Geometric Brownian Motion and Stock Price Forecasting

1. Geometric Brownian Motion (GBM)

Geometric Brownian Motion is a commonly used model for modeling stock prices. It assumes that stock prices follow a stochastic differential equation (SDE): $$dS_t = \mu S_t \, dt + \sigma S_t \, dW_t,$$ where:

- S_t: Stock price at time t.
- μ: Drift (expected rate of return).
- σ: Volatility (standard deviation of returns).
- W_t: Wiener process (standard Brownian motion).

The key feature of GBM is that it models the **logarithmic returns** as normally distributed, making it a natural fit for financial data.

---

2. Calculation Using Ito's Lemma

To derive the expected growth rate of stock prices under GBM, we use **Ito's Lemma**. For $f(S_t) = \ln(S_t)$, we have: $$d(\ln S_t) = \left(\mu - \frac{1}{2} \sigma^2\right) dt + \sigma dW_t.$$ Taking the expectation of both sides: $$\mathbb{E}[d(\ln S_t)] = \left(\mu - \frac{1}{2} \sigma^2\right) dt.$$ Exponentiating this result gives the expected growth rate: $$\mathbb{E}[S_t] = S_0 e^{\left(\mu - \frac{1}{2} \sigma^2\right)t}.$$ This shows that the growth rate is adjusted downward by $\frac{1}{2} \sigma^2$ due to volatility.

---

3. Ito's Lemma: Statement and Proof

Statement: Suppose $f(S_t)$ is twice differentiable with respect to S_t and once with respect to t. Then, for S_t following: $$dS_t = \mu S_t dt + \sigma S_t dW_t,$$ Ito's Lemma states: $$df(S_t) = \frac{\partial f}{\partial S_t} dS_t + \frac{\partial f}{\partial t} dt + \frac{1}{2} \frac{\partial^2 f}{\partial S_t^2} \sigma^2 S_t^2 dt.$$

Proof:

Expanding $f(S_t + dS_t)$ via Taylor series: $$f(S_t + dS_t) \approx f(S_t) + \frac{\partial f}{\partial S_t} dS_t + \frac{\partial f}{\partial t} dt + \frac{1}{2} \frac{\partial^2 f}{\partial S_t^2} (dS_t)^2.$$ Using properties of stochastic processes:

- $dW_t^2 = dt$ and higher-order terms ($dW_t \cdot dt$, dt^2) vanish. Substitute dS_t from the SDE, collect terms, and simplify.

---

4. Forecasting Using MLE of μ and σ

To forecast stock prices:

1. **Estimate Parameters**:

    - Compute log returns: $r_t = \ln(S_t) - \ln(S_{t-1})$.
    - Estimate μ (mean) and σ (standard deviation) from the historical log returns.
2. **Simulate Price Paths**:
    - Use the GBM equation: $$S_{t+1} = S_t \exp\left((\mu - 0.5\sigma^2)\Delta t + \sigma \sqrt{\Delta t} Z\right),$$ where $Z \sim \mathcal{N}(0, 1)$.
3. Generate multiple simulations and compute prediction intervals.

---

5. Upgrading Forecasts: Modeling μ and σ as Distributions

Instead of treating μ and σ as fixed, assume they follow distributions, e.g.:

- $\mu \sim \mathcal{N}(\mu_{\text{mean}}, \mu_{\text{std}})$,
- $\sigma \sim \mathcal{N}(\sigma_{\text{mean}}, \sigma_{\text{std}})$.

Benefits:

- Captures parameter uncertainty.
- Produces more realistic forecasts with wider, adaptive prediction intervals.

---

6. Validating μ and σ with Train-Validate-Test Split

Workflow:

1. **Split Data**:
    - Train: First 70% of the year's daily returns.
    - Validate: Next 20% for tuning hyperparameters or selecting priors.
    - Test: Last 10% for performance evaluation.
2. **Steps**:
    - Train the model using the training set to estimate μ and σ.
    - Validate against the validation set to check the predictive performance of priors.
    - Adjust priors or model complexity based on validation results.

Workflow Improvement:

- Use **rolling validation** to ensure robustness to different market conditions.
- Implement metrics like **coverage probability** (percentage of actual prices within forecast intervals).

---

7. Testing Against the Test Set

Finally, evaluate the model on the test set:

- Simulate future prices using learned distributions of μ and σ.
- Compute metrics:
    - **Mean Absolute Error (MAE)** and **Root Mean Square Error (RMSE)**.
    - **Prediction Interval Coverage**: Percentage of actual prices within the 95% prediction interval.

Workflow Improvement:

- Visualize the test set comparison, overlaying actual prices with the forecasted intervals.
- Report **statistical significance** of improvements using Bayesian methods versus MLE.

Workflow Improvements

1. **Automate Cross-Validation**:

    - Automate train-validate-test splitting and validation across multiple horizons.
    - Use tools like `scikit-learn`'s `TimeSeriesSplit`.

2. **Use Better Priors**:

    - Base priors on domain knowledge or sector-specific insights.
    - Regularize with weakly informative priors to avoid overfitting.

3. **Interpretability**:

    - Explain how posterior distributions influence decision-making.
    - Use visualizations of (\mu) and (\sigma) distributions for communication.

4. **Robust Metrics**:

    - Beyond MAE/RMSE, track long-term accuracy metrics like **Sharpe ratio** for trading strategies derived from forecasts.

By incorporating these steps, the workflow can be made more rigorous, transparent, and adaptive to market dynamics.