Bayesian Scout Simulation

1 Overview

This project simulates how a **scout** learns to identify the best talent market using Bayesian updating. The environment contains:

- M markets
- each market contains many players with latent abilities
- one scout who observes noisy signals from players
- the scout updates posterior beliefs over market quality
- the scout chooses markets using different decision policies

The simulation compares the behavior of three exploration–exploitation strategies:

- Greedy
- ε-Greedy
- Thompson Sampling

The goal is to study how the scout allocates search effort over time and how quickly it identifies the best market.

2 Model

2.1 Player Ability

Each market m has a true distribution of player ability:

$$q \sim \mathcal{N}(\mu_m, \sigma_m^2),$$

where both μ_m and σ_m are **unknown to the scout**.

2.2 Signal Model

The scout does not observe true ability. Instead, they observe a noisy signal:

$$s_i = q_i + \varepsilon_i, \qquad \varepsilon_i \sim \mathcal{N}(0,\ \text{signal\_noise\_sd}^2).$$

The parameter `signal_noise_sd` controls the **true noise of the world**.

2.3 Scout's Belief Model

The scout assumes that:

$$s_i \mid \mu_m \sim \mathcal{N}(\mu_m,\ \text{assumed\_sigma\_sq}).$$

`assumed_sigma_sq` represents the scout's belief about observation noise variance and:

- it is fixed (never updated),
- it may differ from the true noise signal_noise_sd^2,
- it controls the learning speed in Bayesian updating.

2.4 Posterior Updating (Normal–Normal)

The scout maintains a Gaussian posterior over each market's mean:

$$\mu_m \mid s \sim \mathcal{N}(m_{m,t},\ \tau^2_{m,t}).$$

Posterior updates follow Normal–Normal conjugacy. Only the **posterior mean** and **posterior variance** are updated; the scout does *not* update noise variance.

2.5 Decision Policies

- **Greedy**: chooses the market with the highest posterior mean.
- **ε-Greedy**: with probability $1-\varepsilon$, chooses the best market; with probability ε, chooses a random market.
- **Thompson Sampling**: for each market, draws a sample from its posterior distribution and chooses the market with the highest draw. This policy uses both posterior mean and posterior variance.

3 Repository Structure

```
econ_capstone/

 config.py                  # Global simulation configuration
 bayes.py                   # Bayesian Normal{Normal model
 player.py                  # Player entity
 market.py                  # Market environment
 policy.py                  # Decision-making policies
 scout.py                   # Scout agent
 simulation.py              # Simulation engine
 visualize_extended.py      # Visualization module
 main.py                    # Entry point for running simulations
 README.md
```

4 Default Configuration Parameters

The default configuration is defined in `config.py`. Every parameter is described below.

4.1 Global Parameters

- **T**: number of periods (rounds) in the simulation.
- **X**: number of players observed per visit.

4.2 Market Parameters

Each market includes:

- **market_id**: unique index.
- **mu_true**: true mean ability of the market.
- **sigma_true**: true standard deviation of ability.
- **signal_noise_sd**: true standard deviation of observation noise.

4.3 Scout Belief Parameters

- **prior_m0**: prior mean belief about μ_m.

- **prior_tau0_sq**: prior variance of that belief.
- **assumed_sigma_sq**: scout's assumed observation variance.

Notes:

- `assumed_sigma_sq` never changes.
- It directly controls how quickly posterior variance shrinks.
- It may differ from true signal noise (`signal_noise_sd`).

4.4 Policy Parameter

- **epsilon**: probability of random exploration in the ε-greedy policy.

5 Visualization

The simulation generates:

- **Final visit distributions**: bar charts showing how many times each policy visits each market.
- **Choice probability curves**: multi-run averages of $P(\text{choose market } m \mid t)$.

These plots allow comparison of learning speed and exploration behavior across policies.

6 Running the Simulation

Use:

```
uv run econ_capstone/main.py
```

This runs all policies, updates beliefs, produces visualizations, and prints summary statistics.

7 Conceptual Summary

This project provides a clean Bayesian framework for analyzing exploration and learning under uncertainty. The model is relevant for applications in:

- sports scouting,
- labor markets and hiring,
- venture capital deal flow,
- search theory in economics,
- R&D and innovation search.

The core insight is how different exploration strategies—Greedy, ε-Greedy, and Thompson Sampling—allocate search effort and learn the latent quality of markets when observations are noisy.