Probability Theory for Machine Learning

Definition of Probability in ML

Probability is a measure of how likely an event is to occur. It quantifies uncertainty.

Mathematical Definition

For an event A, the probability P(A) is a number between 0 and 1:

```
0 ≤ P(A) ≤ 1
```

Where:

- **P(A) = 0**: Event A is impossible (never happens)
- **P(A) = 1**: Event A is certain (always happens)
- **P(A) = 0.5**: Event A has 50% chance of occurring

Frequency Interpretation

In practice, probability is often understood as a **relative frequency**:

```
P(A) = (Number of times A occurs) / (Total number of trials)
```

Example: Flip a coin 1000 times.

```
Heads appears 495 times
P(Heads) ≈ 495 / 1000 = 0.495 ≈ 0.5
```

With more flips, this estimate approaches the true probability (0.5).

Bayesian Interpretation

Probability can also represent **degree of belief** or confidence:

- P(rain tomorrow) = 0.7 means you're 70% confident it will rain
- P(model is correct) = 0.95 means you have 95% confidence in the model

This interpretation is useful when data is limited and we incorporate prior knowledge.

Why Probability Matters in ML

1. Modeling uncertainty: Real-world data is noisy. Probability quantifies this noise.

2. Decision-making under uncertainty: We can't know future outcomes with certainty, but probability lets us make informed decisions.

3. Learning from data: ML algorithms estimate probabilities from training data to make predictions.

4. Confidence in predictions: Instead of single predictions, probabilistic models output P(class A | data), allowing risk assessment.

5. Hypothesis testing: Determine if observed data supports a hypothesis using p-values (probability of data under null hypothesis).

6. Regularization and Bayesian inference: Probability provides principled framework for incorporating prior knowledge and preventing overfitting.

Random Experiments, Sample Space, and Events

These three concepts form the foundation of probability.

Random Experiment

A **random experiment** is any action or process with an uncertain outcome that can be repeated multiple times under similar conditions.

Examples:

- Flipping a coin
- Rolling a die
- Drawing a card from a deck
- Predicting tomorrow's weather
- Measuring a patient's blood pressure
- Training a neural network with random initialization

Key properties:

- Outcome is not determined in advance (uncertain)
- Can be repeated (at least in principle)
- Results vary despite same conditions (randomness)

Sample Space

The **sample space** is the set of all possible outcomes of a random experiment. Denoted as Ω (omega).

Examples:

Coin flip:

```
Ω = {Heads, Tails}
Size = 2 outcomes
```

Rolling a die:

```
Ω = {1, 2, 3, 4, 5, 6}
Size = 6 outcomes
```

Two coin flips:

```
Ω = {HH, HT, TH, TT}
Size = 4 outcomes
```

Temperature tomorrow:

```
Ω = [−50°C, 50°C]  (any real number in this range)
Size = infinite outcomes (continuous)
```

Events

An **event** is a subset of the sample space—a collection of possible outcomes we're interested in.

Examples:

Rolling a die:

- Event A = "rolling an even number" = {2, 4, 6}
- Event B = "rolling > 3" = {4, 5, 6}
- Event C = "rolling 5" = {5} (simple event, single outcome)

Two coin flips:

- Event A = "at least one heads" = {HH, HT, TH}
- Event B = "both same" = {HH, TT}

Temperature:

- Event A = "temperature > 20°C" = (20, 50] (interval on number line)

Probability of an Event

For a **finite sample space** with equally likely outcomes:

```
P(A) = |A| / |Ω|
```

Where |A| is the number of outcomes in event A, |Ω| is the total number of outcomes.

