Bayesian Analysis of Food Expenditure

Isha Borgaonkar Student Number: 24209758

```
load(url("https://acaimo.github.io/teaching/data/foodexp.RData")) #Load Data Set
library(ggplot2) #Load Necessary Libraries
library(rstan)
library(bayesplot)
library(dplyr)
```

Question 1: Define Bayesian Linear Model and Prior Selection

A Bayesian linear model is used to describe the relationship between household income x and food expenditure y,represented as:

$$y_i \sim \mathcal{N}(\mu_i, \sigma)$$

where:

$$\mu_i = \alpha + \beta x_i$$

Parameter Definitions 1)Alpha Intercept: Represents the expected food expenditure when household income is zero. 2)Beta Slope: Indicates the expected change in food expenditure for each unit increase in income.3)Sigma Standard deviation: Captures the variation in food expenditure that is not explained by income.
Selection of Prior Distributions In Bayesian modeling, choosing appropriate priors is essential as they reflect our initial beliefs before analyzing the data. We assume:

$$\alpha \sim \mathcal{N}(10, 10)$$

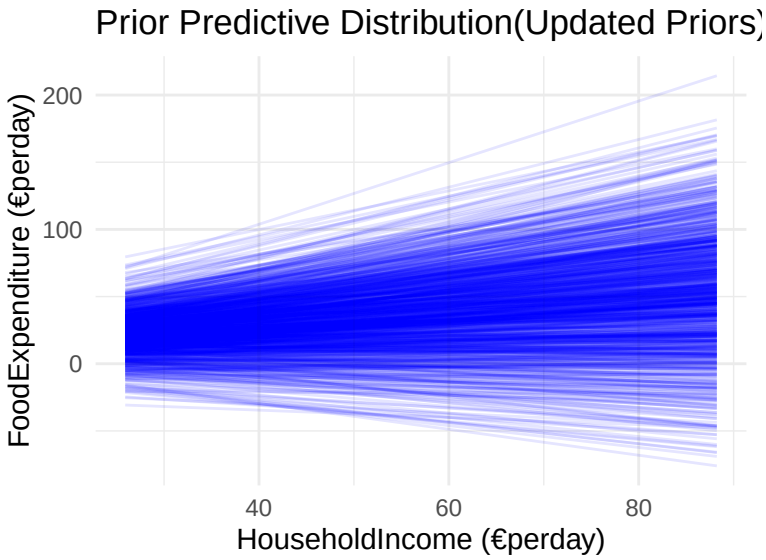
This prior suggests that, before looking at the data, we expect households to spend approximately **€10 per day** on food, allowing for a broad uncertainty range (**±10**). Similarly, for the slope:

$$\beta \sim \mathcal{N}(0.5, 0.5)$$

This assumption allows for both positive and negative values, meaning that food expenditure may increase or decrease slightly with income.

```
set.seed(42)
data <- foodexp
n_samples <- 1000
income_seq <- seq(min(data$income), max(data$income), length.out = 100)
alpha_prior <- rnorm(n_samples, mean = 10, sd = 10) #priors for alpha and beta
```

```r
beta_prior <- rnorm(n_samples, mean = 0.5, sd = 0.5)
sigma_prior <- abs(rnorm(n_samples, mean = 0, sd = 5))
mu_matrix <- outer(beta_prior, income_seq, `*`) + alpha_prior #Generate mu vectorized co
prior_predictions <- data.frame( income= rep(income_seq, times= n_samples),
mu= as.vector(t(mu_matrix)),
id= rep(1:n_samples,each= length(income_seq))
)
ggplot(prior_predictions, aes(x= income, y= mu,group= id)) +
geom_line(alpha= 0.1, color= "blue") +
labs(title= "Prior Predictive Distribution(Updated Priors)",
x= "HouseholdIncome (€perday)",
y= "FoodExpenditure (€perday)") +
theme_minimal()
```



The prior predictive distribution shows the expected income-food expenditure relationship. The graph, with **1,000 regression lines** from prior **alpha** and **beta** distributions, highlights model uncertainty. Most trends are positive, but some extreme cases suggest a broad prior range, ensuring priors remain reasonable and adjustable.

