3273954571e-01,minerror: 0.6412563991, etime: 160.24375



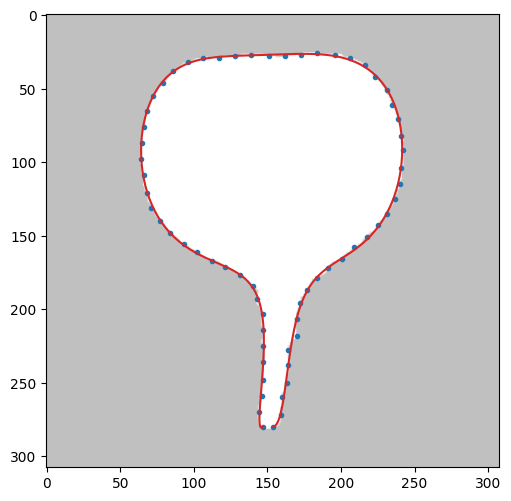
terminal\_coe: 1.2746571724e+00,minerror: 0.6446195983, etime: 179.90583

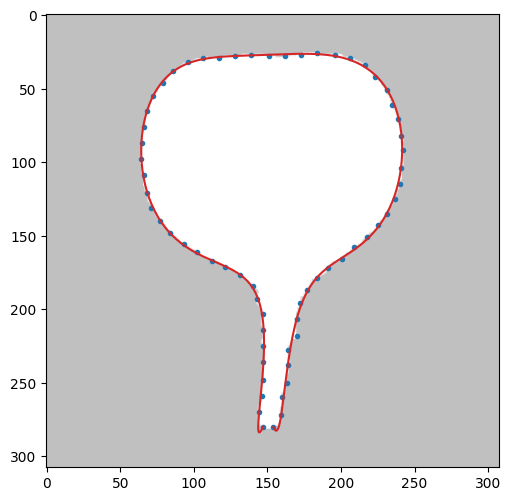


terminal\_coe: 1.6972604601e+00,minerror: 0.6401951845, etime: 179.67000

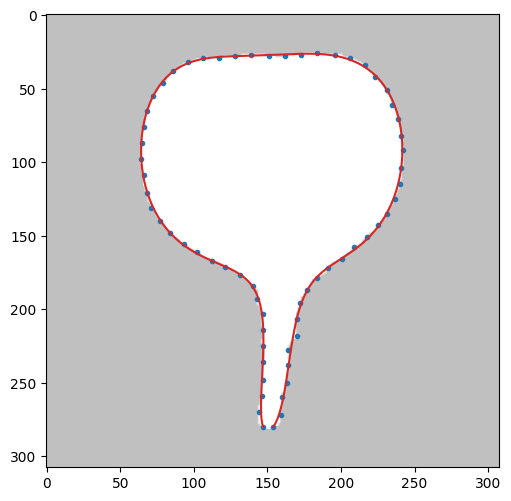


terminal\_coe: 8.3912279500e-01,minerror: 0.6487865609, etime: 171.71978

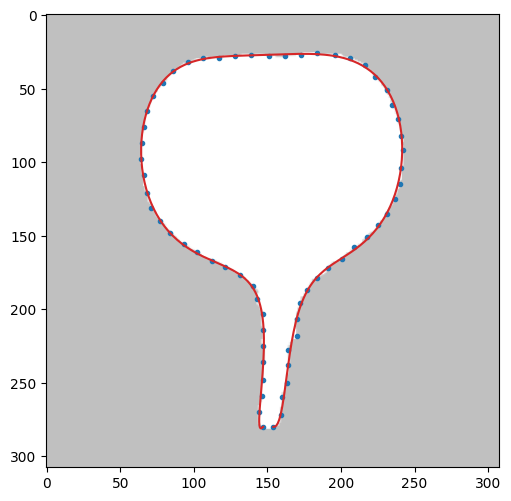
わずかにterminal\_coe: 6.5606923386e-01,minerror: 0.6452219051, etime: 163.17998



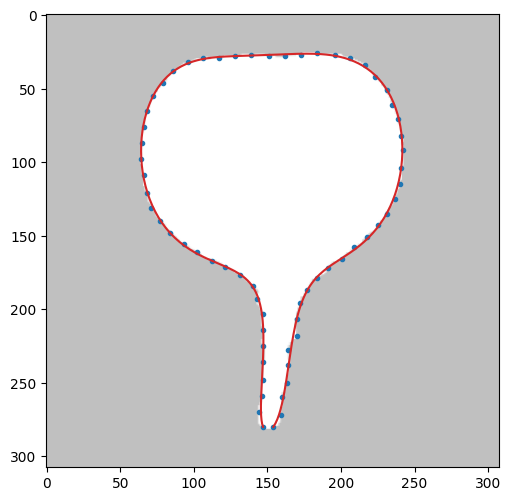
terminal\_coe: 1.6920460918e+00,minerror: 0.6467653621, etime: 183.59584



terminal\_coe: 7.9937989236e-01,minerror: 0.6454095578, etime: 168.94331



terminal\_coe: 1.4622870547e+00,minerror: 0.6487592490, etime: 179.06066



よかった３つ

terminal\_coe: 1.2746571724e+00,minerror: 0.6446195983, etime: 179.90583

terminal\_coe: 1.4622870547e+00,minerror: 0.6487592490, etime: 179.06066

terminal\_coe: 1.6920460918e+00,minerror: 0.6467653621, etime: 183.59584

0.83 がニアミス　1.69もニアミス

smooth\_coe: 7.5974792122e-08,minerror: 0.7020427553, etime: 466.12592

smooth\_coe: 4.7724007308e-08,minerror: 0.6490451257, etime: 195.17984

smooth\_coe: 8.5472130908e-08,minerror: 0.7440276658, etime: 483.40925

smooth\_coe: 1.2417526538e-08,minerror: 0.6494405386, etime: 152.97479

smooth\_coe: 2.8675544408e-08,minerror: 0.6493381263, etime: 172.61314