Graphical user interface, text, application

Description automatically generated

Code-

import numpy as np

import matplotlib.pyplot as plt

from tqdm import tqdm

from estimators import SampleAverageEstimator, WeightedEstimator

from testbed import K\_armed\_testbed

np.random.seed(250)

def plot\_performance(estimator\_names, rewards, action\_optimality):

for i, estimator\_name in enumerate(estimator\_names):

average\_run\_rewards = np.average(rewards[i], axis=0)

plt.plot(average\_run\_rewards, label=estimator\_name)

plt.legend()

plt.xlabel("Steps")

plt.ylabel("Average reward")

plt.show()

for i, estimator\_name in enumerate(estimator\_names):

average\_run\_optimality = np.average(action\_optimality[i], axis=0)

plt.plot(average\_run\_optimality, label=estimator\_name)

plt.legend()

plt.xlabel("Steps")

plt.ylabel("% Optimal action")

plt.show()

if \_\_name\_\_ == "\_\_main\_\_":

K = 10

N\_STEPS = 10000

N\_RUNS = 2000

N\_ESTIMATORS = 2

rewards = np.full((N\_ESTIMATORS, N\_RUNS, N\_STEPS), fill\_value=0.)

optimal\_selections = np.full((N\_ESTIMATORS, N\_RUNS, N\_STEPS), fill\_value=0.)

for run\_i in tqdm(range(N\_RUNS)):

testbed = K\_armed\_testbed(k\_actions=K)

action\_value\_estimates = np.full(K, fill\_value=0.0)

sample\_average\_estimator = SampleAverageEstimator(action\_value\_estimates.copy(), epsilon=0.1)

weighted\_estimator = WeightedEstimator(action\_value\_estimates.copy(), epsilon=0.1, alpha=0.1)

estimators = [sample\_average\_estimator, weighted\_estimator]

for step\_i in range(N\_STEPS):

for estimator\_i, estimator in enumerate(estimators):

action\_selected = estimator.select\_action()

is\_optimal = testbed.is\_optimal\_action(action\_selected)

reward = testbed.sample\_action(action\_selected)

estimator.update\_estimates(action\_selected, reward)

rewards[estimator\_i][run\_i][step\_i] = reward

optimal\_selections[estimator\_i][run\_i][step\_i] = is\_optimal

testbed.random\_walk\_action\_values()

plot\_performance(["Ɛ=0.1", "Ɛ=0.1 α=0.1"], np.array(rewards), np.array(optimal\_selections))

Chart, line chart

Description automatically generated

Chart, line chart

Description automatically generated