Monte Carlo

MC are a class of computational algorithms that use repeated random sampling to estimate numerical results. In the context of prediction problems, they are particularly useful when dealing with complex systems or models where analytical solutions are difficult or impossible to obtain.

How Monte Carlo Methods Work for Prediction

1. **Define the Problem:** Clearly specify the prediction task, including the input variables, output variable, and the underlying model or process.
2. **Generate Random Samples:** Create a large number of random samples from the input variable's distribution. This distribution can be based on historical data, expert knowledge, or assumptions about the system.
3. **Simulate Outcomes:** For each random sample, simulate the output variable using the defined model or process. This might involve running a complex simulation, evaluating a mathematical function, or applying a machine learning algorithm.
4. **Analyze Results:** Collect the simulated outcomes and analyze them to estimate the desired prediction. This could involve calculating the mean, median, variance, or other statistical measures.

Monte-Carlo Policy Evaluation

method used to estimate the value function of a given policy in a Markov Decision Process (MDP). The value function represents the expected cumulative discounted reward starting from a given state and following a specific policy.

Key Steps:

1. **Initialization:**
    - For each state `s`, initialize a counter `N(s)` to 0 and a cumulative return `S(s)` to 0. These variables will be used to track the number of visits to

the state and the total discounted return accumulated in that state, respectively.

2. **Episode Generation:**
   - Generate an episode by following the given policy π from an initial state until a terminal state is reached. An episode is a sequence of states and actions.
3. **Return Calculation:**
   - For each state `s` visited in the episode, calculate the discounted total return `G_t` from that state. This is the sum of discounted rewards from the current step `t` to the end of the episode. The discount factor γ determines the importance of future rewards relative to immediate ones.
4. **Value Function Update:**
   - Update the value function estimate for state `s` using the following formula:

     This updates the value function based on the newly observed return `G_t` and the accumulated statistics `S(s)` and `N(s)`.

$$V(s) = (S(s) + G_t)/(N(s) + 1)$$

5. **Repeat:**
   - Repeat steps 2-4 for many episodes to improve the accuracy of the value function estimate. As the number of episodes increases, the value function will converge to the true value function under the given policy.

Variants:

- **Every-Visit Monte Carlo:** The value function is updated every time a state is visited in an episode.
- **First-Visit Monte Carlo:** The value function is updated only the first time a state is visited in an episode.