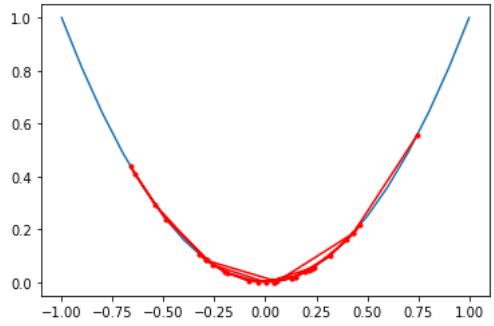
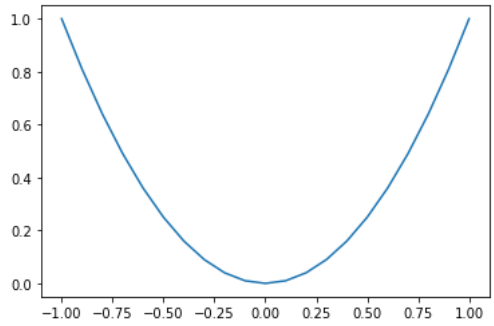
```
from numpy import asarray
from numpy import arange
from numpy.random import rand
from numpy.random import seed
from matplotlib import pyplot

def objective(x):
  return x**2.0

def derivative(x):
  return x * 2.0

def gradient_descent(objective, derivative, bounds, n_iter, step_size, momentum):
  solutions, scores = list(), list()
  solution = bounds[:, 0] + rand(len(bounds)) * (bounds[:, 1] - bounds[:, 0])
  change = 0.0
  for i in range(n_iter):
    gradient = derivative(solution)
    new_change = step_size * gradient + momentum * change
    solution = solution - new_change
    change = new_change
    solution_eval = objective(solution)
    solutions.append(solution)
    scores.append(solution_eval)
    print('%d f(%s) = %.5f' % (i, solution, solution_eval))
  return [solutions, scores]

seed(4)
bounds = asarray([[-1.0, 1.0]])
n_iter = 30
step_size = 0.1
momentum = 0.9
solutions, scores = gradient_descent(objective, derivative, bounds, n_iter, step_size, momentum)
inputs = arange(bounds[0,0], bounds[0,1]+0.1, 0.1)
results = objective(inputs)
pyplot.plot(inputs, results)
pyplot.plot(solutions, scores, '.-', color='red')
pyplot.show()
```

```
0 f([0.74724774]) = 0.55838
1 f([0.42966745]) = 0.18461
2 f([0.0579117]) = 0.00335
3 f([-0.28825082]) = 0.08309
4 f([-0.54214692]) = 0.29392
5 f([-0.66222403]) = 0.43854
6 f([-0.63784862]) = 0.40685
7 f([-0.48834103]) = 0.23848
8 f([-0.25611599]) = 0.06560
9 f([0.00410974]) = 0.00002
10 f([0.23749095]) = 0.05640
11 f([0.40003585]) = 0.16003
12 f([0.46631909]) = 0.21745
13 f([0.43271019]) = 0.18724
14 f([0.31592014]) = 0.09981
15 f([0.14762506]) = 0.02179
16 f([-0.03336551]) = 0.00111
17 f([-0.18958393]) = 0.03594
18 f([-0.29226372]) = 0.08542
19 f([-0.32622279]) = 0.10642
20 f([-0.29154139]) = 0.08500
21 f([-0.20201985]) = 0.04081
22 f([-0.0810465]) = 0.00657
23 f([0.04403882]) = 0.00194
24 f([0.14780784]) = 0.02185
25 f([0.21163839]) = 0.04479
26 f([0.22675821]) = 0.05142
27 f([0.19501441]) = 0.03803
28 f([0.1274421]) = 0.01624
29 f([0.0411386]) = 0.00169
```



```
pyplot.plot(inputs, results)
```

```
[<matplotlib.lines.Line2D at 0x7fa9a80e6730>]
```




```
pyplot.plot(solutions, scores, '.-', color='red')
```

```
[<matplotlib.lines.Line2D at 0x7fa9a80d67c0>]
```