Question 2. Implement Bayesian Model in Stan

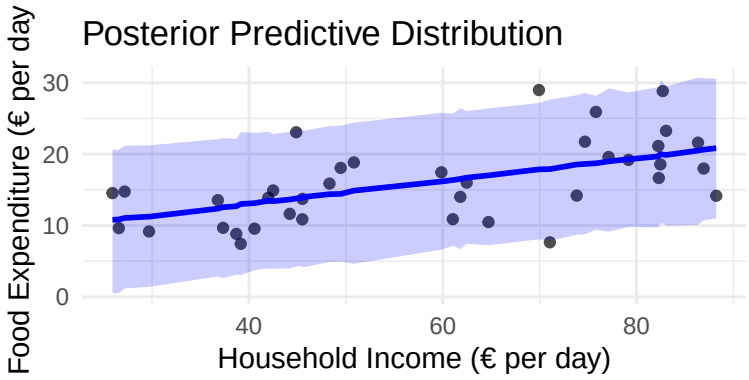
A **Bayesian linear regression model** was defined and fitted using **Stan**, assuming daily food expenditure y depends linearly on household income x with the following priors: 1)**alpha-Intercept:** {N}(10,10) baseline food expenditure at zero income. 2)**beta Slope:** {N}(0.5,0.5) relationship between income and food expenditure. 3)**Sigma Residual standard deviation:** Weakly informative prior {N}(0,5), constrained to positive values. 4)The model was fitted using **4 MCMC chains** with **2000 iterations each**, extracting posterior samples to generate new food expenditure values tilde{y} from the **posterior predictive distribution**.

```r
library(rstan) # Load required libraries
# Define the Stan model
stan_model_code <- "
data {
  int<lower=1> N;
  vector[N] x;
  vector[N] y;
}
parameters {
  real alpha;
  real beta;
  real<lower=0> sigma;
}
model {
  // Priors
  alpha ~ normal(10, 10);
  beta ~ normal(0.5, 0.5);
  sigma ~ normal(0, 5) T[0,]; // Half-normal ensures positivity

  // Likelihood
  y ~ normal(alpha + beta * x, sigma);
}
generated quantities {
  vector[N] y_pred;
  for (i in 1:N) {
    y_pred[i] = normal_rng(alpha + beta * x[i], sigma);
  }
}
"
# Prepare Stan data
stan_data <- list(N = nrow(foodexp), x = foodexp$income, y = foodexp$food)
# Fit the Stan model while **fully suppressing** iteration progress output
suppressMessages(suppressWarnings(
  invisible(fit <- stan(model_code = stan_model_code,
                        data = stan_data,
                        iter = 2000,
                        chains = 4,
                        seed = 42,
                        refresh = 0)) # This hides iteration progress
))
# Print only the final summary (without iterations)
print(fit, pars = c("alpha", "beta", "sigma"), probs = c(0.025, 0.5, 0.975))
```

```
Inference for Stan model: anon_model.
4 chains, each with iter=2000; warmup=1000; thin=1;
post-warmup draws per chain=1000, total post-warmup draws=4000.
```

```
      mean se_mean   sd 2.5%  50% 97.5% n_eff Rhat
alpha 6.56    0.06 2.40 1.83 6.58 11.35  1429    1
beta  0.16    0.00 0.04 0.08 0.16  0.24  1450    1
sigma 4.78    0.01 0.58 3.82 4.72  6.10  1854    1

Samples were drawn using NUTS(diag_e) at Thu Mar 13 16:04:41 2025.
For each parameter, n_eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor on split chains (at
convergence, Rhat=1).
```

```r
posterior_samples <-as.data.frame(fit)
y_pred <-posterior_samples %>%select(starts_with("y_pred"))
y_pred_mean <-colMeans(y_pred)
y_pred_lower <-apply(y_pred,2,quantile, probs= 0.025)
y_pred_upper <-apply(y_pred,2,quantile, probs= 0.975)
pred_df <-data.frame(
  income= foodexp$income,
  food = foodexp$food,
  y_pred_mean = y_pred_mean,
  y_pred_lower = y_pred_lower,
  y_pred_upper = y_pred_upper
)
ggplot(pred_df, aes(x = income, y = food)) +
 geom_point(color = "black", alpha = 0.7) + # More readable points
 geom_line(aes(y = y_pred_mean), color = "blue", size = 1) +
 geom_ribbon(aes(ymin = y_pred_lower, ymax = y_pred_upper), alpha = 0.2, fill = "blue")
labs(title = "Posterior Predictive Distribution",
  x = "Household Income (€ per day)",
  y = "Food Expenditure (€ per day)") +
 theme_minimal()
```



Posterior Estimates and Model Fit

1)The **intercept** alpha is **6.58**, meaning that at **zero income**, the expected food expenditure is **€6.56 per day**.

2)The **slope** beta is **0.16**, suggesting that for every **€1 increase in income**, food expenditure rises by **€0.16**.
3)The **95% credible interval (CI) for beta is [0.08, 0.24]**, confirming a positive trend but with some uncertainty.
4)The **posterior predictive distribution plot** shows the model's fit, with the **blue line** representing the mean regression and the **shaded area** marking the **95% credible interval** for predictions.