Example: Rolling a die

```
P(even number) = |{2, 4, 6}| / |{1, 2, 3, 4, 5, 6}| = 3 / 6 = 0.5
P(> 3) = |{4, 5, 6}| / 6 = 3 / 6 = 0.5
P(rolling a 5) = 1 / 6 ≈ 0.167
```

Complex Events

Events can be combined using set operations:

Union (A or B):

```
A ∪ B: outcomes in A OR B (or both)
P(A ∪ B) = P(A) + P(B) - P(A ∩ B)
```

Example:

```
Event A = rolling even = {2, 4, 6}, P(A) = 1/2
Event B = rolling > 3 = {4, 5, 6}, P(B) = 1/2
A ∩ B = {4, 6}, P(A ∩ B) = 2/6 = 1/3
P(A ∪ B) = 1/2 + 1/2 - 1/3 = 2/3
```

Intersection (A and B):

```
A ∩ B: outcomes in both A AND B
P(A ∩ B) = P(A) × P(B|A)  [conditional probability]
```

Complement (not A):

```
A^c: all outcomes NOT in A
P(A^c) = 1 - P(A)
```

Example:

```
P(not rolling even) = 1 - P(even) = 1 - 1/2 = 1/2
```

Law of Total Probability

If events $B_1, B_2, ..., B_n$ partition the sample space (mutually exclusive and exhaustive):

```
P(A) = Σ P(A ∩ Bᵢ) = Σ P(A | Bᵢ) × P(Bᵢ)
```

This breaks down a complex probability into simpler conditional probabilities.

Example: Disease diagnosis

```
P(positive test) = P(positive | disease) × P(disease) + P(positive | no disease) × P(no disease)
                 = 0.99 × 0.01 + 0.05 × 0.99
                 = 0.0099 + 0.0495 = 0.0594
```

Conditional Probability

The probability of event A given that event B has occurred:

```
P(A | B) = P(A ∩ B) / P(B)
```

This is the foundation of **Bayes' Theorem**, critical to ML:

```
P(A | B) = P(B | A) × P(A) / P(B)
```

Example: Medical testing

```
P(disease | positive test) = P(test positive | disease) × P(disease) / P(test positive)
                           = 0.99 × 0.01 / 0.0594 ≈ 0.167
```

Even with a 99% accurate test, a positive result only means 16.7% chance of disease (if disease is rare). This is **base rate fallacy**.

Expectation (Expected Value)

The **expected value** (also called expectation or mean) is the long-run average outcome of a random experiment.

Mathematical Definition

For a **discrete** random variable X that takes values $x_1, x_2, ..., x_n$ with probabilities $p_1, p_2, ..., p_n$:

$$E[X] = x_1p_1 + x_2p_2 + ... + x_np_n = \Sigma\, x_i \times P(X = x_i)$$

For a **continuous** random variable with probability density function f(x):

$$E[X] = \int x \times f(x)\, dx$$

Intuition

The expected value is the **weighted average** where weights are probabilities.

Example 1: Fair die

```
X = outcome of rolling a fair die
X can be: 1, 2, 3, 4, 5, 6 (each with probability 1/6)

E[X] = 1×(1/6) + 2×(1/6) + 3×(1/6) + 4×(1/6) + 5×(1/6) + 6×(1/6)
     = (1 + 2 + 3 + 4 + 5 + 6) / 6
     = 21 / 6
     = 3.5
```

If you roll the die many times, the average is approximately 3.5.

Example 2: Unfair coin

```
X = payout if heads, 0 if tails
Heads: $10 with probability 0.6
Tails: $0 with probability 0.4

E[X] = 10 × 0.6 + 0 × 0.4 = $6
```

On average, you make $6 per flip.

Properties of Expectation

Linearity:

$$E[aX + b] = a \times E[X] + b$$

Scaling and shifting preserve the relationship.

Additivity (works even for dependent variables):

```
E[X + Y] = E[X] + E[Y]
```

The expected sum equals the sum of expected values.

