Heart Attack Analysis: Comparison with Bayesian and Frequentist Methods

Ishaan Bhandari, Kehuan Wang, April Wu, Max Zhang, Wendy Zheng

Objective

- Comparison with Frequentist and Bayesian methods
- **Frequentist**: cumulative link model - logistic regression with binomial family
- **Bayesian**: brms package and rjags with different priors
- Evaluate each method and comparing the interpretation of the results

Data Description

- Heart Attack Classification from Kaggle
- Target Variable: ***output*** (Binary: 0 - less chance of heart attack, 1 - more chance of heart attack)

Multi-level Categorical variables [used deviation coding]:
cp(4), restecg(3), slp(3), thall(4), caa(5)

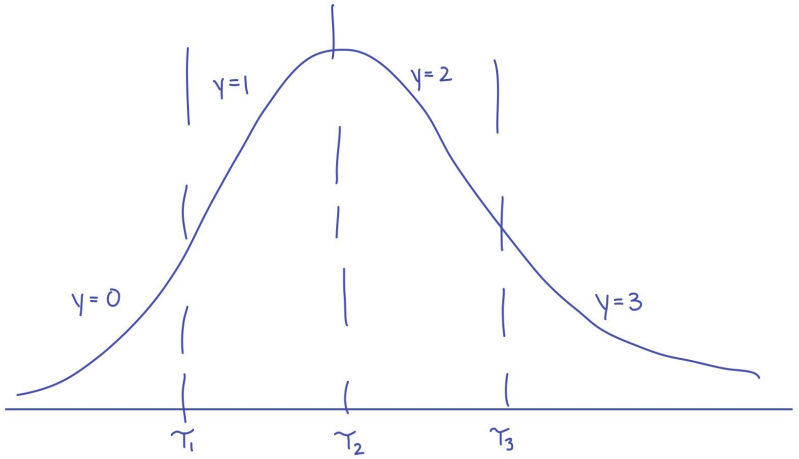
Numerical variables:
age, trtbps, thalachh, oldpeak, chol

Binary variables [used dummy coding]: ***sex, fbs, exng***

- 303 rows and 14 columns
- Randomly split the data into train(70%: 212 observations)and test(30%: 91 observations) sets.
- Use train dataset for Frequentist and Bayesian Models
- Test dataset for evaluation

Cumulative Link Model

- Represents a polytomous variable *y* using a continuous (latent) variable *z*
- Value of *y* is determined by the placement of thresholds and which thresholds *z* falls between



$$y = \begin{cases} 0 & \text{if } z \leq \tau_1 \\ 1 & \text{if } \tau_1 \leq z < \tau_2 \\ 2 & \text{if } \tau_2 \leq z < \tau_3 \\ 3 & \text{if } z \geq \tau_3 \end{cases}$$

Identifiability

- Unique mapping between parameter space and likelihood function

  Given 2 different parameter values ω and ω', the model is identifiable if $\pi_{\square}(\omega) = \pi_{\square}(\omega')$, if and only if $\omega = \omega'$

- Overcome this issue by fixing the value of one of the parameters

Frequentist Model

- Logistic Regression model in R: model_freq
- Glm in R: Fitting Generalized Linear Models
- Binary outcome fits the target variable: Logistic Regression is suitable for handling response variables of binary types.
- The relationship between variables: Through this model, we can explore how different predictive variables affect the outcomes of heart health.

```r
# Logistic Regression model
model_freq = glm(output~., data = train, family = binomial)
summary(model_frq)
```

Bayesian Model

- Bayesian Model in R: model_bayes_normalprior, model_bayes_hourseshoe, jags_model
- Packages use: brms, rjags
- Type: Bernoulli
- Usage: Train and Prediction

Why bayesian model:

