# **Workflow & Overview**

Code:

[implementation](file:///Users/sumit/Library/Application%20Support/JetBrains/PyCharm2024.3/scratches/Reinforcement%20Learning%20Finance/DRL_for_portfolio_trading_chen.py)

# Overview

A screenshot of a portfolio optimization tool

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screenshot of a device

Description automatically generated

A screenshot of a test

Description automatically generated

# **What is Differential Sharpe**

1. Running Statistics

At each time step t (after n returns have been observed), we maintain:

* Sample mean return:

* Sample variance (online update form of *Welford’s algorithm*):

where are the observed returns.

These are updated incrementally with the formula:

## 2. Sharpe Ratio Estimate

At each step, once n≥2:

Where

## 3. Differential Sharpe Ratio

The code computes something resembling the **incremental (differential) change in Sharpe ratio** from adding the new return

The returned value is:

## 4. Interpretation

* The first term

​​

represents the **marginal effect of the new return** on the Sharpe ratio (like a gradient term).

* The second term

acts as a **correction term** that penalizes deviations proportional to the squared residual, scaled by the Sharpe ratio.

**5. Summary**

Mathematically, this implements:

* Online computation of mean and variance of returns.
* At each new return ​, it outputs an approximation of the **instantaneous derivative of the Sharpe ratio** with respect to adding that return.

So in compact notation:

where , ​ are the updated sample mean and standard deviation after including ​.

# APPENDIX

1. <https://gatambook.substack.com/p/deep-reinforcement-learning-for-portfolio?utm_source=substack&utm_medium=email>