predict sin

In [1]:

```
import sys, os
sys.path.append(os.pardir)  # 親ディレクトリのファイルをインポートするための設定
import numpy as np
from common import functions
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split

np.random.seed(0)

# sin曲線
round_num = 10#10πだと、5周。
div_num = 500#分割数　5周を500分割だと、1周100分割。1分割3.6度。
ts = np.linspace(0, round_num * np.pi, div_num)
f = np.sin(ts)

#tanhの導関数
def d_tanh(x):
    return 1/(np.cosh(x)**2 + 1e-4)

# ひとつの時系列データの長さ
maxlen = 2

# sin波予測の入力データ
test_head = [[f[k]] for k in range(0, maxlen)]#配列fのindexの０～k－1番目の値を抽出する。

#print(f)
print(test_head)
```

```
[[0.0], [0.06291618610288617]]
```

In [2]:

```
data = []
target = []

for i in range(div_num - maxlen):#500-2
    data.append(f[i: i + maxlen])#0～1,1～2,2～3,
    target.append(f[i + maxlen])#    2,    3,    4,

X = np.array(data).reshape(len(data), maxlen, 1)
D = np.array(target).reshape(len(data), 1)
```

In [3]:

```
print(X)
```

```
[[[ 0.        ]
  [ 0.06291619]]

 [[ 0.06291619]
  [ 0.12558307]]

 [[ 0.12558307]
  [ 0.18775236]]

 [[ 0.18775236]
  [ 0.24917769]]

 [[ 0.24917769]
  [ 0.3096157 ]]

 [[ 0.3096157 ]
  [ 0.36882689]]

 [[ 0.36882689]
  [ 0.42657665]]

 [[ 0.42657665]
  [ 0.48263615]]

 [[ 0.48263615]
  [ 0.53678328]]

 [[ 0.53678328]
  [ 0.58880346]]

 [[ 0.58880346]
  [ 0.63849059]]

 [[ 0.63849059]
  [ 0.68564779]]

 [[ 0.68564779]
  [ 0.73008819]]

 [[ 0.73008819]
  [ 0.77163571]]

 [[ 0.77163571]
  [ 0.81012572]]

 [[ 0.81012572]
  [ 0.84540571]]

 [[ 0.84540571]
  [ 0.8773359 ]]

 [[ 0.8773359 ]
  [ 0.90578975]]

 [[ 0.90578975]
  [ 0.93065452]]

 [[ 0.93065452]
  [ 0.9518317 ]]

 [[ 0.9518317 ]
  [ 0.96923737]]

 [[ 0.96923737]
  [ 0.98280255]]

 [[ 0.98280255]
  [ 0.99247351]]

 [[ 0.99247351]
  [ 0.99821192]]

 [[ 0.99821192]
  [ 0.99999505]]

 [[ 0.99999505]
  [ 0.99781582]]

 [[ 0.99781582]
  [ 0.99168287]]

 [[ 0.99168287]
  [ 0.98162051]]

 [[ 0.98162051]
  [ 0.9676686 ]]

 [[ 0.9676686 ]
  [ 0.94988243]]

 [[ 0.94988243]
  [ 0.92833248]]

 [[ 0.92833248]
  [ 0.90310412]]

 [[ 0.90310412]
  [ 0.87429733]]

 [[ 0.87429733]
  [ 0.84202625]]

 [[ 0.84202625]
  [ 0.80641875]]

 [[ 0.80641875]
  [ 0.76761592]]

 [[ 0.76761592]
  [ 0.72577151]]

 [[ 0.72577151]
  [ 0.68105132]]

 [[ 0.68105132]
  [ 0.63363256]]

 [[ 0.63363256]
  [ 0.5837031 ]]

 [[ 0.5837031 ]
  [ 0.5314608 ]]
```

```
[[ 0.5314608 ]
 [ 0.47711265]]

[[ 0.47711265]
 [ 0.42087401]]

[[ 0.42087401]
 [ 0.3629677 ]]

[[ 0.3629677 ]
 [ 0.30362319]]

[[ 0.30362319]
 [ 0.2430756 ]]

[[ 0.2430756 ]
 [ 0.18156486]]

[[ 0.18156486]
 [ 0.11933469]]

[[ 0.11933469]
 [ 0.05663168]]

[[ 0.05663168]
 [-0.00629574]]

[[-0.00629574]
 [-0.0691982 ]]

[[-0.0691982 ]
 [-0.13182648]]

[[-0.13182648]
 [-0.19393241]]

[[-0.19393241]
 [-0.25526991]]

[[-0.25526991]
 [-0.31559594]]

[[-0.31559594]
 [-0.37467145]]

[[-0.37467145]
 [-0.43226238]]

[[-0.43226238]
 [-0.48814053]]

[[-0.48814053]
 [-0.54208448]]

[[-0.54208448]
 [-0.59388049]]

[[-0.59388049]
 [-0.64332332]]
```

```
[[-0.64332332]
 [-0.69021707]]

[[-0.69021707]
 [-0.73437593]]

[[-0.73437593]
 [-0.77562491]]

[[-0.77562491]
 [-0.81380058]]

[[-0.81380058]
 [-0.84875167]]

[[-0.84875167]
 [-0.88033969]]

[[-0.88033969]
 [-0.90843947]]

[[-0.90843947]
 [-0.93293968]]

[[-0.93293968]
 [-0.95374324]]

[[-0.95374324]
 [-0.97076771]]

[[-0.97076771]
 [-0.98394564]]

[[-0.98394564]
 [-0.99322482]]

[[-0.99322482]
 [-0.99856846]]

[[-0.99856846]
 [-0.99995541]]

[[-0.99995541]
 [-0.99738016]]

[[-0.99738016]
 [-0.99085292]]

[[-0.99085292]
 [-0.98039956]]

[[-0.98039956]
 [-0.96606148]]

[[-0.96606148]
 [-0.94789551]]

[[-0.94789551]
 [-0.92597363]]

[[-0.92597363]
```

```
 [-0.9003827 ]]

[[-0.9003827 ]
 [-0.87122411]]

[[-0.87122411]
 [-0.83861341]]

[[-0.83861341]
 [-0.80267982]]

[[-0.80267982]
 [-0.76356571]]

[[-0.76356571]
 [-0.72142606]]

[[-0.72142606]
 [-0.67642786]]

[[-0.67642786]
 [-0.6287494 ]]

[[-0.6287494 ]
 [-0.5785796 ]]

[[-0.5785796 ]
 [-0.52611726]]

[[-0.52611726]
 [-0.47157024]]

[[-0.47157024]
 [-0.41515469]]

[[-0.41515469]
 [-0.35709413]]

[[-0.35709413]
 [-0.29761864]]

[[-0.29761864]
 [-0.23696388]]

[[-0.23696388]
 [-0.17537017]]

[[-0.17537017]
 [-0.11308158]]

[[-0.11308158]
 [-0.05034492]]

[[-0.05034492]
 [ 0.01259122]]

[[ 0.01259122]
 [ 0.07547747]]

[[ 0.07547747]
 [ 0.13806466]]
```

```
[[ 0.13806466]
 [ 0.20010478]]

[[ 0.20010478]
 [ 0.26135201]]

[[ 0.26135201]
 [ 0.32156366]]

[[ 0.32156366]
 [ 0.38050117]]

[[ 0.38050117]
 [ 0.43793098]]

[[ 0.43793098]
 [ 0.49362555]]

[[ 0.49362555]
 [ 0.54736419]]

[[ 0.54736419]
 [ 0.59893397]]

[[ 0.59893397]
 [ 0.64813056]]

[[ 0.64813056]
 [ 0.694759  ]]

[[ 0.694759  ]
 [ 0.73863456]]

[[ 0.73863456]
 [ 0.77958337]]

[[ 0.77958337]
 [ 0.81744318]]

[[ 0.81744318]
 [ 0.85206398]]

[[ 0.85206398]
 [ 0.88330858]]

[[ 0.88330858]
 [ 0.91105319]]

[[ 0.91105319]
 [ 0.93518786]]

[[ 0.93518786]
 [ 0.95561698]]

[[ 0.95561698]
 [ 0.97225958]]

[[ 0.97225958]
 [ 0.98504973]]
```

```
[[ 0.98504973]
 [ 0.99393675]]

[[ 0.99393675]
 [ 0.99888542]]

[[ 0.99888542]
 [ 0.99987614]]

[[ 0.99987614]
 [ 0.99690497]]

[[ 0.99690497]
 [ 0.9899837 ]]

[[ 0.9899837 ]
 [ 0.97913974]]

[[ 0.97913974]
 [ 0.96441607]]

[[ 0.96441607]
 [ 0.94587102]]

[[ 0.94587102]
 [ 0.92357809]]

[[ 0.92357809]
 [ 0.89762559]]

[[ 0.89762559]
 [ 0.86811636]]

[[ 0.86811636]
 [ 0.83516734]]

[[ 0.83516734]
 [ 0.79890907]]

[[ 0.79890907]
 [ 0.75948523]]

[[ 0.75948523]
 [ 0.71705202]]

[[ 0.71705202]
 [ 0.67177759]]

[[ 0.67177759]
 [ 0.62384133]]

[[ 0.62384133]
 [ 0.57343317]]

[[ 0.57343317]
 [ 0.52075286]]

[[ 0.52075286]
 [ 0.46600914]]

[[ 0.46600914]
```

```
 [ 0.40941891]]

[[ 0.40941891]
 [ 0.35120641]]

[[ 0.35120641]
 [ 0.2916023 ]]

[[ 0.2916023 ]
 [ 0.23084276]]

[[ 0.23084276]
 [ 0.16916853]]

[[ 0.16916853]
 [ 0.10682399]]

[[ 0.10682399]
 [ 0.04405617]]

[[ 0.04405617]
 [-0.01888621]]

[[-0.01888621]
 [-0.08175375]]

[[-0.08175375]
 [-0.14429736]]

[[-0.14429736]
 [-0.20626921]]

[[-0.20626921]
 [-0.26742375]]

[[-0.26742375]
 [-0.32751865]]

[[-0.32751865]
 [-0.3863158 ]]

[[-0.3863158 ]
 [-0.44358222]]

[[-0.44358222]
 [-0.49909101]]

[[-0.49909101]
 [-0.55262221]]

[[-0.55262221]
 [-0.60396372]]

[[-0.60396372]
 [-0.6529121 ]]

[[-0.6529121 ]
 [-0.6992734 ]]

[[-0.6992734 ]
 [-0.74286391]]
```

```
[[-0.74286391]
 [-0.78351093]]

[[-0.78351093]
 [-0.82105338]]

[[-0.82105338]
 [-0.85534252]]

[[-0.85534252]
 [-0.88624247]]

[[-0.88624247]
 [-0.91363079]]

[[-0.91363079]
 [-0.93739898]]

[[-0.93739898]
 [-0.95745284]]

[[-0.95745284]
 [-0.97371292]]

[[-0.97371292]
 [-0.98611478]]

[[-0.98611478]
 [-0.99460929]]

[[-0.99460929]
 [-0.99916279]]

[[-0.99916279]
 [-0.99975723]]

[[-0.99975723]
 [-0.99639027]]

[[-0.99639027]
 [-0.98907524]]

[[-0.98907524]
 [-0.97784112]]

[[-0.97784112]
 [-0.96273243]]

[[-0.96273243]
 [-0.94380904]]

[[-0.94380904]
 [-0.92114593]]

[[-0.92114593]
 [-0.8948329 ]]

[[-0.8948329 ]
 [-0.8649742 ]]
```

```
[[-0.8649742 ]
 [-0.83168816]]

[[-0.83168816]
 [-0.79510666]]

[[-0.79510666]
 [-0.75537465]]

[[-0.75537465]
 [-0.71264956]]

[[-0.71264956]
 [-0.66710069]]

[[-0.66710069]
 [-0.61890853]]

[[-0.61890853]
 [-0.56826401]]

[[-0.56826401]
 [-0.51536782]]

[[-0.51536782]
 [-0.46042956]]

[[-0.46042956]
 [-0.4036669 ]]

[[-0.4036669 ]
 [-0.34530476]]

[[-0.34530476]
 [-0.2855744 ]]

[[-0.2855744 ]
 [-0.22471249]]

[[-0.22471249]
 [-0.16296018]]

[[-0.16296018]
 [-0.10056216]]

[[-0.10056216]
 [-0.03776568]]

[[-0.03776568]
 [ 0.02518045]]

[[ 0.02518045]
 [ 0.0880268 ]]

[[ 0.0880268 ]
 [ 0.15052435]]

[[ 0.15052435]
 [ 0.21242547]]

[[ 0.21242547]
```

```
 [ 0.27348488]]

[[ 0.27348488]
 [ 0.33346065]]

[[ 0.33346065]
 [ 0.39211512]]

[[ 0.39211512]
 [ 0.44921588]]

[[ 0.44921588]
 [ 0.50453668]]

[[ 0.50453668]
 [ 0.55785832]]

[[ 0.55785832]
 [ 0.60896952]]

[[ 0.60896952]
 [ 0.65766776]]

[[ 0.65766776]
 [ 0.70376007]]

[[ 0.70376007]
 [ 0.74706382]]

[[ 0.74706382]
 [ 0.78740743]]

[[ 0.78740743]
 [ 0.82463104]]

[[ 0.82463104]
 [ 0.85858715]]

[[ 0.85858715]
 [ 0.88914122]]

[[ 0.88914122]
 [ 0.91617219]]

[[ 0.91617219]
 [ 0.93957294]]

[[ 0.93957294]
 [ 0.95925075]]

[[ 0.95925075]
 [ 0.97512765]]

[[ 0.97512765]
 [ 0.98714074]]

[[ 0.98714074]
 [ 0.99524241]]

[[ 0.99524241]
 [ 0.99940055]]
```

```
[[ 0.99940055]
 [ 0.9995987 ]]

[[ 0.9995987 ]
 [ 0.99583607]]

[[ 0.99583607]
 [ 0.98812757]]

[[ 0.98812757]
 [ 0.97650374]]

[[ 0.97650374]
 [ 0.96101064]]

[[ 0.96101064]
 [ 0.94170965]]

[[ 0.94170965]
 [ 0.91867727]]

[[ 0.91867727]
 [ 0.89200474]]

[[ 0.89200474]
 [ 0.86179776]]

[[ 0.86179776]
 [ 0.82817601]]