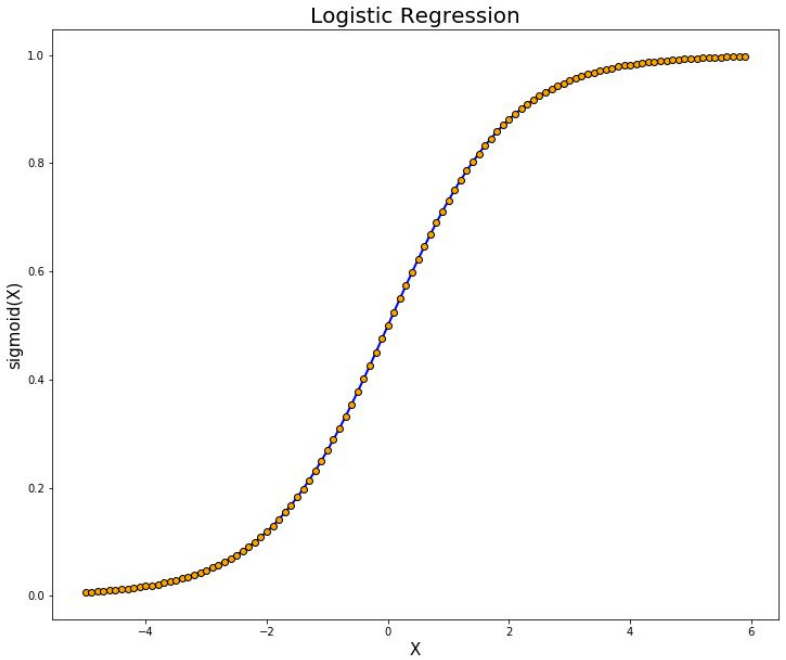
- Handling small data set
- Solving overfitting with prior
- Suitable for predicting model by using probability

Summary: Frequentist vs Bayesian Approach

- Approach
- Flexibility
- Application

Logistic Regression

- Overall Use Cases
- Linear vs Logistic
- Sigmoid Function



Frequentist Result/Interpretation

```
glm(formula = output ~ ., family = binomial, data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.6350  -0.3063   0.1073   0.4510   3.2254

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.749233   3.324481   0.526 0.598772
age          0.033946   0.030708   1.105 0.268964
sex         -2.653409   0.743197  -3.570 0.000357 ***
cp1         -1.670542   0.417365  -4.003 6.27e-05 ***
cp2          0.038045   0.550258   0.069 0.944878
cp3          0.476417   0.406844   1.171 0.241596
trtbps      -0.033793   0.014228  -2.375 0.017547 *
chol        -0.004975   0.005073  -0.981 0.326752
fbs1         0.429690   0.675763   0.636 0.524868
restecg1     0.343526   0.745762   0.461 0.645058
restecg2     0.623701   0.752458   0.829 0.407169
thalachh     0.027708   0.014704   1.884 0.059509 .
exng1       -0.429041   0.550965  -0.779 0.436152
oldpeak     -0.130789   0.262801  -0.498 0.618714
slp1         0.033685   0.645872   0.052 0.958406
slp2        -0.835280   0.415437  -2.011 0.044367 *
caa1         1.138963   0.522056   2.182 0.029133 *
caa2        -0.620371   0.608495  -1.020 0.307957
caa3        -2.632852   0.947894  -2.778 0.005477 **
caa4        -0.642793   0.813689  -0.790 0.429543
thall1      -1.532560   2.114545  -0.725 0.468593
thall2       1.959894   1.025398   1.911 0.055960 .
thall3       0.354996   0.784534   0.452 0.650914
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

- Sex, cp1(angina), caa3(# of vessels) were amongst the most significant
- Accuracy of 0.8571429

```
prediction  0  1
         0 33  6
         1  7 45
[1] 0.8571429
```

Gibbs Sampling

- How it works?
- Applications
- Advantages
  - Flexibility
  - Efficiency
  - Simplicity

Rjags(2008)

- Using BUGS(Bayesian inference Using Gibbs Sampling) language to define complex Bayesian models
- Sampling from these full conditional distributions
- Allow convergence to the target distribution after burn-in
- Collects the samples from the posterior distribution

Modelling(Gibbs Sampling)