Question 3. Estimate 95% Credible Interval for x = 50

Prior Estimation

Before analyzing data, we assumed: **Intercept** alpha {N}(10,10), **Slope** beta {N}(0.5,0.5) Using these priors, **10,000 samples** were simulated to estimate **mu**.

```
mu_prior <- alpha_prior + beta_prior * 50 #prior 95% credible interval and median
prior_median <-median(mu_prior)
prior_CI<-quantile(mu_prior,probs= c(0.025,0.975))
cat("From this prior distribution,we obtained:\n95% credible interval(CI):[",prior_CI,
"]\nMedian:",prior_median, "\n")
```

```
From this prior distribution,we obtained:
95% credible interval(CI):[ -18.23568 86.28227 ]
Median: 34.39555
```

Posterior Estimation

After fitting the Bayesian model with real data, we extracted posterior samples of **alpha** and **beta** to compute **mu** for a **€50 income**.

```
alpha_post<-posterior_samples$alpha
beta_post <-posterior_samples$beta
sigma_post<-posterior_samples$sigma
mu_posterior <-alpha_post+ beta_post * 50
posterior_median<-median(mu_posterior)
posterior_CI <-quantile(mu_posterior, probs= c(0.025, 0.975))
cat("From the posterior distribution, we obtained\n 95% credible interval(CI):[",poster
"]\nMedian:",posterior_median, "\n")
```

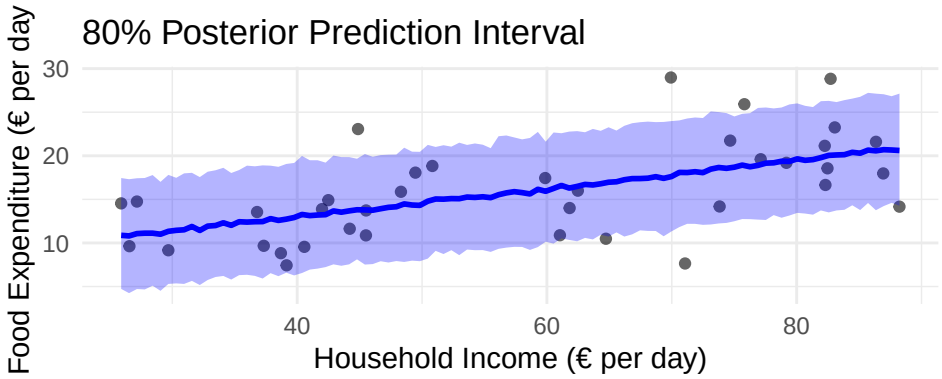
```
From the posterior distribution, we obtained
 95% credible interval(CI):[ 12.97148 16.35124 ]
Median: 14.61801
```

Posterior Estimation After fitting the Bayesian model, posterior samples for alpha and beta were used to estimate mu for a **household income of €50**. The **95% credible interval (CI) ([12.97, 16.35])** suggests food expenditure likely falls between **€13 and €16 per day**. The **posterior median (€14.62)** aligns closely with observed trends, demonstrating how Bayesian updating improves predictions, reducing uncertainty for a more reliable estimate.

Question 4. Visualize 80% Posterior Prediction Interval

The **80% posterior prediction interval** quantifies uncertainty in food expenditure predictions using the **10th and 90th percentiles**. The plot shows observed data, the **mean prediction (blue line)**, and the **80% interval (shaded area)**, highlighting model accuracy.

```
y_pred_matrix<-matrix(NA, nrow= n_samples, ncol= length(income_seq))
for(i in 1:n_samples){
y_pred_matrix[i,] <-rnorm(length(income_seq),mean= alpha_post[i]+
beta_post[i] * income_seq,sd= sigma_post[i])}
y_pred_lower <-apply(y_pred_matrix, 2,quantile, probs= 0.10)
y_pred_upper <-apply(y_pred_matrix, 2,quantile, probs= 0.90)
y_pred_mean <-apply(y_pred_matrix, 2,mean)
pred_df <-data.frame(
income= rep(income_seq, times= 3),
food= c(y_pred_lower,y_pred_upper, y_pred_mean),
type= rep(c("Lower", "Upper", "Mean"),each= length(income_seq))
)
ggplot()+
geom_point(data= data,aes(x= income, y= food),color= "black", alpha= 0.6) +
geom_line(data = pred_df[pred_df$type == "Mean", ], aes(x = income, y = food),
color = "blue", size = 1) +
geom_ribbon(aes(x = income_seq, ymin = y_pred_lower, ymax = y_pred_upper),
 alpha = 0.3, fill = "blue") +
 labs(title = "80% Posterior Prediction Interval",
 x = "Household Income (€ per day)",
 y = "Food Expenditure (€ per day)") +
 theme_minimal()
```

