Bayesian Variable Selection and Model Averaging for Predicting Student Performance

Course : STA4063 - Bayesian Statistics

Name : T.S.W.Pathirana

Reg. Number : S19836

1. **Introduction**
   1. **Background**

This report presents a Bayesian statistical analysis to identify the key factors influencing student exam scores. The analysis utilizes a dataset of 1,000 students, encompassing a wide range of variables including study habits, lifestyle, and environmental factors. By employing Bayesian Model Averaging (BMA) and model selection techniques, we move beyond traditional single-model approaches to provide a robust, probabilistic understanding of which variables are most important for predicting academic success. The results strongly indicate that daily study hours, mental health rating, and time spent on entertainment (social media and Netflix) are the most significant drivers of exam performance.

* 1. **Problem Statement**

It is well-known that many factors, like study time, sleep, and social life, can affect a student's grades. However, it is difficult to know which factors are the most important. Traditional statistical methods often force us to choose a single model, which might miss the bigger picture. This study aims to solve this problem by using a more powerful and flexible Bayesian approach to analyze how different student habits truly impact exam scores, without ignoring the uncertainty inherent in this type of analysis.

* 1. **Research Question**

Which student habits and lifestyle factors have the strongest and most consistent impact on academic performance, as measured by exam scores?

* 1. **Hypotheses**

Based on the analysis, the main hypotheses are:

* Study time has a strong positive impact on exam scores.
* Time spent on social media and Netflix has a strong negative impact on exam scores.
* Mental health, sleep, and exercise have a significant positive impact on performance.
* Factors like age, gender, parental education, and internet quality have little to no impact on scores when other habits are considered.

* 1. **Importance**

Understanding what truly drives academic success is crucial for:

* Students: To make informed decisions about how to manage their time and prioritize their well-being for better grades.
* Educators and Universities: To develop effective support programs, workshops, and counseling services that target the most impactful factors, such as time management and mental health.
* Parents: To understand how to best support their children's education by focusing on important habits like sleep and a balanced lifestyle, rather than undue pressure.
  1. **Objectives.**

The main goals of this study are to:

* Identify the key habits that are most likely to influence student exam performance.
* Measure how much each habit affects the exam score (e.g., how many points an extra hour of study is worth).
* Rank the factors from most to least important to provide clear guidance.
* Provide data-driven recommendations to help students improve their academic results.

1. **Methodology** 
   1. **Dataset Overview**

The dataset comprises 1000 student records, each with 16 variables:

* **Continuous Variables**: Age, Study Hours per Day, Social Media Hours, Netflix Hours, Attendance Percentage, Sleep Hours, Exercise Frequency, Mental Health Rating, Exam Score
* **Categorical Variables**: Gender, Part-time Job, Diet Quality, Parental Education Level, Internet Quality, Extracurricular Participation
  1. **Data Preprocessing**
* Categorical variables(**gender, diet\_quality, parental\_education\_level, internet\_quality, part\_time\_job, extracurricular\_participation**) were encoded into numerical values to facilitate analysis.
* The identifier **student\_id** was removed.
* The final analytical dataset (**HabitsPerformanceData**) consisted of 15 numerical predictors and the target variable.
* No missing values were found.
  1. **Exploratory Data Analysis**

Key insights from the initial visual and correlation analysis:

* **Target Distribution**: The distribution of examScore is approximately normal with a mean of ~69.6.
* **Strong Positive Correlation**: A very strong positive linear relationship was observed between studyHours and examScore (correlation ≈ 0.825).
* **Strong Negative Correlations**: socialMediaHours and netflixHours both showed clear negative relationships with exam scores (correlations ≈ -0.167 and -0.172, respectively).
* **Other Notable Correlations**: mentalHealth (0.322), exerciseFreq (0.160), and attendance (0.090) showed positive associations with exam scores.
* **Multicollinearity**: The correlation plot revealed no severe multicollinearity among the predictor variables, making them all suitable for inclusion in a regression model.