[[ 0.82817601]
 [ 0.79127273]]

[[ 0.79127273]
 [ 0.75123412]]

[[ 0.75123412]
 [ 0.70821885]]

[[ 0.70821885]
 [ 0.66239735]]

[[ 0.66239735]
 [ 0.61395119]]

[[ 0.61395119]
 [ 0.56307233]]

[[ 0.56307233]
 [ 0.50996236]]

[[ 0.50996236]
 [ 0.45483173]]

[[ 0.45483173]
 [ 0.39789889]]

[[ 0.39789889]
 [ 0.33938943]]
```

```
[[ 0.33938943]
 [ 0.27953518]]

[[ 0.27953518]
 [ 0.21857331]]

[[ 0.21857331]
 [ 0.15674537]]

[[ 0.15674537]
 [ 0.09429635]]

[[ 0.09429635]
 [ 0.03147369]]

[[ 0.03147369]
 [-0.03147369]]

[[-0.03147369]
 [-0.09429635]]

[[-0.09429635]
 [-0.15674537]]

[[-0.15674537]
 [-0.21857331]]

[[-0.21857331]
 [-0.27953518]]

[[-0.27953518]
 [-0.33938943]]

[[-0.33938943]
 [-0.39789889]]

[[-0.39789889]
 [-0.45483173]]

[[-0.45483173]
 [-0.50996236]]

[[-0.50996236]
 [-0.56307233]]

[[-0.56307233]
 [-0.61395119]]

[[-0.61395119]
 [-0.66239735]]

[[-0.66239735]
 [-0.70821885]]

[[-0.70821885]
 [-0.75123412]]

[[-0.75123412]
 [-0.79127273]]

[[-0.79127273]
```

```
 [-0.82817601]]

[[-0.82817601]
 [-0.86179776]]

[[-0.86179776]
 [-0.89200474]]

[[-0.89200474]
 [-0.91867727]]

[[-0.91867727]
 [-0.94170965]]

[[-0.94170965]
 [-0.96101064]]

[[-0.96101064]
 [-0.97650374]]

[[-0.97650374]
 [-0.98812757]]

[[-0.98812757]
 [-0.99583607]]

[[-0.99583607]
 [-0.9995987 ]]

[[-0.9995987 ]
 [-0.99940055]]

[[-0.99940055]
 [-0.99524241]]

[[-0.99524241]
 [-0.98714074]]

[[-0.98714074]
 [-0.97512765]]

[[-0.97512765]
 [-0.95925075]]

[[-0.95925075]
 [-0.93957294]]

[[-0.93957294]
 [-0.91617219]]

[[-0.91617219]
 [-0.88914122]]

[[-0.88914122]
 [-0.85858715]]

[[-0.85858715]
 [-0.82463104]]

[[-0.82463104]
 [-0.78740743]]
```

```
[[-0.78740743]
 [-0.74706382]]

[[-0.74706382]
 [-0.70376007]]

[[-0.70376007]
 [-0.65766776]]

[[-0.65766776]
 [-0.60896952]]

[[-0.60896952]
 [-0.55785832]]

[[-0.55785832]
 [-0.50453668]]

[[-0.50453668]
 [-0.44921588]]

[[-0.44921588]
 [-0.39211512]]

[[-0.39211512]
 [-0.33346065]]

[[-0.33346065]
 [-0.27348488]]

[[-0.27348488]
 [-0.21242547]]

[[-0.21242547]
 [-0.15052435]]

[[-0.15052435]
 [-0.0880268 ]]

[[-0.0880268 ]
 [-0.02518045]]

[[-0.02518045]
 [ 0.03776568]]

[[ 0.03776568]
 [ 0.10056216]]

[[ 0.10056216]
 [ 0.16296018]]

[[ 0.16296018]
 [ 0.22471249]]

[[ 0.22471249]
 [ 0.2855744 ]]

[[ 0.2855744 ]
 [ 0.34530476]]
```

```
[[ 0.34530476]
 [ 0.4036669 ]]

[[ 0.4036669 ]
 [ 0.46042956]]

[[ 0.46042956]
 [ 0.51536782]]

[[ 0.51536782]
 [ 0.56826401]]

[[ 0.56826401]
 [ 0.61890853]]

[[ 0.61890853]
 [ 0.66710069]]

[[ 0.66710069]
 [ 0.71264956]]

[[ 0.71264956]
 [ 0.75537465]]

[[ 0.75537465]
 [ 0.79510666]]

[[ 0.79510666]
 [ 0.83168816]]

[[ 0.83168816]
 [ 0.8649742 ]]

[[ 0.8649742 ]
 [ 0.8948329 ]]

[[ 0.8948329 ]
 [ 0.92114593]]

[[ 0.92114593]
 [ 0.94380904]]

[[ 0.94380904]
 [ 0.96273243]]

[[ 0.96273243]
 [ 0.97784112]]

[[ 0.97784112]
 [ 0.98907524]]

[[ 0.98907524]
 [ 0.99639027]]

[[ 0.99639027]
 [ 0.99975723]]

[[ 0.99975723]
 [ 0.99916279]]

[[ 0.99916279]
```

```
 [ 0.99460929]]

[[ 0.99460929]
 [ 0.98611478]]

[[ 0.98611478]
 [ 0.97371292]]

[[ 0.97371292]
 [ 0.95745284]]

[[ 0.95745284]
 [ 0.93739898]]

[[ 0.93739898]
 [ 0.91363079]]

[[ 0.91363079]
 [ 0.88624247]]

[[ 0.88624247]
 [ 0.85534252]]

[[ 0.85534252]
 [ 0.82105338]]

[[ 0.82105338]
 [ 0.78351093]]

[[ 0.78351093]
 [ 0.74286391]]

[[ 0.74286391]
 [ 0.6992734 ]]

[[ 0.6992734 ]
 [ 0.6529121 ]]

[[ 0.6529121 ]
 [ 0.60396372]]

[[ 0.60396372]
 [ 0.55262221]]

[[ 0.55262221]
 [ 0.49909101]]

[[ 0.49909101]
 [ 0.44358222]]

[[ 0.44358222]
 [ 0.3863158 ]]

[[ 0.3863158 ]
 [ 0.32751865]]

[[ 0.32751865]
 [ 0.26742375]]

[[ 0.26742375]
 [ 0.20626921]]
```

```
[[ 0.20626921]
 [ 0.14429736]]

[[ 0.14429736]
 [ 0.08175375]]

[[ 0.08175375]
 [ 0.01888621]]

[[ 0.01888621]
 [-0.04405617]]

[[-0.04405617]
 [-0.10682399]]

[[-0.10682399]
 [-0.16916853]]

[[-0.16916853]
 [-0.23084276]]

[[-0.23084276]
 [-0.2916023 ]]

[[-0.2916023 ]
 [-0.35120641]]

[[-0.35120641]
 [-0.40941891]]

[[-0.40941891]
 [-0.46600914]]

[[-0.46600914]
 [-0.52075286]]

[[-0.52075286]
 [-0.57343317]]

[[-0.57343317]
 [-0.62384133]]

[[-0.62384133]
 [-0.67177759]]

[[-0.67177759]
 [-0.71705202]]

[[-0.71705202]
 [-0.75948523]]

[[-0.75948523]
 [-0.79890907]]

[[-0.79890907]
 [-0.83516734]]

[[-0.83516734]
 [-0.86811636]]
```

```
[[-0.86811636]
 [-0.89762559]]

[[-0.89762559]
 [-0.92357809]]

[[-0.92357809]
 [-0.94587102]]

[[-0.94587102]
 [-0.96441607]]

[[-0.96441607]
 [-0.97913974]]

[[-0.97913974]
 [-0.9899837 ]]

[[-0.9899837 ]
 [-0.99690497]]

[[-0.99690497]
 [-0.99987614]]

[[-0.99987614]
 [-0.99888542]]

[[-0.99888542]
 [-0.99393675]]

[[-0.99393675]
 [-0.98504973]]

[[-0.98504973]
 [-0.97225958]]

[[-0.97225958]
 [-0.95561698]]

[[-0.95561698]
 [-0.93518786]]

[[-0.93518786]
 [-0.91105319]]

[[-0.91105319]
 [-0.88330858]]

[[-0.88330858]
 [-0.85206398]]

[[-0.85206398]
 [-0.81744318]]

[[-0.81744318]
 [-0.77958337]]

[[-0.77958337]
 [-0.73863456]]

[[-0.73863456]
```

```
 [-0.694759  ]]

[[-0.694759  ]
 [-0.64813056]]

[[-0.64813056]
 [-0.59893397]]

[[-0.59893397]
 [-0.54736419]]

[[-0.54736419]
 [-0.49362555]]

[[-0.49362555]
 [-0.43793098]]

[[-0.43793098]
 [-0.38050117]]

[[-0.38050117]
 [-0.32156366]]

[[-0.32156366]
 [-0.26135201]]

[[-0.26135201]
 [-0.20010478]]

[[-0.20010478]
 [-0.13806466]]

[[-0.13806466]
 [-0.07547747]]

[[-0.07547747]
 [-0.01259122]]

[[-0.01259122]
 [ 0.05034492]]

[[ 0.05034492]
 [ 0.11308158]]

[[ 0.11308158]
 [ 0.17537017]]

[[ 0.17537017]
 [ 0.23696388]]

[[ 0.23696388]
 [ 0.29761864]]

[[ 0.29761864]
 [ 0.35709413]]

[[ 0.35709413]
 [ 0.41515469]]

[[ 0.41515469]
 [ 0.47157024]]
```

```
[[ 0.47157024]
 [ 0.52611726]]

[[ 0.52611726]
 [ 0.5785796 ]]

[[ 0.5785796 ]
 [ 0.6287494 ]]

[[ 0.6287494 ]
 [ 0.67642786]]

[[ 0.67642786]
 [ 0.72142606]]

[[ 0.72142606]
 [ 0.76356571]]

[[ 0.76356571]
 [ 0.80267982]]

[[ 0.80267982]
 [ 0.83861341]]

[[ 0.83861341]
 [ 0.87122411]]

[[ 0.87122411]
 [ 0.9003827 ]]

[[ 0.9003827 ]
 [ 0.92597363]]

[[ 0.92597363]
 [ 0.94789551]]

[[ 0.94789551]
 [ 0.96606148]]

[[ 0.96606148]
 [ 0.98039956]]

[[ 0.98039956]
 [ 0.99085292]]

[[ 0.99085292]
 [ 0.99738016]]

[[ 0.99738016]
 [ 0.99995541]]

[[ 0.99995541]
 [ 0.99856846]]

[[ 0.99856846]
 [ 0.99322482]]

[[ 0.99322482]
 [ 0.98394564]]
```

```
[[ 0.98394564]
 [ 0.97076771]]

[[ 0.97076771]
 [ 0.95374324]]

[[ 0.95374324]
 [ 0.93293968]]

[[ 0.93293968]
 [ 0.90843947]]

[[ 0.90843947]
 [ 0.88033969]]

[[ 0.88033969]
 [ 0.84875167]]

[[ 0.84875167]
 [ 0.81380058]]

[[ 0.81380058]
 [ 0.77562491]]

[[ 0.77562491]
 [ 0.73437593]]

[[ 0.73437593]
 [ 0.69021707]]

[[ 0.69021707]
 [ 0.64332332]]

[[ 0.64332332]
 [ 0.59388049]]

[[ 0.59388049]
 [ 0.54208448]]

[[ 0.54208448]
 [ 0.48814053]]

[[ 0.48814053]
 [ 0.43226238]]

[[ 0.43226238]
 [ 0.37467145]]

[[ 0.37467145]
 [ 0.31559594]]

[[ 0.31559594]
 [ 0.25526991]]

[[ 0.25526991]
 [ 0.19393241]]

[[ 0.19393241]
 [ 0.13182648]]

[[ 0.13182648]
```

```
 [ 0.0691982 ]]

[[ 0.0691982 ]
 [ 0.00629574]]

[[ 0.00629574]
 [-0.05663168]]

[[-0.05663168]
 [-0.11933469]]

[[-0.11933469]
 [-0.18156486]]

[[-0.18156486]
 [-0.2430756 ]]

[[-0.2430756 ]
 [-0.30362319]]

[[-0.30362319]
 [-0.3629677 ]]

[[-0.3629677 ]
 [-0.42087401]]

[[-0.42087401]
 [-0.47711265]]

[[-0.47711265]
 [-0.5314608 ]]

[[-0.5314608 ]
 [-0.5837031 ]]

[[-0.5837031 ]
 [-0.63363256]]

[[-0.63363256]
 [-0.68105132]]

[[-0.68105132]
 [-0.72577151]]

[[-0.72577151]
 [-0.76761592]]

[[-0.76761592]
 [-0.80641875]]

[[-0.80641875]
 [-0.84202625]]

[[-0.84202625]
 [-0.87429733]]

[[-0.87429733]
 [-0.90310412]]

[[-0.90310412]
 [-0.92833248]]
```

```
[[-0.92833248]
 [-0.94988243]]

[[-0.94988243]
 [-0.9676686 ]]

[[-0.9676686 ]
 [-0.98162051]]

[[-0.98162051]
 [-0.99168287]]

[[-0.99168287]
 [-0.99781582]]

[[-0.99781582]
 [-0.99999505]]

[[-0.99999505]
 [-0.99821192]]

[[-0.99821192]
 [-0.99247351]]

[[-0.99247351]
 [-0.98280255]]

[[-0.98280255]
 [-0.96923737]]

[[-0.96923737]
 [-0.9518317 ]]

[[-0.9518317 ]
 [-0.93065452]]

[[-0.93065452]
 [-0.90578975]]

[[-0.90578975]
 [-0.8773359 ]]

[[-0.8773359 ]
 [-0.84540571]]

[[-0.84540571]
 [-0.81012572]]

[[-0.81012572]
 [-0.77163571]]

[[-0.77163571]
 [-0.73008819]]

[[-0.73008819]
 [-0.68564779]]

[[-0.68564779]
 [-0.63849059]]
```

```
[[-0.63849059]
 [-0.58880346]]

[[-0.58880346]
 [-0.53678328]]

[[-0.53678328]
 [-0.48263615]]

[[-0.48263615]
 [-0.42657665]]

[[-0.42657665]
 [-0.36882689]]

[[-0.36882689]
 [-0.3096157 ]]

[[-0.3096157 ]
 [-0.24917769]]

[[-0.24917769]
 [-0.18775236]]

[[-0.18775236]
 [-0.12558307]]

[[-0.12558307]
 [-0.06291619]]]
```