```r
# Modify the data preparation for JAGS
data_jags <- list(
  y = as.numeric(levels(train$output))[train$output],
  X = as.matrix(sapply(train[, -which(names(train) == "output")], as.numeric)),
  N = nrow(train),
  K = ncol(train) - 1  # number of predictors
)

# Define initial values function
init_values <- function() {
  list(alpha = 0, beta = rep(0, data_jags$K))
}

# Specify the path to your model file
model_file <- "logistic_regression_model.bug"

# Compile the JAGS model
jags_model <- jags.model(file = model_file, data = data_jags, inits = init_values, n.chains = 4)

# Burn-in period
update(jags_model, n.iter = 500)

# Run the MCMC sampler
samples <- coda.samples(jags_model, variable.names = c("alpha", "beta"), n.iter = 2000)

# Print and summarize the samples
summary(samples)
```

BUGS File

```
model {
    # Priors
    for (i in 1:K) {
        beta[i] ~ dnorm(0, 1/2.5^2)  # Normal prior for coefficients
    }
    alpha ~ dt(0, 1/2.5^2, 1)

    # Likelihood
    for (i in 1:N) {
        logit(p[i]) <- alpha + inprod(beta[], X[i,])
        y[i] ~ dbern(p[i])
    }
}
```

Brms(2015)

- Flexibility in model types
- Stan compiled to C++
- Hamiltonian Monte Carlo (HMC)
- No-U-Turn Sampler (NUTS)

Modelling

```r
# Define priors
priors <- c(
  prior(normal(0, 2.5), class = "b"), # Normal prior for coefficients
  prior(cauchy(0, 2.5), class = "Intercept") # Cauchy prior for the intercept
)

