# Detailed Hand Calculations for Bayesian Models

## Part 1: Beta-Binomial Model

### 1.1 Model Setup

Given:

- Prior: $\theta \sim \text{Beta}(\alpha_0, \beta_0)$
- Data: $y = 2$ successes in $N = 12$ trials
- Likelihood: $y|\theta \sim \text{Binomial}(N, \theta)$

Let's use $\alpha_0 = 1$, $\beta_0 = 1$ (uniform prior)

### 1.2 Prior Distribution

The Beta prior density is:

```
p(θ) = [1/B(α₀, β₀)] × θ^(α₀-1) × (1-θ)^(β₀-1)
     = [1/B(1, 1)] × θ^0 × (1-θ)^0
     = 1  (uniform on [0,1])
```

where $B(\alpha_0, \beta_0) = \Gamma(\alpha_0)\Gamma(\beta_0)/\Gamma(\alpha_0+\beta_0) = 1\times1/1 = 1$

### 1.3 Likelihood

The binomial likelihood is:

```
p(y|θ) = C(N,y) × θ^y × (1-θ)^(N-y)
       = C(12,2) × θ^2 × (1-θ)^10
       = 66 × θ^2 × (1-θ)^10
```

where $C(12,2) = 12!/(2!\times10!) = (12\times11)/(2\times1) = 66$

### 1.4 Joint Distribution

```
p(y,θ) = p(y|θ) × p(θ)
       = 66 × θ^2 × (1-θ)^10 × 1
       = 66 × θ^2 × (1-θ)^10
```

### 1.5 Log Joint Distribution

```
log p(y,θ) = log(66) + 2×log(θ) + 10×log(1-θ)
           = constant + 2×log(θ) + 10×log(1-θ)
```

More generally:

```
log p(y,θ) = (α₀ + y - 1)×log(θ) + (β₀ + N - y - 1)×log(1-θ) + constant
          = (1 + 2 - 1)×log(θ) + (1 + 12 - 2 - 1)×log(1-θ) + constant
          = 2×log(θ) + 10×log(1-θ) + constant
```

## 1.6 Gradient of Log Joint

```
d/dθ log p(y,θ) = (α₀ + y - 1)/θ - (β₀ + N - y - 1)/(1-θ)
                = 2/θ - 10/(1-θ)
```

## 1.7 Setting Gradient to Zero (Finding MAP)

```
2/θ - 10/(1-θ) = 0
2/θ = 10/(1-θ)
2(1-θ) = 10θ
2 - 2θ = 10θ
2 = 12θ
θ_MAP = 2/12 = 1/6 ≈ 0.167
```

## 1.8 Posterior Distribution

By Bayes' theorem:

```
p(θ|y) ∝ p(y|θ) × p(θ)
       ∝ θ^2 × (1-θ)^10 × 1
       ∝ θ^(2+1-1) × (1-θ)^(10+1-1)
       ∝ θ^2 × (1-θ)^10
```

This is Beta($\alpha_0$ + y, $\beta_0$ + N - y) = Beta(3, 11)

## 1.9 Posterior Mean and Variance

```
E[θ|y] = α/(α + β) = 3/(3 + 11) = 3/14 ≈ 0.214

Var[θ|y] = αβ/[(α+β)²(α+β+1)]
         = (3×11)/[(14)²(15)]
         = 33/(196×15)
         = 33/2940
         ≈ 0.0112
```

---

## Part 2: Bayesian Logistic Regression

### 2.1 Model Setup

Given:

- Prior: $w \sim N(0, \alpha^{-1}I)$ with $\alpha = 1$
- Likelihood: $y_n|w,x_n \sim \text{Bernoulli}(\sigma(w^T x_n))$
- Sigmoid: $\sigma(z) = 1/(1 + e^{(-z)})$

Example data:

- $x_1 = [1.0, 0.5]$, $y_1 = 1$
- $x_2 = [-0.5, 1.0]$, $y_2 = 0$
- $x_3 = [0.3, -0.8]$, $y_3 = 1$

### 2.2 Log Prior

For $w = [w_1, w_2]^T$:

```
log p(w) = -½α||w||² + constant
         = -½(w₁² + w₂²) + constant
```

### 2.3 Log Likelihood

```
log p(y|w) = Σₙ [yₙ log σ(fₙ) + (1-yₙ) log(1-σ(fₙ))]
```

where $f_n = w^T x_n$

For our data:

- $f_1 = w_1 \times 1.0 + w_2 \times 0.5 = w_1 + 0.5w_2$
- $f_2 = w_1 \times (-0.5) + w_2 \times 1.0 = -0.5w_1 + w_2$