In [4]:

```
print(D)
```

```
[[ 1.25583075e-01]
 [ 1.87752357e-01]
 [ 2.49177694e-01]
 [ 3.09615697e-01]
 [ 3.68826887e-01]
 [ 4.26576649e-01]
 [ 4.82636155e-01]
 [ 5.36783277e-01]
 [ 5.88803464e-01]
 [ 6.38490593e-01]
 [ 6.85647786e-01]
 [ 7.30088187e-01]
 [ 7.71635707e-01]
 [ 8.10125720e-01]
 [ 8.45405714e-01]
 [ 8.77335896e-01]
 [ 9.05789748e-01]
 [ 9.30654524e-01]
 [ 9.51831701e-01]
 [ 9.69237367e-01]
 [ 9.82802554e-01]
 [ 9.92473513e-01]
 [ 9.98211922e-01]
 [ 9.99995045e-01]
 [ 9.97815817e-01]
 [ 9.91682871e-01]
 [ 9.81620509e-01]
 [ 9.67668602e-01]
 [ 9.49882432e-01]
 [ 9.28332476e-01]
 [ 9.03104121e-01]
 [ 8.74297332e-01]
 [ 8.42026252e-01]
 [ 8.06418751e-01]
 [ 7.67615920e-01]
 [ 7.25771508e-01]
 [ 6.81051321e-01]
 [ 6.33632555e-01]
 [ 5.83703102e-01]
 [ 5.31460800e-01]
 [ 4.77112653e-01]
 [ 4.20874008e-01]
 [ 3.62967704e-01]
 [ 3.03623187e-01]
 [ 2.43075602e-01]
 [ 1.81564862e-01]
 [ 1.19334693e-01]
 [ 5.66316770e-02]
 [-6.29573527e-03]
 [-6.91982015e-02]
 [-1.31826479e-01]
 [-1.93932410e-01]
 [-2.55269910e-01]
 [-3.15595935e-01]
 [-3.74671452e-01]
 [-4.32262381e-01]
 [-4.88140526e-01]
 [-5.42084477e-01]
 [-5.93880488e-01]
 [-6.43323324e-01]
 [-6.90217074e-01]
 [-7.34375927e-01]
 [-7.75624910e-01]
 [-8.13800579e-01]
 [-8.48751667e-01]
 [-8.80339686e-01]
 [-9.08439472e-01]
 [-9.32939684e-01]
 [-9.53743241e-01]
 [-9.70767714e-01]
 [-9.83945644e-01]
 [-9.93224816e-01]
 [-9.98568462e-01]
 [-9.99955409e-01]
 [-9.97380161e-01]
 [-9.90852922e-01]
 [-9.80399556e-01]
 [-9.66061482e-01]
 [-9.47895514e-01]
 [-9.25973632e-01]
 [-9.00382698e-01]
 [-8.71224113e-01]
 [-8.38613415e-01]
 [-8.02679818e-01]
 [-7.63565706e-01]
 [-7.21426063e-01]
 [-6.76427862e-01]
 [-6.28749402e-01]
 [-5.78579603e-01]
 [-5.26117258e-01]
 [-4.71570240e-01]
 [-4.15154685e-01]
 [-3.57094133e-01]
 [-2.97618642e-01]
 [-2.36963875e-01]
 [-1.75370170e-01]
 [-1.13081582e-01]
 [-5.03449231e-02]
 [ 1.25912210e-02]
 [ 7.54774740e-02]
 [ 1.38064657e-01]
 [ 2.00104777e-01]
 [ 2.61352007e-01]
 [ 3.21563664e-01]
 [ 3.80501166e-01]
 [ 4.37930980e-01]
 [ 4.93625550e-01]
 [ 5.47364192e-01]
 [ 5.98933973e-01]
 [ 6.48130555e-01]
 [ 6.94759004e-01]
 [ 7.38634559e-01]
 [ 7.79583369e-01]
 [ 8.17443181e-01]
 [ 8.52063978e-01]
 [ 8.83308582e-01]
 [ 9.11053189e-01]
 [ 9.35187865e-01]
 [ 9.55616979e-01]
 [ 9.72259583e-01]
 [ 9.85049734e-01]
 [ 9.93936751e-01]
 [ 9.98885422e-01]
 [ 9.99876138e-01]
 [ 9.96904972e-01]
 [ 9.89983699e-01]
 [ 9.79139743e-01]
 [ 9.64416071e-01]
 [ 9.45871024e-01]
 [ 9.23578085e-01]
 [ 8.97625587e-01]
 [ 8.68116362e-01]
 [ 8.35167337e-01]
 [ 7.98909070e-01]
 [ 7.59485227e-01]
 [ 7.17052023e-01]
 [ 6.71777591e-01]
 [ 6.23841327e-01]
 [ 5.73433172e-01]
 [ 5.20752862e-01]
 [ 4.66009135e-01]
 [ 4.09418907e-01]
 [ 3.51206409e-01]
 [ 2.91602301e-01]
 [ 2.30842756e-01]
 [ 1.69168526e-01]
 [ 1.06823988e-01]
 [ 4.40561738e-02]
 [-1.88862077e-02]
 [-8.17537549e-02]
 [-1.44297363e-01]
 [-2.06269212e-01]
 [-2.67423746e-01]
 [-3.27518647e-01]
 [-3.86315798e-01]
 [-4.43582221e-01]
 [-4.99091008e-01]
 [-5.52622210e-01]
 [-6.03963718e-01]
 [-6.52912097e-01]
 [-6.99273396e-01]
 [-7.42863914e-01]
 [-7.83510928e-01]
 [-8.21053382e-01]
 [-8.55342517e-01]
 [-8.86242467e-01]
 [-9.13630795e-01]
 [-9.37398978e-01]
 [-9.57452839e-01]
 [-9.73712915e-01]
 [-9.86114779e-01]
 [-9.94609290e-01]
 [-9.99162789e-01]
 [-9.99757234e-01]
 [-9.96390270e-01]
 [-9.89075237e-01]
 [-9.77841120e-01]
 [-9.62732434e-01]
 [-9.43809043e-01]
 [-9.21145931e-01]
 [-8.94832897e-01]
 [-8.64974201e-01]
 [-8.31688157e-01]
```

```
       [-7.95106655e-01]
       [-7.55374645e-01]
       [-7.12649561e-01]
       [-6.67100693e-01]
       [-6.18908525e-01]
       [-5.68264012e-01]
       [-5.15367825e-01]
       [-4.60429559e-01]
       [-4.03666901e-01]
       [-3.45304764e-01]
       [-2.85574401e-01]
       [-2.24712487e-01]
       [-1.62960178e-01]
       [-1.00562160e-01]
       [-3.77656782e-02]
       [ 2.51804457e-02]
       [ 8.80267954e-02]
       [ 1.50524350e-01]
       [ 2.12425471e-01]
       [ 2.73484884e-01]
       [ 3.33460648e-01]
       [ 3.92115117e-01]
       [ 4.49215880e-01]
       [ 5.04536683e-01]
       [ 5.57858325e-01]
       [ 6.08969524e-01]
       [ 6.57667759e-01]
       [ 7.03760071e-01]
       [ 7.47063824e-01]
       [ 7.87407432e-01]
       [ 8.24631039e-01]
       [ 8.58587152e-01]
       [ 8.89141224e-01]
       [ 9.16172188e-01]
       [ 9.39572937e-01]
       [ 9.59250748e-01]
       [ 9.75127652e-01]
       [ 9.87140738e-01]
       [ 9.95242406e-01]
       [ 9.99400553e-01]
       [ 9.99598704e-01]
       [ 9.95836074e-01]
       [ 9.88127571e-01]
       [ 9.76503739e-01]
       [ 9.61010637e-01]
       [ 9.41709653e-01]
       [ 9.18677266e-01]
       [ 8.92004738e-01]
       [ 8.61797756e-01]
       [ 8.28176011e-01]
       [ 7.91272725e-01]
       [ 7.51234123e-01]
       [ 7.08218851e-01]
       [ 6.62397354e-01]
       [ 6.13951192e-01]
       [ 5.63072327e-01]
       [ 5.09962361e-01]
       [ 4.54831734e-01]
       [ 3.97898895e-01]
       [ 3.39389432e-01]
       [ 2.79535183e-01]
```

```
 [ 2.18573311e-01]
 [ 1.56745371e-01]
 [ 9.42963467e-02]
 [ 3.14736857e-02]
 [-3.14736857e-02]
 [-9.42963467e-02]
 [-1.56745371e-01]
 [-2.18573311e-01]
 [-2.79535183e-01]
 [-3.39389432e-01]
 [-3.97898895e-01]
 [-4.54831734e-01]
 [-5.09962361e-01]
 [-5.63072327e-01]
 [-6.13951192e-01]
 [-6.62397354e-01]
 [-7.08218851e-01]
 [-7.51234123e-01]
 [-7.91272725e-01]
 [-8.28176011e-01]
 [-8.61797756e-01]
 [-8.92004738e-01]
 [-9.18677266e-01]
 [-9.41709653e-01]
 [-9.61010637e-01]
 [-9.76503739e-01]
 [-9.88127571e-01]
 [-9.95836074e-01]
 [-9.99598704e-01]
 [-9.99400553e-01]
 [-9.95242406e-01]
 [-9.87140738e-01]
 [-9.75127652e-01]
 [-9.59250748e-01]
 [-9.39572937e-01]
 [-9.16172188e-01]
 [-8.89141224e-01]
 [-8.58587152e-01]
 [-8.24631039e-01]
 [-7.87407432e-01]
 [-7.47063824e-01]
 [-7.03760071e-01]
 [-6.57667759e-01]
 [-6.08969524e-01]
 [-5.57858325e-01]
 [-5.04536683e-01]
 [-4.49215880e-01]
 [-3.92115117e-01]
 [-3.33460648e-01]
 [-2.73484884e-01]
 [-2.12425471e-01]
 [-1.50524350e-01]
 [-8.80267954e-02]
 [-2.51804457e-02]
 [ 3.77656782e-02]
 [ 1.00562160e-01]
 [ 1.62960178e-01]
 [ 2.24712487e-01]
 [ 2.85574401e-01]
 [ 3.45304764e-01]
 [ 4.03666901e-01]
 [ 4.60429559e-01]
 [ 5.15367825e-01]
 [ 5.68264012e-01]
 [ 6.18908525e-01]
 [ 6.67100693e-01]
 [ 7.12649561e-01]
 [ 7.55374645e-01]
 [ 7.95106655e-01]
 [ 8.31688157e-01]
 [ 8.64974201e-01]
 [ 8.94832897e-01]
 [ 9.21145931e-01]
 [ 9.43809043e-01]
 [ 9.62732434e-01]
 [ 9.77841120e-01]
 [ 9.89075237e-01]
 [ 9.96390270e-01]
 [ 9.99757234e-01]
 [ 9.99162789e-01]
 [ 9.94609290e-01]
 [ 9.86114779e-01]
 [ 9.73712915e-01]
 [ 9.57452839e-01]
 [ 9.37398978e-01]
 [ 9.13630795e-01]
 [ 8.86242467e-01]
 [ 8.55342517e-01]
 [ 8.21053382e-01]
 [ 7.83510928e-01]
 [ 7.42863914e-01]
 [ 6.99273396e-01]
 [ 6.52912097e-01]
 [ 6.03963718e-01]
 [ 5.52622210e-01]
 [ 4.99091008e-01]
 [ 4.43582221e-01]
 [ 3.86315798e-01]
 [ 3.27518647e-01]
 [ 2.67423746e-01]
 [ 2.06269212e-01]
 [ 1.44297363e-01]
 [ 8.17537549e-02]
 [ 1.88862077e-02]
 [-4.40561738e-02]
 [-1.06823988e-01]
 [-1.69168526e-01]
 [-2.30842756e-01]
 [-2.91602301e-01]
 [-3.51206409e-01]
 [-4.09418907e-01]
 [-4.66009135e-01]
 [-5.20752862e-01]
 [-5.73433172e-01]
 [-6.23841327e-01]
 [-6.71777591e-01]
 [-7.17052023e-01]
 [-7.59485227e-01]
 [-7.98909070e-01]
 [-8.35167337e-01]
 [-8.68116362e-01]
 [-8.97625587e-01]
 [-9.23578085e-01]
 [-9.45871024e-01]
 [-9.64416071e-01]
 [-9.79139743e-01]
 [-9.89983699e-01]
 [-9.96904972e-01]
 [-9.99876138e-01]
 [-9.98885422e-01]
 [-9.93936751e-01]
 [-9.85049734e-01]
 [-9.72259583e-01]
 [-9.55616979e-01]
 [-9.35187865e-01]
 [-9.11053189e-01]
 [-8.83308582e-01]
 [-8.52063978e-01]
 [-8.17443181e-01]
 [-7.79583369e-01]
 [-7.38634559e-01]
 [-6.94759004e-01]
 [-6.48130555e-01]
 [-5.98933973e-01]
 [-5.47364192e-01]
 [-4.93625550e-01]
 [-4.37930980e-01]
 [-3.80501166e-01]
 [-3.21563664e-01]
 [-2.61352007e-01]
 [-2.00104777e-01]
 [-1.38064657e-01]
 [-7.54774740e-02]
 [-1.25912210e-02]
 [ 5.03449231e-02]
 [ 1.13081582e-01]
 [ 1.75370170e-01]
 [ 2.36963875e-01]
 [ 2.97618642e-01]
 [ 3.57094133e-01]
 [ 4.15154685e-01]
 [ 4.71570240e-01]
 [ 5.26117258e-01]
 [ 5.78579603e-01]
 [ 6.28749402e-01]
 [ 6.76427862e-01]
 [ 7.21426063e-01]
 [ 7.63565706e-01]
 [ 8.02679818e-01]
 [ 8.38613415e-01]
 [ 8.71224113e-01]
 [ 9.00382698e-01]
 [ 9.25973632e-01]
 [ 9.47895514e-01]
 [ 9.66061482e-01]
 [ 9.80399556e-01]
 [ 9.90852922e-01]
 [ 9.97380161e-01]
 [ 9.99955409e-01]
 [ 9.98568462e-01]
 [ 9.93224816e-01]
 [ 9.83945644e-01]
 [ 9.70767714e-01]
 [ 9.53743241e-01]
 [ 9.32939684e-01]
 [ 9.08439472e-01]
 [ 8.80339686e-01]
 [ 8.48751667e-01]
 [ 8.13800579e-01]
 [ 7.75624910e-01]
 [ 7.34375927e-01]
 [ 6.90217074e-01]
 [ 6.43323324e-01]
 [ 5.93880488e-01]
 [ 5.42084477e-01]
 [ 4.88140526e-01]
 [ 4.32262381e-01]
 [ 3.74671452e-01]
 [ 3.15595935e-01]
 [ 2.55269910e-01]
 [ 1.93932410e-01]
 [ 1.31826479e-01]
 [ 6.91982015e-02]
 [ 6.29573527e-03]
 [-5.66316770e-02]
 [-1.19334693e-01]
 [-1.81564862e-01]
 [-2.43075602e-01]
 [-3.03623187e-01]
 [-3.62967704e-01]
 [-4.20874008e-01]
 [-4.77112653e-01]
 [-5.31460800e-01]
 [-5.83703102e-01]
 [-6.33632555e-01]
 [-6.81051321e-01]
 [-7.25771508e-01]
 [-7.67615920e-01]
 [-8.06418751e-01]
 [-8.42026252e-01]
 [-8.74297332e-01]
 [-9.03104121e-01]
 [-9.28332476e-01]
 [-9.49882432e-01]
 [-9.67668602e-01]
 [-9.81620509e-01]
 [-9.91682871e-01]
 [-9.97815817e-01]
 [-9.99995045e-01]
 [-9.98211922e-01]
 [-9.92473513e-01]
 [-9.82802554e-01]
 [-9.69237367e-01]
 [-9.51831701e-01]
 [-9.30654524e-01]
 [-9.05789748e-01]
 [-8.77335896e-01]
 [-8.45405714e-01]
 [-8.10125720e-01]
 [-7.71635707e-01]
 [-7.30088187e-01]
 [-6.85647786e-01]
 [-6.38490593e-01]
 [-5.88803464e-01]
 [-5.36783277e-01]
 [-4.82636155e-01]
 [-4.26576649e-01]
 [-3.68826887e-01]
 [-3.09615697e-01]
 [-2.49177694e-01]
 [-1.87752357e-01]
 [-1.25583075e-01]
 [-6.29161861e-02]
 [-1.22464680e-15]]
```