Example:

```
X = first die roll, E[X] = 3.5
Y = second die roll, E[Y] = 3.5
E[X + Y] = E[X] + E[Y] = 3.5 + 3.5 = 7
```

Independence (only for independent variables):

```
If X and Y are independent: E[X × Y] = E[X] × E[Y]
```

Applications in ML

1. Loss function minimization: ML algorithms minimize **expected loss** (average error on all data):

```
min E_data[L(predictions, true_values)]
```

2. Risk assessment: Expected value of different decisions:

```
E[profit | strategy A] vs E[profit | strategy B]
```

Choose the strategy with higher expected value.

3. Feature importance: Expected impact of each feature on predictions.

4. Uncertainty quantification: Instead of single prediction, output E[Y | data] = expected value of outcome given data.

5. Reinforcement learning: Expected future reward guides learning:

```
E[future_reward | action] guides which action to take
```

Example: Expected Value in Medical Decisions

A patient has two treatment options:

Option A: Risky surgery

- 80% chance of recovery (restore $1M quality of life)

- 20% chance of severe complications (-$500k in costs)

```
E[A] = 0.8 × $1M + 0.2 × (-$500k) = $800k - $100k = $700k
```

Option B: Conservative treatment

- Guaranteed partial recovery ($300k improvement)

```
E[B] = $300k
```

Decision: Option A has higher expected value ($700k vs $300k), but higher risk. The choice depends on risk tolerance.

Variance and Standard Deviation of a Probability Distribution

While **expectation** measures the center of a distribution, **variance** measures spread around that center.

Mathematical Definition of Variance

Variance measures how much a random variable deviates from its expected value on average:

```
Var(X) = E[(X - E[X])²]
```

Or equivalently (using a useful trick):

```
Var(X) = E[X²] - (E[X])²
```

Step-by-step calculation:

1. Find $E[X]$ (expected value)
2. Find $E[X^2]$ (expected value of X squared)
3. Compute $E[X^2] - (E[X])^2$

Example: Fair Die

```
X = outcome of fair die
E[X] = 3.5 (computed earlier)

E[X²] = 1² × (1/6) + 2² × (1/6) + 3² × (1/6) + 4² × (1/6) + 5² × (1/6) + 6² × (1/6)
     = (1 + 4 + 9 + 16 + 25 + 36) / 6
```

$$= 91 / 6$$
$$\approx 15.167$$

$$\mathrm{Var}(X) = E[X^2] - (E[X])^2$$
$$= 15.167 - (3.5)^2$$
$$= 15.167 - 12.25$$
$$= 2.917$$

Standard Deviation

Standard deviation is the square root of variance:

$$SD(X) = \sigma(X) = \sqrt{\mathrm{Var}(X)}$$

This brings variance back to original units (easier to interpret).

From the die example:

$$SD(X) = \sqrt{2.917} \approx 1.71$$

This means outcomes typically deviate from 3.5 by about 1.71.

Variance Comparison

Comparing two investments:

Stock A:

- 50% chance of +20% return
- 50% chance of −10% return
- $E[A] = 0.5 \times 20 + 0.5 \times (-10) = 5\%$
- $\mathrm{Var}(A) = 0.5 \times (20-5)^2 + 0.5 \times (-10-5)^2 = 0.5 \times 225 + 0.5 \times 225 = 225$
- $SD(A) = \sqrt{225} = 15\%$

Stock B:

- 100% chance of +5% return
- $E[B] = 5\%$
- $\mathrm{Var}(B) = 0$ (no variability)
- $SD(B) = 0\%$

Both have the same expected return (5%), but Stock A is much riskier (variance 225 vs 0).

Properties of Variance

Scaling:

$$\mathrm{Var}(aX) = a^2 \times \mathrm{Var}(X)$$

Doubling a random variable quadruples variance (because variance depends on squared deviations).

Adding constants:

$$\mathrm{Var}(X + b) = \mathrm{Var}(X)$$

Shifting a random variable doesn't change variance (deviation from mean unchanged).

Sums of independent variables:

$$\mathrm{Var}(X + Y) = \mathrm{Var}(X) + \mathrm{Var}(Y) \quad \text{[if X and Y independent]}$$

Variance of sum equals sum of variances.