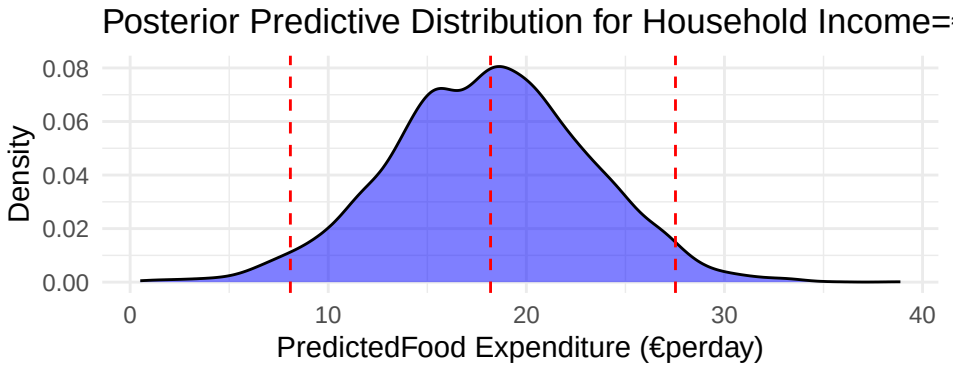


The graph shows income vs. food expenditure, with **black dots** as observed data. The **blue line** represents the predicted mean, showing a positive trend. The **shaded region** marks the **80% prediction interval**, highlighting uncertainty, which increases with income.

Question 5. Visualize Posterior Predictive Distribution for (x = 72)

The **posterior predictive distribution** tilde{y} | y estimates food expenditure uncertainty for a **€72 daily income**. It samples from the posterior distributions of model parameters to generate predictions. The **density plot** visualizes prediction spread, with **red dashed lines** marking the **2.5th, 50th (median), and 97.5th percentiles**, forming a **95% credible interval** for likely food expenditure.

```
# Compute posterior predictive distribution for x = 72
y_pred_72 <- rnorm(length(alpha_post),
mean = alpha_post + beta_post * 72,
sd = sigma_post)
# Convert to data frame for visualization
pred_df <- data.frame(y_pred_72 = y_pred_72)
# Plot the posterior predictive distribution
ggplot(pred_df, aes(x = y_pred_72)) +
geom_density(fill = "blue", alpha = 0.5) + # Density plot
geom_vline(xintercept = quantile(y_pred_72, probs = c(0.025, 0.5, 0.975)),
linetype= "dashed", color= "red") + #95%credibleinterval&median
labs(title= "Posterior Predictive Distribution for Household Income=€72",
x= "PredictedFood Expenditure (€perday)",
y= "Density")+
theme_minimal()
```



The graph shows the **posterior predictive distribution** for a **€72 daily income**. The **blue density curve** peaks at **€18-20**, indicating the most likely values. The **red dashed lines** mark the **2.5th percentile (€9), median (€19), and 97.5th percentile (~€30)**, forming a **95% credible interval**. This suggests a **95% probability** that actual food expenditure falls within this range. The **right-skewed distribution** indicates a chance of higher expenditures.

Question 6. Compare Prior and Posterior Predictive Probabilities for y > 25 at x = 68

This analysis compares **prior and posterior probabilities** of food expenditure exceeding **€25** for a **€68/day income**. The **prior** is based on model assumptions, while the **posterior** reflects observed data, showing how real data influences expenditure likelihood.

```
#Compute prior predictive probability of y>25
y_prior <-rnorm(n_samples, mean= alpha_prior + beta_prior* 68,sd= sigma_prior)
prior_prob<-mean(y_prior> 25)
#Generate posterior predictive samples for x=68
y_posterior <-rnorm(length(alpha_post), mean= alpha_post +beta_post * 68, sd= sigma_post
#Compute posterior predictive probability of y>25
posterior_prob <-mean(y_posterior > 25)
cat("Prior Predictive Probability P(ỹ>25|prior):",prior_prob, "\n")
```

```
Prior Predictive Probability P(ỹ>25|prior): 0.703
```

```
cat("Posterior Predictive ProbabilityP(ỹ>25|posterior):",posterior_prob, "\n")
```

```
Posterior Predictive ProbabilityP(ỹ>25|posterior): 0.06675
```

After incorporating data, the probability of food expenditure exceeding **€25** drops significantly. The **prior probability** was **70.5%**, suggesting a high chance of households spending over **€25 per day**. However, after updating the model with observations, the **posterior probability** falls to **6.05%**, indicating that high expenditures are much less frequent than initially expected.