Approximating Polynomials Using CMA-ES

Defining the Original Polynomial and the Loss

[1]:
```
import numpy as np
import cma
import re
from IPython.display import display, Markdown
```

[2]:
```
COEFF_RANGE = [-25, 25]  # Range of the coefficients for the polynomial
POLY_ORDER = 10          # Order of the polynomial
POLY_SEED = 123          # Seed for numpy.random when generating the coefficients
LOSS_SEED = 321          # Seed for numpy.random when generating the evaluation points␣
 ↪in the loss
NUM_LOSS_EVAL = 1000     # Number of function evaluations in the loss
```

[3]:
```
np.random.seed(POLY_SEED)
coeff = np.random.randint(COEFF_RANGE[0], COEFF_RANGE[1], POLY_ORDER)

def polynomial(x):
    return sum([c * x ** i for i, c in enumerate(coeff)])

def poly_latex(coeff):
    latex_string = ' + '.join([f'{c}x^{i}' for i, c in list(enumerate(coeff))[::-1]])
    latex_string = latex_string.replace('+ -', '- ').replace('x^0', '').replace('x^1',␣
 ↪'x')
    latex_string = re.sub('[+-]\s0(x\^\d+)?', '', latex_string).strip()
    latex_string = re.sub('\s1x', ' x', latex_string).strip()
    return latex_string
```

[4]:
```
display(Markdown(rf'**Original polynomial:** $\quad f(x)={poly_latex(coeff)}$'))
```

Original polynomial: $\quad f(x)=8x^9 - 3x^8 + 17x^7 - 6x^6 - 8x^5 + 13x^4 + 9x^3 + 3x^2 - 23x + 20$

[5]:
```
np.random.seed(LOSS_SEED)
points = np.random.rand(NUM_LOSS_EVAL) / (np.random.rand() + 1e-5)
y_true = np.array([polynomial(x) for x in points])

def loss(coeff):
    """ RMSE loss for given coefficients and true coefficients. """
    y_pred = np.array([sum([c * x ** i for i, c in enumerate(coeff)]) for x in points])
    return np.sqrt(np.mean((y_pred - y_true) ** 2))
```

Find the Best Coefficients Using CMA-ES

[6]:
```
es = cma.CMAEvolutionStrategy(POLY_ORDER * [0], 0.5, {'verbose': -3})
es.optimize(loss)
```

```
Iterat #Fevals   function value  axis ratio  sigma  min&max std  t[m:s]
    1     10 9.888194945854134e+01 1.0e+00 4.94e-01  5e-01  5e-01 0:00.1
    2     20 9.183691142900661e+01 1.2e+00 5.15e-01  5e-01  5e-01 0:00.2
    3     30 8.515738659341694e+01 1.3e+00 5.68e-01  5e-01  6e-01 0:00.3
```

```
   40    400 4.836309220390717e+00 6.2e+00 2.33e+00  1e+00  3e+00 0:03.4
   89    890 1.047068845199528e+00 3.9e+01 1.04e+00  6e-01  2e+00 0:07.4
  100   1000 9.094891067928869e-01 4.9e+01 6.66e-01  3e-01  1e+00 0:08.3
  173   1730 3.125422597515681e-01 2.7e+02 4.69e-01  1e-01  7e-01 0:14.4
  200   2000 9.308110328177320e-02 8.3e+02 6.46e-01  1e-01  2e+00 0:16.6
  295   2950 1.334761601438233e-02 4.8e+03 1.53e-01  1e-02  6e-01 0:24.7
  300   3000 1.320316396821978e-02 4.7e+03 1.31e-01  1e-02  5e-01 0:25.1
  400   4000 3.777950984929764e-03 2.5e+04 2.29e-01  1e-02  1e+00 0:33.6
  500   5000 2.532395621114231e-03 6.3e+04 8.34e-02  1e-03  5e-01 0:42.1
  600   6000 2.355180444679707e-04 4.2e+05 3.64e-03  2e-05  5e-02 0:50.5
  700   7000 1.355192972512089e-04 1.6e+00 1.75e-01  2e-01  2e-01 0:59.0
  800   8000 4.312072320980080e-08 5.2e+00 1.51e-04  5e-05  1e-04 1:07.6
  900   9000 2.351297888496353e-10 6.4e+00 1.33e-06  3e-07  5e-07 1:16.3
 1000  10000 2.909297487988929e-10 1.2e+01 8.50e-07  1e-07  3e-07 1:25.0
Iterat #Fevals   function value  axis ratio  sigma  min&max std  t[m:s]
 1100  11000 2.438311312620849e-10 2.3e+01 5.79e-07  5e-08  2e-07 1:33.6
 1200  12000 2.620593227442595e-10 4.7e+01 1.84e-07  2e-08  6e-08 1:42.2
 1300  13000 2.616208858483796e-10 5.8e+01 1.60e-07  1e-08  4e-08 1:50.8
 1400  14000 2.620043155261143e-10 1.3e+02 3.29e-07  2e-08  9e-08 1:59.5
 1425  14250 2.640857105877048e-10 1.2e+02 2.65e-07  2e-08  7e-08 2:01.6
```

[6]: <cma.evolution_strategy.CMAEvolutionStrategy at 0x7f062f78f670>

[7]:
```python
print(f'Best loss value: {loss(es.result.xbest):.2}')
```

```
Best loss value: 2e-10
```

Compare the Learned Polynomial With the Original One

[8]:
```python
display(Markdown(rf'**Original polynomial:** $\newline \quad␣
↪f(x)={poly_latex(coeff)}$'))
display(Markdown(rf'**Learned polynomial:** $\newline\quad f(x)={poly_latex(es.result.
↪xbest)}$'))
```

Original polynomial:
$f(x) = 8x^9 - 3x^8 + 17x^7 - 6x^6 - 8x^5 + 13x^4 + 9x^3 + 3x^2 - 23x + 20$

Learned polynomial:
$f(x) = 8.000000353444193x^9 - 3.000002246146323x^8 + 17.000005979905836x^7 - 6.000008706294466x^6 - 7.999992380326148x^5 + 12.99999583459794x^4 + 9.000001424479706x^3 + 2.9999997140585037x^2 - 22.99999997334089x + 19.99999999959079$