Day 3: Bayes' Theorem — Updating Beliefs with Evidence

The Mathematical Foundation of Learning from Data

1. The Fundamental Question

> *How do we systematically update our beliefs when new evidence becomes available?*

Bayes' Theorem provides the mathematical machinery for combining prior knowledge with new observations, forming the cornerstone of modern probabilistic AI and machine learning.

2. Mathematical Foundation

2.1. The Theorem

For events A and B with $P(B) > 0$:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

2.2. Extended Form Using Law of Total Probability

$$P(A|B) = \frac{P(B|A)P(A)}{P(B|A)P(A) + P(B|\neg A)P(\neg A)}$$

3.3. General Form for Multiple Hypotheses

For mutually exclusive hypotheses $H_1, H_2, \ldots, H_n$:

$$P(H_i|E) = \frac{P(E|H_i)P(H_i)}{\sum_{j=1}^{n} P(E|H_j)P(H_j)}$$

Bayesian Terminology

- **Prior ($P(A)$):** Initial belief before seeing evidence
- **Likelihood ($P(B|A)$):** Probability of evidence given hypothesis
- **Marginal Likelihood ($P(B)$):** Total probability of evidence
- **Posterior ($P(A|B)$):** Updated belief after seeing evidence
- **Bayesian Update:** Posterior $\propto$ Likelihood $\times$ Prior

3. Comprehensive Medical Testing Example

Problem Setup

- Disease prevalence: $P(D) = 0.01$ (1% of population)
- Test sensitivity: $P(T|D) = 0.99$ (99% true positive rate)
- Test specificity: $P(\neg T|\neg D) = 0.95$ (95% true negative rate)
- False positive rate: $P(T|\neg D) = 0.05$

3.1. Step-by-Step Calculation

Step 1: Define Complementary Probabilities

$$P(\neg D) = 1 - P(D) = 0.99, \quad P(\neg T|D) = 1 - P(T|D) = 0.01$$

Step 2: Calculate Marginal Probability of Positive Test

$$\begin{aligned} P(T) &= P(T|D)P(D) + P(T|\neg D)P(\neg D) \\ &= (0.99 \times 0.01) + (0.05 \times 0.99) \\ &= 0.0099 + 0.0495 = 0.0594 \end{aligned}$$

Step 3: Apply Bayes' Theorem

$$\begin{aligned} P(D|T) &= \frac{P(T|D)P(D)}{P(T)} \\ &= \frac{0.99 \times 0.01}{0.0594} \\ &= \frac{0.0099}{0.0594} \approx 0.1667 \end{aligned}$$

Counterintuitive Result Analysis

- **Initial intuition:** "99% accurate test" suggests high confidence
- **Reality:** Only 16.67% probability of actually having disease
- **Explanation:** Low prior probability dominates calculation
- **Base rate fallacy:** Ignoring prior probabilities leads to wrong conclusions

4. Why Bayes' Theorem is Fundamental to AI

Revolutionary Impact on Machine Learning

- **Bayesian Neural Networks:** Uncertainty-aware deep learning
- **Probabilistic Programming:** Flexible model specification
- **Reinforcement Learning:** Thompson sampling and Bayesian optimization
- **Computer Vision:** Bayesian inference for image understanding
- **Natural Language Processing:** Topic modeling and semantic analysis

5. Real-World AI Applications

5.1. Spam Detection Systems

- Prior: Base rate of spam emails (e.g., 20%)
- Likelihood: Probability of words appearing in spam vs. ham
- Posterior: Probability email is spam given its content
- Performance: Achieves 99%+ accuracy with proper training

5.2. Autonomous Vehicle Perception

- Prior: Expected objects in environment (cars, pedestrians)
- Likelihood: Sensor data patterns for different objects
- Posterior: Probability estimates for object classification
- Real-time: Updates beliefs as new sensor data arrives

5.3. Medical Diagnosis AI

- Prior: Disease prevalence in population
- Likelihood: Symptom patterns for different diseases
- Posterior: Diagnostic probabilities for patient
- Sequential: Updates diagnosis as test results come in