In [5]:

```
# データ設定
N_train = int(len(data) * 0.8)
N_validation = len(data) - N_train
print(N_train,N_validation,len(data))
```

```
398 100 498
```

In [6]:

```
x_train, x_test, d_train, d_test = train_test_split(X, D, test_size=N_validation)
```

In [7]:

```
x_train
```

Out[7]:

```
array([[[ 0.43793098],
        [ 0.49362555]],

       [[-0.13806466],
        [-0.07547747]],

       [[-0.97076771],
        [-0.98394564]],

       [[-0.8773359 ],
        [-0.84540571]],

       [[ 0.36882689],
        [ 0.42657665]],

       [[ 0.6287494 ],
        [ 0.67642786]],

       [[ 0.73863456],
        [ 0.77958337]],

       [[ 0.75123412],
        [ 0.70821885]],

       [[-0.02518045],
        [ 0.03776568]],

       [[-0.32751865],
        [-0.3863158 ]],

       [[-0.98812757],
        [-0.99583607]],

       [[ 0.42657665],
        [ 0.48263615]],

       [[ 0.71705202],
        [ 0.67177759]],

       [[ 0.48263615],
        [ 0.53678328]],

       [[ 0.95374324],
        [ 0.93293968]],

       [[ 0.93293968],
        [ 0.90843947]],

       [[-0.86811636],
        [-0.89762559]],

       [[-0.82463104],
        [-0.78740743]],

       [[ 0.60896952],
        [ 0.65766776]],

       [[-0.66710069],
        [-0.61890853]],
       [[ 0.97076771],
        [ 0.95374324]],

       [[ 0.05663168],
        [-0.00629574]],

       [[ 0.83516734],
        [ 0.79890907]],

       [[-0.75948523],
        [-0.79890907]],

       [[ 0.52611726],
        [ 0.5785796 ]],

       [[ 0.61395119],
        [ 0.56307233]],

       [[-0.54208448],
        [-0.59388049]],

       [[-0.36882689],
        [-0.3096157 ]],

       [[-0.68105132],
        [-0.72577151]],

       [[ 0.5785796 ],
        [ 0.6287494 ]],

       [[-0.84540571],
        [-0.81012572]],

       [[ 0.96101064],
        [ 0.94170965]],

       [[-0.15052435],
        [-0.0880268 ]],

       [[-0.31559594],
        [-0.37467145]],

       [[-0.39211512],
        [-0.33346065]],

       [[-0.63849059],
        [-0.58880346]],

       [[-0.42087401],
        [-0.47711265]],

       [[-0.3629677 ],
        [-0.42087401]],

       [[ 0.98039956],
        [ 0.99085292]],

       [[-0.75537465],
        [-0.71264956]],
       [[-0.34530476],
        [-0.2855744 ]],

       [[-0.13182648],
        [-0.19393241]],

       [[-0.99856846],
        [-0.99995541]],

       [[ 0.99781582],
        [ 0.99168287]],

       [[-0.73008819],
        [-0.68564779]],

       [[ 0.24917769],
        [ 0.3096157 ]],

       [[ 0.92114593],
        [ 0.94380904]],

       [[ 0.95745284],
        [ 0.93739898]],

       [[ 0.27953518],
        [ 0.21857331]],

       [[ 0.3096157 ],
        [ 0.36882689]],

       [[ 0.52075286],
        [ 0.46600914]],

       [[-0.91105319],
        [-0.88330858]],

       [[ 0.79890907],
        [ 0.75948523]],

       [[-0.48263615],
        [-0.42657665]],

       [[ 0.98504973],
        [ 0.99393675]],

       [[ 0.98280255],
        [ 0.99247351]],

       [[-0.90843947],
        [-0.93293968]],

       [[ 0.9518317 ],
        [ 0.96923737]],

       [[-0.89762559],
        [-0.92357809]],

       [[ 0.77163571],
        [ 0.81012572]],

       [[-0.97512765],
        [-0.95925075]],

       [[ 0.9995987 ],
        [ 0.99583607]],

       [[-0.77562491],
        [-0.81380058]],

       [[-0.95561698],
        [-0.93518786]],

       [[ 0.91867727],
        [ 0.89200474]],

       [[-0.92597363],
        [-0.9003827 ]],

       [[-0.2916023 ],
        [-0.35120641]],

       [[ 0.16296018],
        [ 0.22471249]],

       [[-0.12558307],
        [-0.06291619]],

       [[-0.94988243],
        [-0.9676686 ]],

       [[-0.2430756 ],
        [-0.30362319]],

       [[-0.60396372],
        [-0.6529121 ]],

       [[-0.99247351],
        [-0.98280255]],

       [[-0.71705202],
        [-0.75948523]],

       [[-0.3096157 ],
        [-0.24917769]],

       [[ 0.39211512],
        [ 0.44921588]],

       [[ 0.94380904],
        [ 0.96273243]],

       [[-0.0691982 ],
        [-0.13182648]],

       [[ 0.4036669 ],
        [ 0.46042956]],

       [[-0.05663168],
        [-0.11933469]],

       [[ 0.93739898],
        [ 0.91363079]],
       [[-0.78351093],
        [-0.82105338]],

       [[ 0.38050117],
        [ 0.43793098]],

       [[-0.9518317 ],
        [-0.93065452]],

       [[-0.73437593],
        [-0.77562491]],

       [[ 0.44358222],
        [ 0.3863158 ]],

       [[ 0.99856846],
        [ 0.99322482]],

       [[ 0.69021707],
        [ 0.64332332]],

       [[ 0.96273243],
        [ 0.97784112]],

       [[-0.6287494 ],
        [-0.5785796 ]],

       [[ 0.71264956],
        [ 0.75537465]],

       [[-0.63363256],
        [-0.68105132]],

       [[-0.41515469],
        [-0.35709413]],

       [[-0.0880268 ],
        [-0.02518045]],

       [[-0.47711265],
        [-0.5314608 ]],

       [[ 0.51536782],
        [ 0.56826401]],

       [[-0.38050117],
        [-0.32156366]],

       [[-0.47157024],
        [-0.41515469]],

       [[ 0.90578975],
        [ 0.93065452]],

       [[-0.10056216],
        [-0.03776568]],

       [[ 0.2916023 ],
        [ 0.23084276]],
       [[-0.49909101],
        [-0.55262221]],

       [[-0.01888621],
        [-0.08175375]],

       [[-0.98280255],
        [-0.96923737]],

       [[-0.88624247],
        [-0.91363079]],

       [[-0.33938943],
        [-0.39789889]],

       [[ 0.94170965],
        [ 0.91867727]],

       [[ 0.84875167],
        [ 0.81380058]],

       [[-0.84875167],
        [-0.88033969]],

       [[ 0.56826401],
        [ 0.61890853]],

       [[-0.21857331],
        [-0.27953518]],

       [[ 0.57343317],
        [ 0.52075286]],

       [[ 0.07547747],
        [ 0.13806466]],

       [[-0.694759  ],
        [-0.64813056]],

       [[ 0.12558307],
        [ 0.18775236]],

       [[ 0.41515469],
        [ 0.47157024]],

       [[ 0.8773359 ],
        [ 0.90578975]],

       [[ 0.23084276],
        [ 0.16916853]],

       [[ 0.03147369],
        [-0.03147369]],

       [[-0.75123412],
        [-0.79127273]],

       [[ 0.94988243],
        [ 0.92833248]],

       [[ 0.77958337],
        [ 0.81744318]],

       [[ 0.15674537],
        [ 0.09429635]],

       [[ 0.20010478],
        [ 0.26135201]],

       [[ 0.23696388],
        [ 0.29761864]],

       [[ 0.61890853],
        [ 0.66710069]],

       [[-0.99639027],
        [-0.98907524]],

       [[ 0.21857331],
        [ 0.15674537]],

       [[ 0.85206398],
        [ 0.88330858]],

       [[-0.91363079],
        [-0.93739898]],

       [[ 0.76356571],
        [ 0.80267982]],

       [[ 0.95561698],
        [ 0.97225958]],

       [[-0.66239735],
        [-0.70821885]],

       [[ 0.694759  ],
        [ 0.73863456]],

       [[ 0.9003827 ],
        [ 0.92597363]],

       [[ 0.92597363],
        [ 0.94789551]],

       [[-0.88914122],
        [-0.85858715]],

       [[ 0.75948523],
        [ 0.71705202]],

       [[-0.56826401],
        [-0.51536782]],

       [[ 0.26742375],
        [ 0.20626921]],

       [[-0.92114593],
        [-0.8948329 ]],

       [[ 0.99690497],
        [ 0.9899837 ]],
       [[-0.91617219],
        [-0.88914122]],

       [[-0.15674537],
        [-0.21857331]],

       [[ 0.35709413],
        [ 0.41515469]],

       [[-0.97913974],
        [-0.9899837 ]],

       [[-0.16916853],
        [-0.23084276]],

       [[ 0.89200474],
        [ 0.86179776]],

       [[ 0.86811636],
        [ 0.83516734]],

       [[ 0.87429733],
        [ 0.84202625]],

       [[-0.99999505],
        [-0.99821192]],

       [[-0.6992734 ],
        [-0.74286391]],

       [[ 0.84202625],
        [ 0.80641875]],

       [[ 0.3629677 ],
        [ 0.30362319]],

       [[-0.17537017],
        [-0.11308158]],

       [[-0.80267982],
        [-0.76356571]],

       [[-0.64332332],
        [-0.69021707]],

       [[-0.03776568],
        [ 0.02518045]],

       [[-0.91867727],
        [-0.94170965]],

       [[ 0.91617219],
        [ 0.93957294]],

       [[-0.99322482],
        [-0.99856846]],

       [[ 0.80641875],
        [ 0.76761592]],
       [[ 0.22471249],
        [ 0.2855744 ]],

       [[ 0.9676686 ],
        [ 0.94988243]],

       [[-0.53678328],
        [-0.48263615]],

       [[ 0.60396372],
        [ 0.55262221]],

       [[ 0.48814053],
        [ 0.43226238]],

       [[ 0.88914122],
        [ 0.91617219]],

       [[ 0.99168287],
        [ 0.98162051]],

       [[ 0.88624247],
        [ 0.85534252]],

       [[-0.97225958],
        [-0.95561698]],

       [[-0.3863158 ],
        [-0.44358222]],

       [[-0.9676686 ],
        [-0.98162051]],

       [[ 0.3863158 ],
        [ 0.32751865]],

       [[ 0.67177759],
        [ 0.62384133]],

       [[ 0.70376007],
        [ 0.74706382]],

       [[ 0.26135201],
        [ 0.32156366]],

       [[ 0.08175375],
        [ 0.01888621]],

       [[ 0.32751865],
        [ 0.26742375]],

       [[ 0.99085292],
        [ 0.99738016]],

       [[ 0.82463104],
        [ 0.85858715]],

       [[-0.85206398],
        [-0.81744318]],

       [[-0.61890853],
        [-0.56826401]],

       [[ 0.78740743],
        [ 0.82463104]],

       [[ 0.27348488],
        [ 0.33346065]],

       [[ 0.82817601],
        [ 0.79127273]],

       [[-0.56307233],
        [-0.61395119]],

       [[-0.88033969],
        [-0.90843947]],

       [[ 0.99821192],
        [ 0.99999505]],

       [[ 0.85858715],
        [ 0.88914122]],

       [[ 0.99524241],
        [ 0.99940055]],

       [[ 0.96441607],
        [ 0.94587102]],

       [[ 0.64813056],
        [ 0.694759  ]],

       [[-0.85534252],
        [-0.88624247]],

       [[ 0.83861341],
        [ 0.87122411]],

       [[-0.43793098],
        [-0.38050117]],

       [[ 0.5837031 ],
        [ 0.5314608 ]],

       [[-0.9995987 ],
        [-0.99940055]],

       [[-0.35120641],
        [-0.40941891]],

       [[-0.96606148],
        [-0.94789551]],

       [[ 0.75537465],
        [ 0.79510666]],

       [[ 0.83168816],
        [ 0.8649742 ]],

       [[ 0.73008819],
        [ 0.77163571]],
       [[-0.70376007],
        [-0.65766776]],

       [[ 0.47157024],
        [ 0.52611726]],

       [[-0.42657665],
        [-0.36882689]],

       [[ 0.97650374],
        [ 0.96101064]],

       [[-0.6529121 ],
        [-0.6992734 ]],

       [[-0.87122411],
        [-0.83861341]],

       [[-0.24917769],
        [-0.18775236]],

       [[-0.98162051],
        [-0.99168287]],

       [[ 0.59893397],
        [ 0.64813056]],

       [[ 0.04405617],
        [-0.01888621]],

       [[-0.14429736],
        [-0.20626921]],

       [[ 0.84540571],
        [ 0.8773359 ]],

       [[-0.99781582],
        [-0.99999505]],

       [[ 0.54736419],
        [ 0.59893397]],

       [[-0.04405617],
        [-0.10682399]],

       [[ 0.62384133],
        [ 0.57343317]],

       [[ 0.70821885],
        [ 0.66239735]],

       [[-0.61395119],
        [-0.66239735]],

       [[-0.01259122],
        [ 0.05034492]],

       [[ 0.8948329 ],
        [ 0.92114593]],
       [[ 0.67642786],
        [ 0.72142606]],

       [[ 0.09429635],
        [ 0.03147369]],

       [[ 0.72142606],
        [ 0.76356571]],

       [[ 0.93065452],
        [ 0.9518317 ]],

       [[-0.21242547],
        [-0.15052435]],

       [[-0.94170965],
        [-0.96101064]],

       [[ 0.99583607],
        [ 0.98812757]],

       [[-0.88330858],
        [-0.85206398]],

       [[ 0.18775236],
        [ 0.24917769]],

       [[ 0.14429736],
        [ 0.08175375]],

       [[ 0.99987614],
        [ 0.99690497]],

       [[-0.93957294],
        [-0.91617219]],

       [[-0.74706382],
        [-0.70376007]],

       [[-0.99085292],
        [-0.98039956]],

       [[-0.83168816],
        [-0.79510666]],

       [[-0.99987614],
        [-0.99888542]],

       [[-0.99821192],
        [-0.99247351]],

       [[-0.99940055],
        [-0.99524241]],

       [[-0.33346065],
        [-0.27348488]],

       [[ 0.81380058],
        [ 0.77562491]],

       [[-0.8948329 ],
        [-0.8649742 ]],

       [[ 0.25526991],
        [ 0.19393241]],

       [[-0.94789551],
        [-0.92597363]],

       [[ 0.46042956],
        [ 0.51536782]],

       [[-0.50996236],
        [-0.56307233]],

       [[ 0.63849059],
        [ 0.68564779]],

       [[-0.93518786],
        [-0.91105319]],

       [[-0.76356571],
        [-0.72142606]],

       [[-0.86179776],
        [-0.89200474]],

       [[ 0.76761592],
        [ 0.72577151]],

       [[-0.96923737],
        [-0.9518317 ]],

       [[-0.48814053],
        [-0.54208448]],

       [[ 0.5314608 ],
        [ 0.47711265]],

       [[-0.96101064],
        [-0.97650374]],

       [[-0.00629574],
        [-0.0691982 ]],

       [[ 0.55785832],
        [ 0.60896952]],

       [[-0.26135201],
        [-0.20010478]],

       [[-0.32156366],
        [-0.26135201]],

       [[ 0.80267982],
        [ 0.83861341]],

       [[ 0.99393675],
        [ 0.99888542]],

       [[ 0.98714074],
        [ 0.99524241]],
       [[-0.69021707],
        [-0.73437593]],

       [[-0.99888542],
        [-0.99393675]],

       [[ 0.94587102],
        [ 0.92357809]],

       [[-0.71264956],
        [-0.66710069]],

       [[ 0.99247351],
        [ 0.99821192]],

       [[ 0.42087401],
        [ 0.3629677 ]],

       [[ 0.59388049],
        [ 0.54208448]],

       [[ 0.        ],
        [ 0.06291619]],

       [[ 0.0880268 ],
        [ 0.15052435]],

       [[-0.92357809],
        [-0.94587102]],

       [[ 0.99995541],
        [ 0.99856846]],

       [[-0.11308158],
        [-0.05034492]],

       [[ 0.01888621],
        [-0.04405617]],

       [[ 0.2855744 ],
        [ 0.34530476]],

       [[-0.77163571],
        [-0.73008819]],

       [[-0.97784112],
        [-0.96273243]],

       [[-0.94587102],
        [-0.96441607]],

       [[-0.39789889],
        [-0.45483173]],

       [[-0.35709413],
        [-0.29761864]],

       [[-0.76761592],
        [-0.80641875]],
       [[-0.29761864],
        [-0.23696388]],

       [[ 0.54208448],
        [ 0.48814053]],

       [[-0.93739898],
        [-0.95745284]],

       [[-0.93293968],
        [-0.95374324]],

       [[ 0.34530476],
        [ 0.4036669 ]],

       [[ 0.11933469],
        [ 0.05663168]],

       [[ 0.49909101],
        [ 0.44358222]],

       [[-0.99690497],
        [-0.99987614]],

       [[ 0.44921588],
        [ 0.50453668]],

       [[-0.95925075],
        [-0.93957294]],

       [[ 0.98812757],
        [ 0.97650374]],

       [[ 0.10682399],
        [ 0.04405617]],

       [[ 0.40941891],
        [ 0.35120641]],

       [[ 0.82105338],
        [ 0.78351093]],

       [[-0.94380904],
        [-0.92114593]],

       [[-0.40941891],
        [-0.46600914]],

       [[ 0.92357809],
        [ 0.89762559]],

       [[ 0.19393241],
        [ 0.13182648]],

       [[-0.18775236],
        [-0.12558307]],

       [[-0.70821885],
        [-0.75123412]],

       [[ 0.99975723],
        [ 0.99916279]],

       [[ 0.21242547],
        [ 0.27348488]],

       [[-0.83861341],
        [-0.80267982]],

       [[ 0.97225958],
        [ 0.98504973]],

       [[-0.93065452],
        [-0.90578975]],

       [[ 0.88033969],
        [ 0.84875167]],

       [[-0.5314608 ],
        [-0.5837031 ]],

       [[-0.18156486],
        [-0.2430756 ]],

       [[-0.81744318],
        [-0.77958337]],

       [[-0.52611726],
        [-0.47157024]],

       [[-0.9003827 ],
        [-0.87122411]],

       [[-0.89200474],
        [-0.91867727]],

       [[ 0.93518786],
        [ 0.95561698]],

       [[-0.52075286],
        [-0.57343317]],

       [[-0.50453668],
        [-0.44921588]],

       [[-0.43226238],
        [-0.48814053]],

       [[ 0.97784112],
        [ 0.98907524]],

       [[-0.45483173],
        [-0.50996236]],

       [[-0.99393675],
        [-0.98504973]],

       [[ 0.31559594],
        [ 0.25526991]],

       [[ 0.47711265],
        [ 0.42087401]],
       [[ 0.32156366],
        [ 0.38050117]],

       [[-0.64813056],
        [-0.59893397]],

       [[-0.72577151],
        [-0.76761592]],

       [[-0.99583607],
        [-0.9995987 ]],

       [[ 0.43226238],
        [ 0.37467145]],

       [[ 0.68105132],
        [ 0.63363256]],

       [[-0.59893397],
        [-0.54736419]],

       [[-0.19393241],
        [-0.25526991]],

       [[ 0.96606148],
        [ 0.98039956]],

       [[ 0.77562491],
        [ 0.73437593]],

       [[ 0.97913974],
        [ 0.96441607]],

       [[-0.55785832],
        [-0.50453668]],

       [[ 0.98162051],
        [ 0.9676686 ]],

       [[-0.8649742 ],
        [-0.83168816]],

       [[-0.96441607],
        [-0.97913974]],

       [[-0.74286391],
        [-0.78351093]],

       [[-0.08175375],
        [-0.14429736]],

       [[ 0.33938943],
        [ 0.27953518]],

       [[ 0.15052435],
        [ 0.21242547]],

       [[ 0.92833248],
        [ 0.90310412]],
       [[ 0.90310412],
        [ 0.87429733]],

       [[ 0.9899837 ],
        [ 0.97913974]],

       [[-0.79510666],
        [-0.75537465]],

       [[-0.92833248],
        [-0.94988243]],

       [[ 0.00629574],
        [-0.05663168]],

       [[-0.65766776],
        [-0.60896952]],

       [[ 0.99738016],
        [ 0.99995541]],

       [[-0.07547747],
        [-0.01259122]],

       [[-0.30362319],
        [-0.3629677 ]],

       [[ 0.16916853],
        [ 0.10682399]],

       [[-0.78740743],
        [-0.74706382]],

       [[-0.84202625],
        [-0.87429733]],

       [[-0.98907524],
        [-0.97784112]],

       [[-0.05034492],
        [ 0.01259122]],

       [[ 0.6529121 ],
        [ 0.60396372]],

       [[ 0.0691982 ],
        [ 0.00629574]],

       [[ 0.90843947],
        [ 0.88033969]],

       [[ 0.78351093],
        [ 0.74286391]],

       [[-0.16296018],
        [-0.10056216]],

       [[ 0.39789889],
        [ 0.33938943]],

       [[-0.80641875],
        [-0.84202625]],

       [[ 0.81744318],
        [ 0.85206398]],

       [[ 0.29761864],
        [ 0.35709413]],

       [[-0.82817601],
        [-0.86179776]],

       [[-0.98394564],
        [-0.99322482]],

       [[ 0.85534252],
        [ 0.82105338]],

       [[ 0.99999505],
        [ 0.99781582]],

       [[-0.82105338],
        [-0.85534252]],

       [[ 0.6992734 ],
        [ 0.6529121 ]],

       [[-0.90578975],
        [-0.8773359 ]],

       [[-0.99916279],
        [-0.99975723]],

       [[-0.81012572],
        [-0.77163571]],

       [[ 0.63363256],
        [ 0.5837031 ]],

       [[-0.4036669 ],
        [-0.34530476]],

       [[ 0.79510666],
        [ 0.83168816]],

       [[-0.20010478],
        [-0.13806466]],

       [[-0.67642786],
        [-0.6287494 ]],

       [[-0.99168287],
        [-0.99781582]],

       [[-0.95374324],
        [-0.97076771]],

       [[-0.72142606],
        [-0.67642786]],

       [[-0.44921588],
        [-0.39211512]],
       [[ 0.45483173],
        [ 0.39789889]],

       [[-0.98714074],
        [-0.97512765]],

       [[ 0.65766776],
        [ 0.70376007]],

       [[ 0.53678328],
        [ 0.58880346]],

       [[-0.57343317],
        [-0.62384133]],

       [[-0.2855744 ],
        [-0.22471249]],

       [[-0.09429635],
        [-0.15674537]],

       [[ 0.99639027],
        [ 0.99975723]],

       [[-0.46042956],
        [-0.4036669 ]],

       [[ 0.88330858],
        [ 0.91105319]],

       [[ 0.18156486],
        [ 0.11933469]],

       [[-0.98611478],
        [-0.99460929]]])
```