- $f_3 = w_1 \times 0.3 + w_2 \times (-0.8) = 0.3w_1 - 0.8w_2$

### 2.4 Log Joint

```
log p(y,w) = log p(w) + log p(y|w)
           = -½(w₁² + w₂²) +
             [log σ(w₁ + 0.5w₂) +
              log(1-σ(-0.5w₁ + w₂)) +
              log σ(0.3w₁ - 0.8w₂)]
```

### 2.5 Gradient Calculation

Using the identity: $d/dz \log \sigma(z) = 1 - \sigma(z)$

```
∇w log p(y,w) = -αw - Σₙ (σ(fₙ) - yₙ)xₙ
```

Component-wise:

```
∂/∂w₁ log p = -w₁ - [(σ(f₁) - 1)×1.0 + (σ(f₂) - 0)×(-0.5) + (σ(f₃) - 1)×0.3]
            = -w₁ - [σ(f₁) - 1 - 0.5σ(f₂) + 0.3(σ(f₃) - 1)]

∂/∂w₂ log p = -w₂ - [(σ(f₁) - 1)×0.5 + (σ(f₂) - 0)×1.0 + (σ(f₃) - 1)×(-0.8)]
            = -w₂ - [0.5(σ(f₁) - 1) + σ(f₂) - 0.8(σ(f₃) - 1)]
```

### 2.6 Hessian Calculation

The Hessian is:

```
H = -X^T S X - αI
```

where S is diagonal with $S_{nn} = \sigma(f_n)(1-\sigma(f_n))$

For our example:

```
X = [1.0   0.5]
    [-0.5  1.0]
    [0.3  -0.8]

S = diag([σ(f₁)(1-σ(f₁)), σ(f₂)(1-σ(f₂)), σ(f₃)(1-σ(f₃))])
```

### 2.7 MAP Optimization

To find w_MAP, solve ∇log p(y,w) = 0

This requires iterative optimization. Suppose we find: w_MAP ≈ [0.8, -0.3]^T

## 2.8 Laplace Approximation

At w_MAP:

1. Evaluate Hessian H
2. Posterior covariance: $S = -H^{-1}$
3. Posterior: q(w) = N(w|w_MAP, S)

Example calculation (simplified):

```
H ≈ -[2.5  -0.8]  - [1  0] = [-3.5   0.8]
     [-0.8  2.1]    [0  1]   [0.8  -3.1]

S = -H⁻¹ ≈ [0.32  0.08]
           [0.08  0.31]
```

## 2.9 Posterior Predictive for New Point x* = [0.7, -0.2]

### Step 1: Distribution of f* = w^T x*

```
E[f*] = w_MAP^T x* = 0.8×0.7 + (-0.3)×(-0.2) = 0.56 + 0.06 = 0.62
Var[f*] = x*^T S x* = [0.7, -0.2] × [0.32  0.08] × [0.7 ]
                                    [0.08  0.31]   [-0.2]
        = [0.7, -0.2] × [0.208]
                        [0.006]
        = 0.1456 + 0.0012 = 0.147
```

### Step 2: Plugin Approximation

```
p(y*=1|x*, y) ≈ σ(E[f*]) = σ(0.62) = 1/(1 + e^(-0.62)) ≈ 0.650
```

### Step 3: Probit Approximation

```
p(y*=1|x*, y) ≈ Φ(E[f*]/√(Var[f*] + π/8))
              = Φ(0.62/√(0.147 + 0.393))
              = Φ(0.62/√0.540)
              = Φ(0.844)
              ≈ 0.801
```

**Step 4: Monte Carlo (conceptual)**

```
1. Sample f*₁, f*₂, ..., f*₁₀₀₀ ~ N(0.62, 0.147)
2. Compute σ(f*ᵢ) for each sample
3. p(y*=1|x*, y) ≈ (1/1000) Σᵢ σ(f*ᵢ)
```

Example samples:

- $f^*_1 = 0.62 + 0.383\times\varepsilon_1$ ($\varepsilon_1 \sim N(0,1)$)
- $\sigma(f^*_1) = \sigma(0.62 + 0.383\times\varepsilon_1)$
- Average over all samples ≈ 0.745

## Summary of Predictions

For x* = [0.7, -0.2]:

- Plugin: 0.650 (ignores uncertainty)
- Probit: 0.801 (accounts for uncertainty)
- Monte Carlo: 0.745 (most accurate)

The differences show how accounting for parameter uncertainty affects predictions!