6. Computational Implementation

6.1. Python Implementation

```
def bayes_theorem(prior, likelihood, evidence):
    """Calculate posterior probability using Bayes' theorem"""
    return (likelihood * prior) / evidence

def medical_test_example():
    # Given probabilities
    p_disease = 0.01
    p_positive_given_disease = 0.99
    p_positive_given_no_disease = 0.05

    # Calculate evidence
    p_positive = (p_positive_given_disease * p_disease +
                 p_positive_given_no_disease * (1 - p_disease))

    # Calculate posterior
    p_disease_given_positive = bayes_theorem(
        p_disease, p_positive_given_disease, p_positive)

    return p_disease_given_positive
```

6.2. Numerical Stability Considerations

- Use log probabilities for numerical stability: $\log P(A|B) = \log P(B|A) + \log P(A) - \log P(B)$
- Handle zero probabilities with Laplace smoothing
- Use floating-point precision appropriate for small probabilities

7. Advanced Bayesian Concepts

7.1. Conjugate Priors

- Beta-Binomial: For binary outcomes
- Dirichlet-Multinomial: For categorical outcomes
- Normal-Normal: For continuous measurements
- Enable analytical posterior computation

7.2. Bayesian Inference Methods

- Markov Chain Monte Carlo (MCMC): Sampling-based approximation
- Variational Inference: Optimization-based approximation
- Expectation Propagation: Message-passing algorithm
- Laplace Approximation: Local Gaussian approximation

8. Performance Analysis and Empirical Results

Bayesian vs Frequentist Performance

- **Small datasets:** Bayesian methods typically outperform by 10-30%
- **Uncertainty quantification:** Bayesian models provide calibrated confidence intervals
- **Computational cost:** Bayesian inference can be 2-100x more expensive
- **Data efficiency:** Bayesian methods need 30-50% less data for same accuracy
- **Robustness:** More resistant to overfitting, especially with informative priors

9. Practical Exercises for Mastery

Hands-On Bayesian Reasoning

1. **Security System:** 2% hackers, 90% detection rate, 10% false alarms. Calculate $P(\text{hacker}|\text{flagged})$
2. **Dice Analysis:** For fair die, $A = \{\text{even}\}$, $B =$ {¿3}. Compute $P(A|B)$ using Bayes
3. **Parameter Estimation:** Given $P(A) = 0.3$, $P(B|A) = 0.8$, $P(B|\neg A) = 0.4$, find $P(A|B)$
4. **Sequential Updates:** Update posterior multiple times with new evidence
5. **Real Dataset:** Implement Naive Bayes classifier on UCI dataset

10. Common Pitfalls and Best Practices

Bayesian Modeling Guidelines

- **Base Rate Fallacy:** Always consider prior probabilities
- **Prior Selection:** Use informative priors when possible
- **Model Checking:** Validate posterior predictions
- **Computational Limits:** Choose inference method based on problem size
- **Interpretability:** Ensure stakeholders understand probabilistic outputs

11. Historical Context and Modern Impact

11.1. Historical Development

- Thomas Bayes (1701-1761): Original formulation
- Pierre-Simon Laplace: Formalized and popularized
- 20th Century: Frequentist dominance in statistics
- 21st Century: Bayesian renaissance in machine learning

11.2. Modern Bayesian Revolution

- MCMC algorithms made Bayesian computation feasible
- Probabilistic programming languages (Stan, PyMC3)
- Integration with deep learning (Bayesian neural networks)
- Applications in personalized medicine and AI safety

12. Key Insight: The Universal Learning Principle

Bayes' Theorem is not just a mathematical formula—it's a fundamental principle of rational reasoning and learning. Its power lies in providing a systematic framework for:

- **Evidence Integration:** Combining multiple sources of information
- **Uncertainty Quantification:** Expressing confidence in conclusions
- **Sequential Learning:** Updating beliefs with streaming data
- **Decision Making:** Choosing optimal actions under uncertainty
- **Model Comparison:** Evaluating competing hypotheses objectively

The Bayesian approach transforms machine learning from mere pattern recognition to genuine knowledge accumulation, where each new data point refines our understanding of the world.

Next: Random Variables and Probability Distributions

Tomorrow we'll explore how to mathematically represent uncertain quantities and their behavior—the building blocks for statistical modeling in AI.