In [8]:

```
input_layer_size = 1#入力層サイズ
hidden_layer_size = 5#隠れ層サイズ
output_layer_size = 1#出力層サイズ

weight_init_std = 0.01
learning_rate = 0.1#学習率

iters_num = 500#反復回数
```

In [9]:

```
# ウェイト初期化（バイアスは簡単のため省略）
W_in = weight_init_std * np.random.randn(input_layer_size, hidden_layer_size)#係数×標準正規分布
W_out = weight_init_std * np.random.randn(hidden_layer_size, output_layer_size)
W = weight_init_std * np.random.randn(hidden_layer_size, hidden_layer_size)
```

In [10]:

```
print(W_in)
```

```
[[-0.00596314 -0.00052567 -0.0193628   0.00188779  0.00523891]]
```

In [11]:

```
# 勾配
W_in_grad = np.zeros_like(W_in)
W_out_grad = np.zeros_like(W_out)
W_grad = np.zeros_like(W)
```

In [12]:

```
print(W_in_grad)
```

```
[[0. 0. 0. 0. 0.]]
```

In [13]:

```
us = []
zs = []

u = np.zeros(hidden_layer_size)
z = np.zeros(hidden_layer_size)
y = np.zeros(output_layer_size)
```

In [14]:

```
print(u,z,y)
```

```
[0. 0. 0. 0. 0.] [0. 0. 0. 0. 0.] [0.]
```

In [15]:

```
delta_out = np.zeros(output_layer_size)
delta = np.zeros(hidden_layer_size)
```

In [16]:

```
print(delta_out,delta)
print(x_train.shape[0])
```

```
[0.] [0. 0. 0. 0. 0.]
398
```

In [17]:

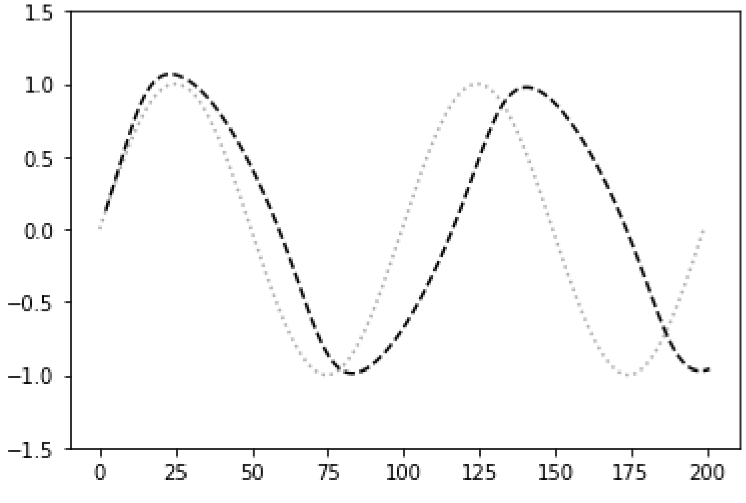
```
losses = []

# トレーニング
for i in range(iters_num):#500
    for s in range(x_train.shape[0]):##398行
        us.clear()
        zs.clear()
        z *= 0

        # sにおける正解データ
        d = d_train[s]

        xs = x_train[s]

        # 時系列ループ
        for t in range(maxlen):#t=0, t=1.

            # 入力値
            x = xs[t]
            u = np.dot(x, W_in) + np.dot(z, W)
            us.append(u)
            z = np.tanh(u)
            zs.append(z)

        y = np.dot(z, W_out)# z1 * W_out

        #誤差
        loss = functions.mean_squared_error(d, y)

        delta_out = functions.d_mean_squared_error(d, y)

        delta *= 0

        for t in range(maxlen)[::-1]:#range(1, -1, -1), t=1,t=0.

            #t=1の場合 delta = 0 + np.dot(delta_out, W_out.T) * d_tanh(us[t])
            #t=0の場合 delta = (np.dot(delta, W.T) + np.dot(delta_out, W_out.T)) * d_tanh(us[t])
            delta = (np.dot(delta, W.T) + np.dot(delta_out, W_out.T)) * d_tanh(us[t])

            # 勾配更新
            W_grad += np.dot(zs[t].reshape(-1,1), delta.reshape(1,-1))
            W_in_grad += np.dot(xs[t], delta.reshape(1,-1))
        W_out_grad = np.dot(z.reshape(-1,1), delta_out)


        # 勾配適用
        W -= learning_rate * W_grad
        W_in -= learning_rate * W_in_grad
        W_out -= learning_rate * W_out_grad.reshape(-1,1)

        #勾配初期化
        W_in_grad *= 0
        W_out_grad *= 0
        W_grad *= 0
```