Example:

```
Total risk = Var(investment A) + Var(investment B) if they're uncorrelated
```

Relationship to Empirical Data

When you have actual data (not just probabilities), these formulas become:

Sample mean (estimates E[X]):

$$\bar{x} = \Sigma x_i / n$$

Sample variance (estimates Var(X)):

$$s^2 = \Sigma(x_i - \bar{x})^2 / (n-1)$$

Sample standard deviation:

$$s = \sqrt{s^2}$$

The division by (n−1) instead of n is called **Bessel's correction**—it makes the sample variance an unbiased estimate of population variance.

Applications in ML

1. Model uncertainty: **Low variance model**: Stable predictions, doesn't change much with training data changes. **High variance model**: Predictions vary wildly, overfits to noise.

2. Confidence intervals: A 95% confidence interval for parameter θ is approximately:

```
[E[θ] − 1.96 × SD(θ), E[θ] + 1.96 × SD(θ)]
```

Wider intervals indicate more uncertainty.

3. Regularization: Penalizing model variance (L2 regularization):

```
Loss = prediction_error + λ × Var(weights)
```

This discourages large, variable weights that might fit noise.

4. Bayesian inference: Posterior distribution has mean (point estimate) and variance (uncertainty):

```
Posterior = (E[θ|data], Var(θ|data))
```

5. Ensemble methods: Combining models reduces variance:

```
Var(average of predictions) = Var(single prediction) / n_models
```

This is why averaging predictions improves robustness.

The Bias-Variance Tradeoff

In machine learning, **total error** has two components:

```
Total Error = Bias² + Variance + Irreducible Noise
```

High bias, low variance:

- Model is too simple
- Consistent but wrong
- Underfitting

Low bias, high variance:

- Model is too complex
- Fits training data perfectly but unstable
- Overfitting

Optimal balance:

- Right model complexity
- Good generalization

Understanding variance helps navigate this tradeoff.

Putting It Together: A Complete Example

Scenario: Predicting Customer Churn

Random experiment: Whether a customer churns next month **Sample space**: Ω = {churn, stay}

Event of interest: A = {customer churns} **Probability**: P(churn) = 0.2 (20% of customers churn)

Computing Expectations

Let's say churning costs the company $1000 in lost revenue:

```
X = cost if customer churns
X = $1000 if churn (probability 0.2)
X = $0 if stay (probability 0.8)

E[X] = $1000 × 0.2 + $0 × 0.8 = $200
```

Interpretation: On average, each customer represents an expected $200 churn risk.

Computing Variance

```
E[X²] = ($1000)² × 0.2 + ($0)² × 0.8 = $200,000

Var(X) = E[X²] - (E[X])²
      = $200,000 - ($200)²
      = $200,000 - $40,000
      = $160,000

SD(X) = √$160,000 ≈ $400
```

Interpretation: There's $400 standard deviation around the $200 average—significant uncertainty.

Decision Under Uncertainty

The company can spend $150 per customer on retention program with 80% success rate:

```
Option A: No intervention
E[loss] = $200

Option B: Intervention
P(churn despite intervention) = 0.2 × 0.2 = 0.04
E[loss] = $1000 × 0.04 + $150 = $190

Decision: Intervene (saves $10 per customer)
```

With 10,000 customers: $10 × 10,000 = $100,000 annual savings.

Summary: Probability as the Language of Uncertainty

Sample space and events define what outcomes are possible and which we care about.

Probability quantifies likelihood of events.

Expectation (expected value) is the long-run average outcome—what we expect on average.

Variance and standard deviation quantify uncertainty around that average—how much outcomes typically vary.

Together, these form the foundation for:

- Making decisions under uncertainty
- Building probabilistic models
- Quantifying model confidence
- Understanding overfitting and generalization
- Designing robust systems

Master probability, and you understand why ML models work, how to interpret their predictions, and how to make good decisions when facing uncertainty.