# Build the Bayesian model
model_bayes_normal <- brm(
  formula = output ~ .,
  data = train,
  family = bernoulli(),
  prior = priors,
  chains = 4,
  iter = 2000,
  warmup = 500
)
# Summary
summary(model_bayes_normal)
```

Rjags vs Brms

Their output have similar accuracy and RSS! Which one should we choose?

Brms

Advantages:

- User-friendly interface
- Uses Hamiltonian Monte Carlo (HMC) via Stan, which can handle more complex models and priors.

Disadvantages:

- Models can take longer to run due to the complexity of HMC compared to Gibbs sampling.

Rjags

Advantages:

- Uses Gibbs sampling, which can be efficient for models where full-conditionals are known.
- Flexibility with the BUGS language for model specification.

Disadvantages:

- Less efficient for models with complex or non-conjugate priors.

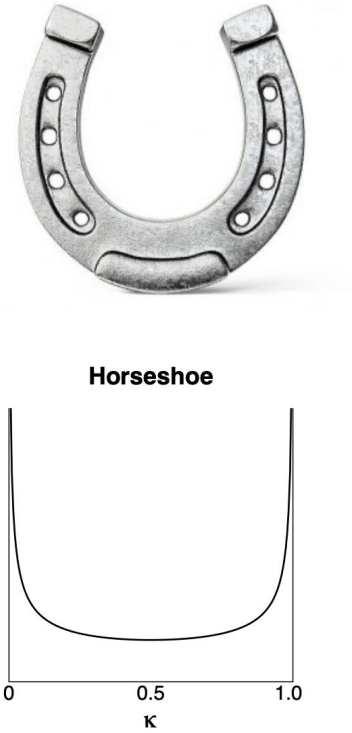
Horseshoe Prior

Goal: Auto variable selection

Mechanism:

$$
\begin{aligned}
(\beta_i|\lambda_i,\tau) &\sim \mathrm{N}(0,\lambda_i^2\tau^2) \\
\lambda_i &\sim \mathrm{C}^+(0,1)\,,
\end{aligned}
$$

- Identify sparsity in High-Dimensional Data
- Shrinkage and Selection
- Automatic Selection
- Balancing Overfitting and Underfitting



Referred: Handling Sparsity via the Horseshoe (2009)

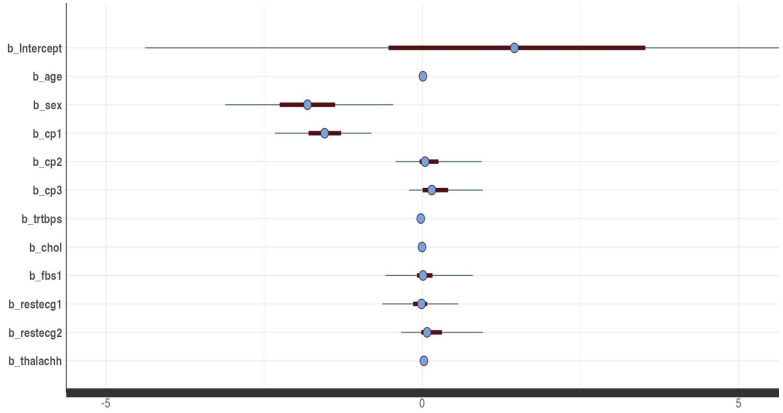
Bayesian Result

Population-Level Effects:

	Estimate	Est.Error	l-95% CI	u-95% CI	Rhat	Bulk_ESS	Tail_ESS
Intercept	1.48	2.98	-4.33	7.26	1.00	3107	3704
age	0.01	0.03	-0.03	0.07	1.00	3141	5727
sex	-1.80	0.67	-3.07	-0.40	1.00	1850	2917
cp1	-1.55	0.38	-2.30	-0.82	1.00	3350	4614
cp2	0.11	0.31	-0.42	0.94	1.01	1253	637
cp3	0.22	0.30	-0.19	0.93	1.00	2228	3202
trtbps	-0.02	0.01	-0.05	0.00	1.00	2833	2543
chol	-0.00	0.00	-0.01	0.00	1.00	3210	5027
fbs1	0.05	0.32	-0.58	0.82	1.00	2113	2575
restecg1	-0.03	0.26	-0.61	0.54	1.00	2712	4111
restecg2	0.17	0.32	-0.31	0.97	1.00	2391	2411
thalachh	0.03	0.01	0.00	0.05	1.00	2121	1667
exng1	-0.23	0.37	-1.19	0.29	1.00	1924	2368
oldpeak	-0.18	0.19	-0.63	0.11	1.00	2492	4373
slp1	-0.14	0.33	-0.95	0.42	1.00	1595	3457
slp2	-0.40	0.31	-1.05	0.07	1.00	2698	3506
caa1	0.84	0.49	-0.02	1.79	1.00	1938	1757
caa2	-0.29	0.41	-1.31	0.26	1.00	2831	3548
caa3	-1.37	0.92	-3.19	0.08	1.01	873	611
caa4	-0.22	0.46	-1.46	0.40	1.00	3986	4617
thall1	0.18	0.53	-0.75	1.50	1.00	1783	2314
thall2	0.61	0.56	-0.18	1.80	1.00	2104	2099
thall3	0.21	0.32	-0.30	0.94	1.00	2028	1704

- RSS of Rjags model: 11.7666030453929
- RSS of bayesian model with normal prior: 12
- RSS of bayesian model with horseshoe prior: 8
- RSS of Frequentist model: 13
- Estimates are the posterior mean of each parameter
- Estimates converged well

Conclusion



- Bayesian model with horseshoe prior performs the best

- 1 (male) is associated with the log odds decrease in the outcome by 1.64 - Female is more likely to get heart attack. Typical chest pain would have a higher impact on the outcome variable compared to other type of chest pains.

Further exploring

- Larger dataset with outcome variables that are ordinal with more than 2 levels
- Explore Gibbs Sampling Algorithm
- Further exploration in brms() and the algorithm it implements

References

- Dataset:https://www.kaggle.com/datasets/rashikrahmanpritom/heart-attack-analysis-prediction-dataset
- Cumulative link model: https://uofi.app.box.com/s/40e8wow49vi3qtbh7ot31z1b8gz81v70
- Horseshoe Prior: https://proceedings.mlr.press/v5/carvalho09a
- Brms: https://www.jstatsoft.org/article/view/v080i01

THANK YOU !