In [18]:

```
# テスト
for s in range(x_test.shape[0]):
    z *= 0

    # sにおける正解データ
    d = d_test[s]

    xs = x_test[s]

    # 時系列ループ
    for t in range(maxlen):

        # 入力値
        x = xs[t]
        u = np.dot(x, W_in) + np.dot(z, W)
        z = np.tanh(u)

    y = np.dot(z, W_out)

    #誤差
    loss = functions.mean_squared_error(d, y)
    print('loss:', loss, '   d:', d, '   y:', y)



original = np.full(maxlen, None)
pred_num = 200

xs = test_head

# sin波予測
for s in range(0, pred_num):
    z *= 0
    for t in range(maxlen):

        # 入力値
        x = xs[t]
        u = np.dot(x, W_in) + np.dot(z, W)
        z = np.tanh(u)

    y = np.dot(z, W_out)
    original = np.append(original, y)
    xs = np.delete(xs, 0)
    xs = np.append(xs, y)

plt.figure()
plt.ylim([-1.5, 1.5])
plt.plot(np.sin(np.linspace(0, round_num* pred_num / div_num * np.pi, pred_num)), linestyle='dot
ted', color='#aaaaaa')
plt.plot(original, linestyle='dashed', color='black')
plt.show()
```

```
loss: 5.111112056304763e-07    d: [-0.47157024]    y: [-0.47258129]
loss: 1.4153640452728792e-06    d: [-0.39789889]    y: [-0.39958137]
loss: 2.559902991488718e-06    d: [-0.78740743]    y: [-0.78967013]
loss: 7.216263290783507e-09    d: [0.25526991]    y: [0.25514977]
loss: 1.692063852303632e-06    d: [0.6529121]    y: [0.6547517]
loss: 7.238070929589735e-07    d: [0.8773359]    y: [0.87613273]
loss: 6.94623672640184e-07    d: [0.92114593]    y: [0.91996727]
loss: 1.2284734878711618e-06    d: [-0.58880346]    y: [-0.59037093]
loss: 1.249746554038834e-08    d: [-0.6529121]    y: [-0.65307019]
loss: 1.747012227351786e-06    d: [-0.32751865]    y: [-0.32938788]
loss: 2.0214401036935557e-07    d: [0.39789889]    y: [0.39853473]
loss: 1.7248378220182004e-06    d: [-0.15674537]    y: [-0.1586027]
loss: 1.3462666565758106e-06    d: [0.97076771]    y: [0.97240861]
loss: 6.884294927342299e-07    d: [-0.50453668]    y: [-0.50571008]
loss: 8.964485145726196e-09    d: [-0.99460929]    y: [-0.99447539]
loss: 1.6819644091700236e-06    d: [0.95745284]    y: [0.95928694]
loss: 6.636708769141727e-07    d: [-0.92833248]    y: [-0.92718037]
loss: 1.6550737476982935e-06    d: [-0.35120641]    y: [-0.35302579]
loss: 1.4812813327813093e-06    d: [-0.3863158]    y: [-0.38803701]
loss: 1.5504762308512516e-06    d: [0.63363256]    y: [0.63539351]
loss: 1.0859688992783592e-06    d: [0.44921588]    y: [0.45068963]
loss: 7.024336524123257e-07    d: [-0.86811636]    y: [-0.86693109]
loss: 4.298419312275074e-07    d: [0.45483173]    y: [0.45575893]
loss: 1.5800953987150462e-07    d: [0.59893397]    y: [0.59949613]
loss: 1.305987382511987e-07    d: [0.18156486]    y: [0.18105379]
loss: 1.6215636803484033e-06    d: [0.64332332]    y: [0.64512419]
loss: 2.6565935995811714e-08    d: [0.99247351]    y: [0.99224301]
loss: 7.407414549659186e-07    d: [-0.89762559]    y: [-0.89640842]
loss: 3.7108253202912373e-07    d: [-0.11308158]    y: [-0.11222009]
loss: 2.5763917247784403e-06    d: [0.79127273]    y: [0.7935427]
loss: 1.656899449754742e-06    d: [-0.64813056]    y: [-0.64995094]
loss: 4.637650286011886e-07    d: [0.95561698]    y: [0.9546539]
loss: 3.6393779690636794e-07    d: [0.99690497]    y: [0.99775813]
loss: 2.2160406235611887e-07    d: [-0.4036669]    y: [-0.40433264]
loss: 5.740145938550149e-08    d: [-0.99987614]    y: [-1.00021496]
loss: 2.762903013775288e-07    d: [-0.57343317]    y: [-0.57417653]
loss: 1.855495428968903e-07    d: [0.75537465]    y: [0.75476547]
loss: 9.152860658438427e-07    d: [0.00629574]    y: [0.00494275]
loss: 1.8797284550728012e-06    d: [-0.23084276]    y: [-0.23278169]
loss: 1.0524302613338606e-06    d: [0.56307233]    y: [0.56452314]
loss: 4.631459840874837e-08    d: [0.99975723]    y: [1.00006158]
loss: 3.447189422141964e-07    d: [0.11933469]    y: [0.11850437]
loss: 1.883099480769221e-06    d: [0.23696388]    y: [0.23890455]
loss: 5.1214730714751955e-09    d: [0.99524241]    y: [0.9951412]
loss: 1.051230447773728e-06    d: [-0.98039956]    y: [-0.98184954]
loss: 1.813882872732977e-06    d: [0.18775236]    y: [0.18965703]
loss: 5.27036239975623e-07    d: [0.82463104]    y: [0.82360436]
loss: 5.169924703573374e-07    d: [-0.94988243]    y: [-0.94886558]
loss: 6.747741177793689e-07    d: [0.92597363]    y: [0.92481193]
loss: 1.8809353374298544e-06    d: [0.26135201]    y: [0.26329156]
loss: 2.0325852791579403e-07    d: [-0.75948523]    y: [-0.75884764]
loss: 9.92544090722427e-08    d: [-0.98611478]    y: [-0.98566924]
loss: 3.54176004989751e-07    d: [-0.43793098]    y: [-0.43877262]
loss: 3.137779621326153e-08    d: [0.23084276]    y: [0.23059225]
loss: 2.6937112046164813e-06    d: [0.83516734]    y: [0.83748842]
loss: 2.588227073819169e-07    d: [0.77163571]    y: [0.77091623]
loss: 1.2613599482853231e-06    d: [0.97371292]    y: [0.97530122]
loss: 1.4907373517710847e-07    d: [-0.99975723]    y: [-1.00030326]
loss: 4.232044587656063e-07    d: [0.99583607]    y: [0.99675608]
loss: 3.787447800805401e-07    d: [0.44358222]    y: [0.44445256]
loss: 1.8816335271670955e-06    d: [-0.94789551]    y: [-0.94983543]
```

```
loss: 1.2190012629922352e-06    d: [-0.97512765]    y: [-0.97668906]
loss: 1.996710038100712e-06    d: [-0.694759]    y: [-0.69675736]
loss: 1.8849134192596554e-06    d: [-0.2430756]    y: [-0.24501721]
loss: 1.8748123442560268e-06    d: [0.22471249]    y: [0.22664888]
loss: 4.2620357543512357e-07    d: [-0.10056216]    y: [-0.0996389]
loss: 4.325800120889982e-09    d: [0.68564779]    y: [0.68555477]
loss: 1.8306176723615103e-06    d: [0.29761864]    y: [0.29953208]
loss: 2.332044298284346e-09    d: [-0.99583607]    y: [-0.99576778]
loss: 6.585761031379055e-07    d: [-0.99085292]    y: [-0.99200059]
loss: 7.651581590766538e-08    d: [0.35120641]    y: [0.3515976]
loss: 7.290859282135365e-07    d: [-0.88033969]    y: [-0.87913214]
loss: 1.6685814037161896e-09    d: [-0.68105132]    y: [-0.68099355]
loss: 2.691838713149061e-06    d: [0.85534252]    y: [0.85766279]
loss: 7.327100876992387e-07    d: [-0.98907524]    y: [-0.99028578]
loss: 5.4908929482703205e-08    d: [-0.71705202]    y: [-0.71672063]
loss: 6.837326417292688e-07    d: [-0.86179776]    y: [-0.86062837]
loss: 1.6566551435475068e-06    d: [0.13806466]    y: [0.13988491]
loss: 5.681550927763364e-07    d: [-0.48263615]    y: [-0.48370213]
loss: 2.2589028031750398e-07    d: [-0.15052435]    y: [-0.1498522]
loss: 1.7452123204601994e-06    d: [0.16296018]    y: [0.16482845]
loss: 2.09155034432836e-06    d: [0.70821885]    y: [0.71026411]
loss: 1.7631822152185102e-06    d: [0.95374324]    y: [0.9556211]
loss: 2.4114224765198005e-07    d: [0.97512765]    y: [0.97443319]
loss: 2.067490890143368e-06    d: [0.93739898]    y: [0.93943244]
loss: 8.485394806789291e-07    d: [0.98611478]    y: [0.9874175]
loss: 1.72986325908338e-07    d: [0.98039956]    y: [0.97981136]
loss: 3.9139136578767244e-07    d: [-0.55262221]    y: [-0.55350696]
loss: 5.144245939051554e-07    d: [0.08175375]    y: [0.08073943]
loss: 1.5434925122990647e-06    d: [-0.37467145]    y: [-0.37642843]
loss: 8.64601116962286e-08    d: [0.98714074]    y: [0.9867249]
loss: 1.7821870821634795e-06    d: [0.17537017]    y: [0.17725812]
loss: 1.7227716039716527e-06    d: [-0.95561698]    y: [-0.9574732]
loss: 1.7032584357931716e-06    d: [0.15052435]    y: [0.15237003]
loss: 2.2961399277292755e-06    d: [-0.92114593]    y: [-0.92328889]
loss: 1.4861572178323642e-07    d: [-0.38050117]    y: [-0.38104635]
loss: 8.149646046184465e-07    d: [-0.48814053]    y: [-0.48941721]
loss: 9.154701847834834e-07    d: [0.54208448]    y: [0.5434376]
loss: 8.19187220025201e-09    d: [-0.69021707]    y: [-0.69008907]
loss: 1.1946819048400479e-07    d: [0.60896952]    y: [0.60945833]
```



[try]

- iters_numを100にしよう

In [19]:

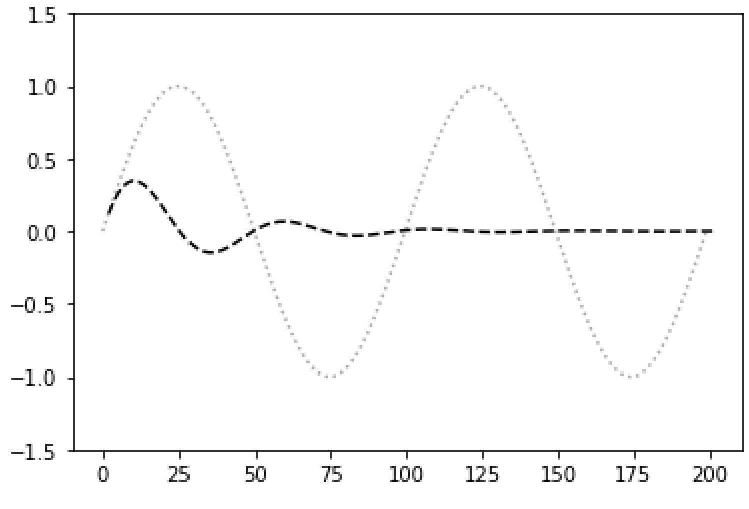
```
import sys, os
sys.path.append(os.pardir)  # 親ディレクトリのファイルをインポートするための設定
import numpy as np
from common import functions
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split

np.random.seed(0)

# sin曲線
round_num = 10
div_num = 500
ts = np.linspace(0, round_num * np.pi, div_num)
f = np.sin(ts)

def d_tanh(x):
    return 1/(np.cosh(x)**2 + 1e-4)

# ひとつの時系列データの長さ
maxlen = 2

# sin波予測の入力データ
test_head = [[f[k]] for k in range(0, maxlen)]

data = []
target = []

for i in range(div_num - maxlen):
    data.append(f[i: i + maxlen])
    target.append(f[i + maxlen])

X = np.array(data).reshape(len(data), maxlen, 1)
D = np.array(target).reshape(len(data), 1)

# データ設定
N_train = int(len(data) * 0.8)
N_validation = len(data) - N_train

x_train, x_test, d_train, d_test = train_test_split(X, D, test_size=N_validation)

input_layer_size = 1
hidden_layer_size = 5
output_layer_size = 1

weight_init_std = 0.01
learning_rate = 0.1

iters_num = 100

# ウェイト初期化 (バイアスは簡単のため省略)
W_in = weight_init_std * np.random.randn(input_layer_size, hidden_layer_size)
W_out = weight_init_std * np.random.randn(hidden_layer_size, output_layer_size)
W = weight_init_std * np.random.randn(hidden_layer_size, hidden_layer_size)

# 勾配
W_in_grad = np.zeros_like(W_in)
W_out_grad = np.zeros_like(W_out)
W_grad = np.zeros_like(W)
```

```
us = []
zs = []

u = np.zeros(hidden_layer_size)
z = np.zeros(hidden_layer_size)
y = np.zeros(output_layer_size)

delta_out = np.zeros(output_layer_size)
delta = np.zeros(hidden_layer_size)

losses = []

# トレーニング
for i in range(iters_num):
    for s in range(x_train.shape[0]):
        us.clear()
        zs.clear()
        z *= 0

        # sにおける正解データ
        d = d_train[s]

        xs = x_train[s]

        # 時系列ループ
        for t in range(maxlen):

            # 入力値
            x = xs[t]
            u = np.dot(x, W_in) + np.dot(z, W)
            us.append(u)
            z = np.tanh(u)
            zs.append(z)

        y = np.dot(z, W_out)

        #誤差
        loss = functions.mean_squared_error(d, y)

        delta_out = functions.d_mean_squared_error(d, y)

        delta *= 0
        for t in range(maxlen)[::-1]:

            delta = (np.dot(delta, W.T) + np.dot(delta_out, W_out.T)) * d_tanh(us[t])

            # 勾配更新
            W_grad += np.dot(zs[t].reshape(-1,1), delta.reshape(1,-1))
            W_in_grad += np.dot(xs[t], delta.reshape(1,-1))
        W_out_grad = np.dot(z.reshape(-1,1), delta_out)

        # 勾配適用
        W -= learning_rate * W_grad
        W_in -= learning_rate * W_in_grad
        W_out -= learning_rate * W_out_grad.reshape(-1,1)

        W_in_grad *= 0
        W_out_grad *= 0
        W_grad *= 0

# テスト
```

```
for s in range(x_test.shape[0]):
    z *= 0

    # sにおける正解データ
    d = d_test[s]

    xs = x_test[s]

    # 時系列ループ
    for t in range(maxlen):

        # 入力値
        x = xs[t]
        u = np.dot(x, W_in) + np.dot(z, W)
        z = np.tanh(u)

    y = np.dot(z, W_out)

    #誤差
    loss = functions.mean_squared_error(d, y)
    print('loss:', loss, '    d:', d, '    y:', y)



original = np.full(maxlen, None)
pred_num = 200

xs = test_head

# sin波予測
for s in range(0, pred_num):
    z *= 0
    for t in range(maxlen):

        # 入力値
        x = xs[t]
        u = np.dot(x, W_in) + np.dot(z, W)
        z = np.tanh(u)

    y = np.dot(z, W_out)
    original = np.append(original, y)
    xs = np.delete(xs, 0)
    xs = np.append(xs, y)

plt.figure()
plt.ylim([-1.5, 1.5])
plt.plot(np.sin(np.linspace(0, round_num* pred_num / div_num * np.pi, pred_num)), linestyle='dotted', color='#aaaaaa')
plt.plot(original, linestyle='dashed', color='black')
plt.show()
```

```
loss: 1.0018211688363152e-06    d: [-0.47157024]    y: [-0.47298574]
loss: 3.831628581739869e-05    d: [-0.39789889]    y: [-0.38914489]
loss: 1.1955502192448862e-05    d: [-0.78740743]    y: [-0.79229732]
loss: 5.688169957032234e-07    d: [0.25526991]    y: [0.25633651]
loss: 5.042766131766361e-06    d: [0.6529121]    y: [0.65608787]
loss: 2.323535484703895e-05    d: [0.8773359]    y: [0.87051895]
loss: 1.2851597420607724e-05    d: [0.92114593]    y: [0.9160761]
loss: 2.9577805540623868e-06    d: [-0.58880346]    y: [-0.59123566]
loss: 5.3033087017894245e-05    d: [-0.6529121]    y: [-0.64261325]
loss: 3.068291439526736e-05    d: [-0.32751865]    y: [-0.31968501]
loss: 5.638683232237903e-07    d: [0.39789889]    y: [0.39896084]
loss: 1.4135523848939483e-05    d: [-0.15674537]    y: [-0.15142832]
loss: 1.3233060812018978e-05    d: [0.97076771]    y: [0.97591224]
loss: 1.3588146407522809e-06    d: [-0.50453668]    y: [-0.50618521]
loss: 1.352410323795142e-07    d: [-0.99460929]    y: [-0.99512937]
loss: 1.4962796333390856e-05    d: [0.95745284]    y: [0.96292327]
loss: 1.1148085202439717e-05    d: [-0.92833248]    y: [-0.92361059]
loss: 3.3266626628950615e-05    d: [-0.35120641]    y: [-0.34304962]
loss: 3.707863108292281e-05    d: [-0.3863158]    y: [-0.37770434]
loss: 4.326570868918968e-06    d: [0.63363256]    y: [0.63657418]
loss: 4.3539234693184196e-05    d: [0.44921588]    y: [0.4398843]
loss: 2.533672093849227e-05    d: [-0.86811636]    y: [-0.86099783]
loss: 8.639496897928521e-07    d: [0.45483173]    y: [0.45614623]
loss: 5.311920595825067e-05    d: [0.59893397]    y: [0.58862677]
loss: 1.0055949688712286e-06    d: [0.18156486]    y: [0.18298303]
loss: 4.676740710063253e-06    d: [0.64332332]    y: [0.64638167]
loss: 1.59930658073789e-08    d: [0.99247351]    y: [0.99265236]
loss: 1.8474236177344764e-05    d: [-0.89762559]    y: [-0.89154706]
loss: 1.8802261802955873e-06    d: [-0.11308158]    y: [-0.11502077]
loss: 1.2182950786145393e-05    d: [0.79127273]    y: [0.79620891]
loss: 4.857808115858404e-06    d: [-0.64813056]    y: [-0.65124754]
loss: 5.0259896375981215e-06    d: [0.95561698]    y: [0.95244649]
loss: 5.84751441954738e-06    d: [0.99690497]    y: [1.00032477]
loss: 5.842918948781338e-07    d: [-0.4036669]    y: [-0.40474791]
loss: 2.0465335767793966e-06    d: [-0.99987614]    y: [-1.00189927]
loss: 5.237713037590953e-05    d: [-0.57343317]    y: [-0.56319822]
loss: 4.550293795779566e-05    d: [0.75537465]    y: [0.74583495]
loss: 4.716398434628641e-06    d: [0.00629574]    y: [0.00936702]
loss: 2.065453201950312e-05    d: [-0.23084276]    y: [-0.22441554]
loss: 2.3468307818580476e-06    d: [0.56307233]    y: [0.56523881]
loss: 1.8459664559315096e-06    d: [0.99975723]    y: [1.00167867]
loss: 1.7754700800083484e-06    d: [0.11933469]    y: [0.12121909]
loss: 2.1247195911347073e-05    d: [0.23696388]    y: [0.2304451]
loss: 2.0004302281892026e-07    d: [0.99524241]    y: [0.99587493]
loss: 1.1411501423016343e-05    d: [-0.98039956]    y: [-0.9851769]
loss: 1.6706877736517124e-05    d: [0.18775236]    y: [0.18197189]
loss: 3.446312694027539e-05    d: [0.82463104]    y: [0.81632886]
loss: 6.2409520302279e-06    d: [-0.94988243]    y: [-0.94634946]
loss: 1.1705129757380487e-05    d: [0.92597363]    y: [0.92113522]
loss: 2.3677213141816805e-05    d: [0.26135201]    y: [0.25447055]
loss: 4.49811295488685e-05    d: [-0.75948523]    y: [-0.75000038]
loss: 2.028533646768841e-07    d: [-0.98611478]    y: [-0.98547783]
loss: 7.499350947699529e-07    d: [-0.43793098]    y: [-0.43915567]
loss: 6.702616914592727e-07    d: [0.23084276]    y: [0.23200057]
loss: 1.4664769817573572e-05    d: [0.83516734]    y: [0.84058301]
loss: 4.333798205022689e-05    d: [0.77163571]    y: [0.76232571]
loss: 1.2737717225792436e-05    d: [0.97371292]    y: [0.97876023]
loss: 3.4112375431131424e-06    d: [-0.99975723]    y: [-1.00236922]
loss: 6.429288289821193e-06    d: [0.99583607]    y: [0.99942196]
loss: 7.85436168297347e-07    d: [0.44358222]    y: [0.44483557]
loss: 1.5805387662329316e-05    d: [-0.94789551]    y: [-0.95351786]
```

```
loss: 1.2481961197274553e-05    d: [-0.97512765]    y: [-0.98012404]
loss: 6.866202285129303e-06    d: [-0.694759]    y: [-0.69846473]
loss: 2.1846135041325457e-05    d: [-0.2430756]    y: [-0.23646559]
loss: 2.00684638533191e-05    d: [0.22471249]    y: [0.21837712]
loss: 2.1074553733212728e-06    d: [-0.10056216]    y: [-0.10261519]
loss: 5.174173943527327e-05    d: [0.68564779]    y: [0.6754751]
loss: 2.74601418124138e-05    d: [0.29761864]    y: [0.29020782]
loss: 2.7697525031755335e-07    d: [-0.99583607]    y: [-0.99658035]
loss: 8.510577037380914e-06    d: [-0.99085292]    y: [-0.99497859]
loss: 4.6342324398754784e-07    d: [0.35120641]    y: [0.35216914]
loss: 2.2541089852561257e-05    d: [-0.88033969]    y: [-0.87362536]
loss: 5.198366295972673e-05    d: [-0.68105132]    y: [-0.67085488]
loss: 1.5650869216024103e-05    d: [0.85534252]    y: [0.86093731]
loss: 9.106005186775855e-06    d: [-0.98907524]    y: [-0.99334279]
loss: 4.9537135434721655e-05    d: [-0.71705202]    y: [-0.70709842]
loss: 2.674810502869443e-05    d: [-0.86179776]    y: [-0.85448365]
loss: 1.2704010990953187e-05    d: [0.13806466]    y: [0.13302402]
loss: 1.108184027478091e-06    d: [-0.48263615]    y: [-0.4841249]
loss: 1.3322624649906846e-06    d: [-0.15052435]    y: [-0.15215669]
loss: 1.4631672720098082e-05    d: [0.16296018]    y: [0.15755062]
loss: 7.5277718530109865e-06    d: [0.70821885]    y: [0.712099]
loss: 1.532365497732746e-05    d: [0.95374324]    y: [0.95927924]
loss: 1.4698452123511701e-06    d: [0.97512765]    y: [0.9734131]
loss: 1.6440458209345824e-05    d: [0.93739898]    y: [0.94313317]
loss: 9.989092604746525e-06    d: [0.98611478]    y: [0.99058448]
loss: 7.597360561243469e-07    d: [0.98039956]    y: [0.97916689]
loss: 5.144766400804239e-05    d: [-0.55262221]    y: [-0.54247848]
loss: 2.49562656838687e-06    d: [0.08175375]    y: [0.08398787]
loss: 3.582176538836955e-05    d: [-0.37467145]    y: [-0.3662072]
loss: 1.3465686259774947e-07    d: [0.98714074]    y: [0.98662178]
loss: 1.5651516544468314e-05    d: [0.17537017]    y: [0.16977526]
loss: 1.5147090412232574e-05    d: [-0.95561698]    y: [-0.96112099]
loss: 1.3648771002149447e-05    d: [0.15052435]    y: [0.14529965]
loss: 1.695825274903518e-05    d: [-0.92114593]    y: [-0.92696972]
loss: 5.135842537369791e-07    d: [-0.38050117]    y: [-0.38151466]
loss: 4.704410899615774e-05    d: [-0.48814053]    y: [-0.47844062]
loss: 1.933106793590669e-06    d: [0.54208448]    y: [0.54405075]
loss: 5.148111233523464e-05    d: [-0.69021707]    y: [-0.68007004]
loss: 5.3281078139314694e-05    d: [0.60896952]    y: [0.59864663]
```



[try]

- iters_numを3000(※時間がかかる)にしよう

In [20]:

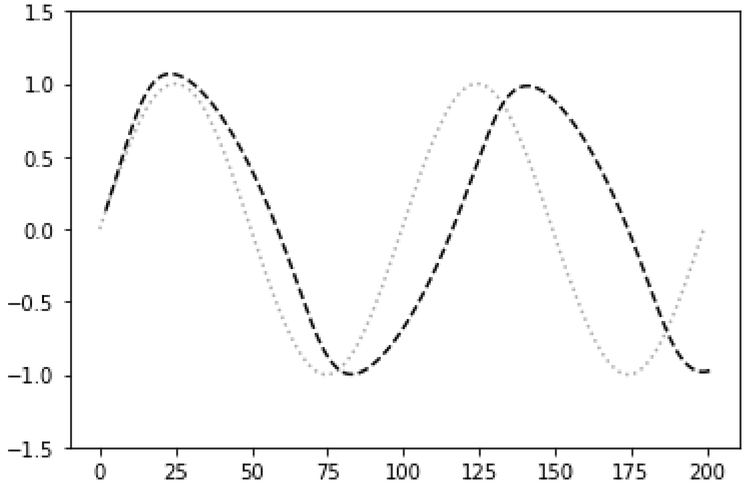
```
import sys, os
sys.path.append(os.pardir)  # 親ディレクトリのファイルをインポートするための設定
import numpy as np
from common import functions
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split

np.random.seed(0)

# sin曲線
round_num = 10
div_num = 500
ts = np.linspace(0, round_num * np.pi, div_num)
f = np.sin(ts)

def d_tanh(x):
    return 1/(np.cosh(x)**2 + 1e-4)

# ひとつの時系列データの長さ
maxlen = 2

# sin波予測の入力データ
test_head = [[f[k]] for k in range(0, maxlen)]

data = []
target = []

for i in range(div_num - maxlen):
    data.append(f[i: i + maxlen])
    target.append(f[i + maxlen])

X = np.array(data).reshape(len(data), maxlen, 1)
D = np.array(target).reshape(len(data), 1)

# データ設定
N_train = int(len(data) * 0.8)
N_validation = len(data) - N_train

x_train, x_test, d_train, d_test = train_test_split(X, D, test_size=N_validation)

input_layer_size = 1
hidden_layer_size = 5
output_layer_size = 1

weight_init_std = 0.01
learning_rate = 0.1

iters_num = 3000

# ウェイト初期化 (バイアスは簡単のため省略)
W_in = weight_init_std * np.random.randn(input_layer_size, hidden_layer_size)
W_out = weight_init_std * np.random.randn(hidden_layer_size, output_layer_size)
W = weight_init_std * np.random.randn(hidden_layer_size, hidden_layer_size)

# 勾配
W_in_grad = np.zeros_like(W_in)
W_out_grad = np.zeros_like(W_out)
W_grad = np.zeros_like(W)
```

```
us = []
zs = []

u = np.zeros(hidden_layer_size)
z = np.zeros(hidden_layer_size)
y = np.zeros(output_layer_size)

delta_out = np.zeros(output_layer_size)
delta = np.zeros(hidden_layer_size)

losses = []

# トレーニング
for i in range(iters_num):
    for s in range(x_train.shape[0]):
        us.clear()
        zs.clear()
        z *= 0

        # sにおける正解データ
        d = d_train[s]

        xs = x_train[s]

        # 時系列ループ
        for t in range(maxlen):

            # 入力値
            x = xs[t]
            u = np.dot(x, W_in) + np.dot(z, W)
            us.append(u)
            z = np.tanh(u)
            zs.append(z)

        y = np.dot(z, W_out)

        #誤差
        loss = functions.mean_squared_error(d, y)

        delta_out = functions.d_mean_squared_error(d, y)

        delta *= 0
        for t in range(maxlen)[::-1]:

            delta = (np.dot(delta, W.T) + np.dot(delta_out, W_out.T)) * d_tanh(us[t])

            # 勾配更新
            W_grad += np.dot(zs[t].reshape(-1,1), delta.reshape(1,-1))
            W_in_grad += np.dot(xs[t], delta.reshape(1,-1))
        W_out_grad = np.dot(z.reshape(-1,1), delta_out)

        # 勾配適用
        W -= learning_rate * W_grad
        W_in -= learning_rate * W_in_grad
        W_out -= learning_rate * W_out_grad.reshape(-1,1)

        W_in_grad *= 0
        W_out_grad *= 0
        W_grad *= 0

# テスト
```

```
for s in range(x_test.shape[0]):
    z *= 0

    # sにおける正解データ
    d = d_test[s]

    xs = x_test[s]

    # 時系列ループ
    for t in range(maxlen):

        # 入力値
        x = xs[t]
        u = np.dot(x, W_in) + np.dot(z, W)
        z = np.tanh(u)

    y = np.dot(z, W_out)

    #誤差
    loss = functions.mean_squared_error(d, y)
    print('loss:', loss, '   d:', d, '   y:', y)



original = np.full(maxlen, None)
pred_num = 200

xs = test_head

# sin波予測
for s in range(0, pred_num):
    z *= 0
    for t in range(maxlen):

        # 入力値
        x = xs[t]
        u = np.dot(x, W_in) + np.dot(z, W)
        z = np.tanh(u)

    y = np.dot(z, W_out)
    original = np.append(original, y)
    xs = np.delete(xs, 0)
    xs = np.append(xs, y)

plt.figure()
plt.ylim([-1.5, 1.5])
plt.plot(np.sin(np.linspace(0, round_num* pred_num / div_num * np.pi, pred_num)), linestyle='dot
ted', color='#aaaaaa')
plt.plot(original, linestyle='dashed', color='black')
plt.show()
```

```
loss: 3.912343122231309e-07     d: [-0.47157024]     y: [-0.47245481]
loss: 1.3052648322202188e-06     d: [-0.39789889]     y: [-0.39951461]
loss: 2.4568661324794487e-06     d: [-0.78740743]     y: [-0.78962413]
loss: 1.836933553802922e-08     d: [0.25526991]     y: [0.25507824]
loss: 1.5234519740797024e-06     d: [0.6529121]     y: [0.65465764]
loss: 5.187974276348698e-07     d: [0.8773359]     y: [0.87631727]
loss: 5.147787696349783e-07     d: [0.92114593]     y: [0.92013126]
loss: 1.0581260161824844e-06     d: [-0.58880346]     y: [-0.5902582]
loss: 3.273298436623086e-08     d: [-0.6529121]     y: [-0.65316796]
loss: 1.577987966024491e-06     d: [-0.32751865]     y: [-0.32929515]
loss: 1.336466170586074e-07     d: [0.39789889]     y: [0.3984159]
loss: 1.536853774722165e-06     d: [-0.15674537]     y: [-0.15849857]
loss: 1.3150157772974342e-06     d: [0.97076771]     y: [0.97238945]
loss: 5.488011547346638e-07     d: [-0.50453668]     y: [-0.50558435]
loss: 3.4729876467868902e-09     d: [-0.99460929]     y: [-0.99452595]
loss: 1.6448993535149984e-06     d: [0.95745284]     y: [0.95926662]
loss: 4.937438724118824e-07     d: [-0.92833248]     y: [-0.92733875]
loss: 1.5034914183015375e-06     d: [-0.35120641]     y: [-0.35294047]
loss: 1.3602120481600566e-06     d: [-0.3863158]     y: [-0.38796517]
loss: 1.3788941646447499e-06     d: [0.63363256]     y: [0.63529321]
loss: 1.026676639415935e-06     d: [0.44921588]     y: [0.45064883]
loss: 4.989099370011925e-07     d: [-0.86811636]     y: [-0.86711745]
loss: 3.2105982617360104e-07     d: [0.45483173]     y: [0.45563306]
loss: 1.9274300037523416e-07     d: [0.59893397]     y: [0.59955485]
loss: 1.4944014901163935e-07     d: [0.18156486]     y: [0.18101816]
loss: 1.45121458712902e-06     d: [0.64332332]     y: [0.64502698]
loss: 1.4921537554827992e-08     d: [0.99247351]     y: [0.99230076]
loss: 5.402630094375798e-07     d: [-0.89762559]     y: [-0.8965861]
loss: 3.707778123070018e-07     d: [-0.11308158]     y: [-0.11222045]
loss: 2.475796598999519e-06     d: [0.79127273]     y: [0.79349794]
loss: 1.4873575430756097e-06     d: [-0.64813056]     y: [-0.64985529]
loss: 3.474647728818484e-07     d: [0.95561698]     y: [0.95478335]
loss: 3.6223161724552973e-07     d: [0.99690497]     y: [0.99775613]
loss: 1.489766397919691e-07     d: [-0.4036669]     y: [-0.40421275]
loss: 6.468374433663414e-08     d: [-0.99987614]     y: [-1.00023581]
loss: 3.0702144887264835e-07     d: [-0.57343317]     y: [-0.57421678]
loss: 1.000399708517381e-07     d: [0.75537465]     y: [0.75492734]
loss: 8.444804874432303e-07     d: [0.00629574]     y: [0.00499613]
loss: 1.675728118498778e-06     d: [-0.23084276]     y: [-0.23267345]
loss: 8.878245718019855e-07     d: [0.56307233]     y: [0.56440486]
loss: 5.348947467931103e-08     d: [0.99975723]     y: [1.00008431]
loss: 3.4716720303622646e-07     d: [0.11933469]     y: [0.11850143]
loss: 1.6794114068719324e-06     d: [0.23696388]     y: [0.23879658]
loss: 1.4042006221287043e-09     d: [0.99524241]     y: [0.99518941]
loss: 1.0268531315089448e-06     d: [-0.98039956]     y: [-0.98183263]
loss: 1.6149340072836495e-06     d: [0.18775236]     y: [0.18954954]
loss: 3.5352983082116036e-07     d: [0.82463104]     y: [0.82379017]
loss: 3.873534373161674e-07     d: [-0.94988243]     y: [-0.94900226]
loss: 5.013989623342182e-07     d: [0.92597363]     y: [0.92497223]
loss: 1.6811027696115039e-06     d: [0.26135201]     y: [0.26318564]
loss: 1.121968297083886e-07     d: [-0.75948523]     y: [-0.75901153]
loss: 6.853245118558047e-08     d: [-0.98611478]     y: [-0.98574456]
loss: 2.57089195761961e-07     d: [-0.43793098]     y: [-0.43864804]
loss: 4.827092212115843e-08     d: [0.23084276]     y: [0.23053204]
loss: 2.6191637340832623e-06     d: [0.83516734]     y: [0.83745608]
loss: 1.5126301136728962e-07     d: [0.77163571]     y: [0.77108568]
loss: 1.2319066819462866e-06     d: [0.97371292]     y: [0.97528257]
loss: 1.5486378230634107e-07     d: [-0.99975723]     y: [-1.00031377]
loss: 4.192727163127846e-07     d: [0.99583607]     y: [0.99675179]
loss: 2.7770361244151876e-07     d: [0.44358222]     y: [0.44432748]
loss: 1.841913853315806e-06     d: [-0.94789551]     y: [-0.94981484]
```

```
loss: 1.1904984010087743e-06     d: [-0.97512765]     y: [-0.9766707]
loss: 1.8413639121995479e-06     d: [-0.694759]     y: [-0.69667805]
loss: 1.6817990829983107e-06     d: [-0.2430756]     y: [-0.24490961]
loss: 1.670761128353524e-06     d: [0.22471249]     y: [0.22654047]
loss: 4.198304467446638e-07     d: [-0.10056216]     y: [-0.09964583]
loss: 3.7564423210744303e-10     d: [0.68564779]     y: [0.6856752]
loss: 1.6441452134331962e-06     d: [0.29761864]     y: [0.29943201]
loss: 2.5076515871196765e-10     d: [-0.99583607]     y: [-0.99581368]
loss: 6.461401578331407e-07     d: [-0.99085292]     y: [-0.99198971]
loss: 4.021287785915039e-08     d: [0.35120641]     y: [0.35149]
loss: 5.240456662428166e-07     d: [-0.88033969]     y: [-0.87931592]
loss: 1.7735508691111668e-09     d: [-0.68105132]     y: [-0.68111088]
loss: 2.6272940036240856e-06     d: [0.85534252]     y: [0.85763481]
loss: 7.177811412733664e-07     d: [-0.98907524]     y: [-0.99027339]
loss: 1.8205365057779595e-08     d: [-0.71705202]     y: [-0.71686121]
loss: 4.823656298617088e-07     d: [-0.86179776]     y: [-0.86081555]
loss: 1.4778770546303914e-06     d: [0.13806466]     y: [0.13978389]
loss: 4.412968820138988e-07     d: [-0.48263615]     y: [-0.48357562]
loss: 2.3912968440796433e-07     d: [-0.15052435]     y: [-0.14983279]
loss: 1.5545796530955029e-06     d: [0.16296018]     y: [0.16472346]
loss: 1.9422431386775e-06     d: [0.70821885]     y: [0.71018976]
loss: 1.7249794746621768e-06     d: [0.95374324]     y: [0.95560065]
loss: 1.7742172036673962e-07     d: [0.97512765]     y: [0.97453197]
loss: 2.0256007449752334e-06     d: [0.93739898]     y: [0.93941174]
loss: 8.299315679013527e-07     d: [0.98611478]     y: [0.98740314]
loss: 1.2494659733137965e-07     d: [0.98039956]     y: [0.97989966]
loss: 4.142129095805074e-07     d: [-0.55262221]     y: [-0.55353239]
loss: 4.976219112443149e-07     d: [0.08175375]     y: [0.08075614]
loss: 1.4117892688671288e-06     d: [-0.37467145]     y: [-0.3763518]
loss: 5.8851642607912266e-08     d: [0.98714074]     y: [0.98679766]
loss: 1.5869287610358092e-06     d: [0.17537017]     y: [0.1771517]
loss: 1.6851238184123744e-06     d: [-0.95561698]     y: [-0.9574528]
loss: 1.5181379496026437e-06     d: [0.15052435]     y: [0.15226684]
loss: 2.251389961868721e-06     d: [-0.92114593]     y: [-0.92326791]
loss: 9.244665177336762e-08     d: [-0.38050117]     y: [-0.38093116]
loss: 7.925032084200643e-07     d: [-0.48814053]     y: [-0.4893995]
loss: 7.580591186093183e-07     d: [0.54208448]     y: [0.54331578]
loss: 1.0255350134194105e-11     d: [-0.69021707]     y: [-0.69021254]
loss: 1.5392344980496395e-07     d: [0.60896952]     y: [0.60952436]
```



[try]

- maxlenを5にしよう

In [ ]:

```
import sys, os
sys.path.append(os.pardir)  # 親ディレクトリのファイルをインポートするための設定
import numpy as np
from common import functions
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split

np.random.seed(0)

# sin曲線
round_num = 10
div_num = 500
ts = np.linspace(0, round_num * np.pi, div_num)
f = np.sin(ts)

def d_tanh(x):
    return 1/(np.cosh(x)**2 + 1e-4)

# ひとつの時系列データの長さ
maxlen = 5

# sin波予測の入力データ
test_head = [[f[k]] for k in range(0, maxlen)]

data = []
target = []

for i in range(div_num - maxlen):
    data.append(f[i: i + maxlen])
    target.append(f[i + maxlen])

X = np.array(data).reshape(len(data), maxlen, 1)
D = np.array(target).reshape(len(data), 1)

# データ設定
N_train = int(len(data) * 0.8)
N_validation = len(data) - N_train

x_train, x_test, d_train, d_test = train_test_split(X, D, test_size=N_validation)

input_layer_size = 1
hidden_layer_size = 5
output_layer_size = 1

weight_init_std = 0.01
learning_rate = 0.1

iters_num = 500

# ウェイト初期化 (バイアスは簡単のため省略)
W_in = weight_init_std * np.random.randn(input_layer_size, hidden_layer_size)
W_out = weight_init_std * np.random.randn(hidden_layer_size, output_layer_size)
W = weight_init_std * np.random.randn(hidden_layer_size, hidden_layer_size)

# 勾配
W_in_grad = np.zeros_like(W_in)
W_out_grad = np.zeros_like(W_out)
W_grad = np.zeros_like(W)
```

```
us = []
zs = []

u = np.zeros(hidden_layer_size)
z = np.zeros(hidden_layer_size)
y = np.zeros(output_layer_size)

delta_out = np.zeros(output_layer_size)
delta = np.zeros(hidden_layer_size)

losses = []

# トレーニング
for i in range(iters_num):
    for s in range(x_train.shape[0]):
        us.clear()
        zs.clear()
        z *= 0

        # sにおける正解データ
        d = d_train[s]

        xs = x_train[s]

        # 時系列ループ
        for t in range(maxlen):

            # 入力値
            x = xs[t]
            u = np.dot(x, W_in) + np.dot(z, W)
            us.append(u)
            z = np.tanh(u)
            zs.append(z)

        y = np.dot(z, W_out)

        #誤差
        loss = functions.mean_squared_error(d, y)

        delta_out = functions.d_mean_squared_error(d, y)

        delta *= 0
        for t in range(maxlen)[::-1]:

            delta = (np.dot(delta, W.T) + np.dot(delta_out, W_out.T)) * d_tanh(us[t])

            # 勾配更新
            W_grad += np.dot(zs[t].reshape(-1,1), delta.reshape(1,-1))
            W_in_grad += np.dot(xs[t], delta.reshape(1,-1))
        W_out_grad = np.dot(z.reshape(-1,1), delta_out)

        # 勾配適用
        W -= learning_rate * W_grad
        W_in -= learning_rate * W_in_grad
        W_out -= learning_rate * W_out_grad.reshape(-1,1)

        W_in_grad *= 0
        W_out_grad *= 0
        W_grad *= 0

# テスト
```

```
for s in range(x_test.shape[0]):
    z *= 0

    # sにおける正解データ
    d = d_test[s]

    xs = x_test[s]

    # 時系列ループ
    for t in range(maxlen):

        # 入力値
        x = xs[t]
        u = np.dot(x, W_in) + np.dot(z, W)
        z = np.tanh(u)

    y = np.dot(z, W_out)

    #誤差
    loss = functions.mean_squared_error(d, y)
    print('loss:', loss, '    d:', d, '    y:', y)



original = np.full(maxlen, None)
pred_num = 200

xs = test_head

# sin波予測
for s in range(0, pred_num):
    z *= 0
    for t in range(maxlen):

        # 入力値
        x = xs[t]
        u = np.dot(x, W_in) + np.dot(z, W)
        z = np.tanh(u)

    y = np.dot(z, W_out)
    original = np.append(original, y)
    xs = np.delete(xs, 0)
    xs = np.append(xs, y)

plt.figure()
plt.ylim([-1.5, 1.5])
plt.plot(np.sin(np.linspace(0, round_num* pred_num / div_num * np.pi, pred_num)), linestyle='dotted', color='#aaaaaa')
plt.plot(original, linestyle='dashed', color='black')
plt.show()
```

In [ ]:

```
# predict sin

------------------------------------------------------------------------------------------------
---------------------------------------------------------------------------------------------
### [try]
-  maxlenを5, iters_numを500, 3000(※時間がかかる)にしよう
```

In [ ]:

```
import sys, os
sys.path.append(os.pardir)  # 親ディレクトリのファイルをインポートするための設定
import numpy as np
from common import functions
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split

np.random.seed(0)

# sin曲線
round_num = 10
div_num = 500
ts = np.linspace(0, round_num * np.pi, div_num)
f = np.sin(ts)

def d_tanh(x):
    return 1/(np.cosh(x)**2 + 1e-4)

# ひとつの時系列データの長さ
maxlen = 5

# sin波予測の入力データ
test_head = [[f[k]] for k in range(0, maxlen)]

data = []
target = []

for i in range(div_num - maxlen):
    data.append(f[i: i + maxlen])
    target.append(f[i + maxlen])

X = np.array(data).reshape(len(data), maxlen, 1)
D = np.array(target).reshape(len(data), 1)

# データ設定
N_train = int(len(data) * 0.8)
N_validation = len(data) - N_train

x_train, x_test, d_train, d_test = train_test_split(X, D, test_size=N_validation)

input_layer_size = 1
hidden_layer_size = 5
output_layer_size = 1

weight_init_std = 0.01
learning_rate = 0.1

iters_num = 3000

# ウェイト初期化 (バイアスは簡単のため省略)
W_in = weight_init_std * np.random.randn(input_layer_size, hidden_layer_size)
W_out = weight_init_std * np.random.randn(hidden_layer_size, output_layer_size)
W = weight_init_std * np.random.randn(hidden_layer_size, hidden_layer_size)

# 勾配
W_in_grad = np.zeros_like(W_in)
W_out_grad = np.zeros_like(W_out)
W_grad = np.zeros_like(W)
```

```
us = []
zs = []

u = np.zeros(hidden_layer_size)
z = np.zeros(hidden_layer_size)
y = np.zeros(output_layer_size)

delta_out = np.zeros(output_layer_size)
delta = np.zeros(hidden_layer_size)

losses = []

# トレーニング
for i in range(iters_num):
    for s in range(x_train.shape[0]):
        us.clear()
        zs.clear()
        z *= 0

        # sにおける正解データ
        d = d_train[s]

        xs = x_train[s]

        # 時系列ループ
        for t in range(maxlen):

            # 入力値
            x = xs[t]
            u = np.dot(x, W_in) + np.dot(z, W)
            us.append(u)
            z = np.tanh(u)
            zs.append(z)

        y = np.dot(z, W_out)

        #誤差
        loss = functions.mean_squared_error(d, y)

        delta_out = functions.d_mean_squared_error(d, y)

        delta *= 0
        for t in range(maxlen)[::-1]:

            delta = (np.dot(delta, W.T) + np.dot(delta_out, W_out.T)) * d_tanh(us[t])

            # 勾配更新
            W_grad += np.dot(zs[t].reshape(-1,1), delta.reshape(1,-1))
            W_in_grad += np.dot(xs[t], delta.reshape(1,-1))
        W_out_grad = np.dot(z.reshape(-1,1), delta_out)

        # 勾配適用
        W -= learning_rate * W_grad
        W_in -= learning_rate * W_in_grad
        W_out -= learning_rate * W_out_grad.reshape(-1,1)

        W_in_grad *= 0
        W_out_grad *= 0
        W_grad *= 0
```

```
# テスト
for s in range(x_test.shape[0]):
    z *= 0

    # sにおける正解データ
    d = d_test[s]

    xs = x_test[s]

    # 時系列ループ
    for t in range(maxlen):

        # 入力値
        x = xs[t]
        u = np.dot(x, W_in) + np.dot(z, W)
        z = np.tanh(u)

    y = np.dot(z, W_out)

    #誤差
    loss = functions.mean_squared_error(d, y)
    print('loss:', loss, '    d:', d, '    y:', y)



original = np.full(maxlen, None)
pred_num = 200

xs = test_head

# sin波予測
for s in range(0, pred_num):
    z *= 0
    for t in range(maxlen):

        # 入力値
        x = xs[t]
        u = np.dot(x, W_in) + np.dot(z, W)
        z = np.tanh(u)

    y = np.dot(z, W_out)
    original = np.append(original, y)
    xs = np.delete(xs, 0)
    xs = np.append(xs, y)

plt.figure()
plt.ylim([-1.5, 1.5])
plt.plot(np.sin(np.linspace(0, round_num* pred_num / div_num * np.pi, pred_num)), linestyle='dotted', color='#aaaaaa')
plt.plot(original, linestyle='dashed', color='black')
plt.show()
```