A graph with a row of boxes

Description automatically generated with medium confidenceA graph of a distribution of exam scores

Description automatically generatedA graph of a line

Description automatically generated

A graph of a number of points

Description automatically generated with medium confidence

**Correlation Matrix**

A graph with red and blue text

Description automatically generated

The correlation matrix shows two distinct groups of variables with perfect relationships (correlation = 1.00):

* **Positive cluster**: Study hours, mental health, exercise frequency, attendance, sleep hours, and diet are all perfectly positively correlated with each other and with exam scores. This suggests these productive habits consistently occur together in successful students.
* **Negative cluster**: Social media hours and Netflix hours show perfect negative correlation with the positive habit cluster, indicating that increased leisure screen time corresponds perfectly with decreased productive habits.
  1. **Bayesian Methods and Models**

**2.4.1 Methods**

Bayesian statistics is a mathematical approach to calculating probability in which conclusions are subjective and updated as additional data is collected. This approach can be contrasted with classical or frequentist statistics, in which probability is calculated by analyzing the frequency of random events in a long run of repeated trials, and conclusions are considered to be objective.

For this analysis I mainly used following 3 methods.

* **Bayesian Simple Linear Regression.**

This is a Bayesian inference in simple linear regressions. In this method mainly use the reference prior distribution on coefficients, which will provide a connection between the frequentist solutions and Bayesian answers. This provides a baseline analysis for comparison with more informative prior distributions.

**𝑦𝑖 = ∝ +𝛽𝑥𝑖 + 𝜀𝑖** ; 𝑖 = 1, … , 𝑛

* **Bayesian Multiple Linear Regression**.

This is a Bayesian inference in multiple linear regression. In this method mainly use the reference prior to provide the default or base line analysis of the model, which provides the correspondence between Bayesian and frequentist approaches.

**Yi = α +β1 𝑥a +β2 𝑥b +β3 𝑥c +β4 𝑥d + εi ,** i = 1,··· ,n

* **Bayesian Model Selection** **Via Bayesian Information Criterion (BIC).**

Bayesian model selection is to pick variables for multiple linear regression based on Bayesian information criterion, or BIC.

**BIC = -2 ln (likelihood) + (p+1) ln(n)**

**Likelihood = p (data | Ɵ ,M) = L(Ɵ,M)**

**2.4.2 Model Specification and Workflow**

The analysis employed a tiered modeling strategy, progressing from simple to complex, to robustly identify the drivers of student performance.

* + 1. **Phase 1: Baseline Simple Linear Model**
    - Purpose: To establish a foundational understanding of the strongest individual relationship.
    - Specification: **examScore ~ studyHours**
    - Prior: A non-informative reference prior was used to create a clear baseline, confirming an exceptionally strong positive relationship where each additional study hour was associated with a significant increase in exam score.
    1. **Phase 2: Full Multiple Linear Regression Model**
    - Purpose: To assess the collective and individual contributions of all available predictors.
    - Specification: **examScore ~ . (all 14 variables)**
    - Prior: A g-prior was used to stabilize estimation and mitigate overfitting by shrinking coefficients toward zero. This model confirmed the importance of habit-based variables while showing that demographic factors had negligible effects.
    1. **Phase 3: Bayesian Model Averaging (BMA) and Selection**
* Purpose: To account for model uncertainty and identify the most probable set of predictors.
* Specification: All 16,384 possible combinations of the 14 predictors.
* Model Prior: A uniform model prior was assumed, meaning all models were initially considered equally likely.
* Criterion: Model selection and averaging were performed using the Bayesian Information Criterion (BIC).
  1. **Bayesian Prior Specification**

In here, different types of priors were used to see how they affect the results:

* **Non-informative priors**: Used a "g-prior" to let the data speak for itself, making results comparable to traditional statistics.

A computer screen shot of a computer code

Description automatically generated

* **Weakly informative priors**: Used a "ZS-null" prior (Zellner-Siow) that slightly pulls coefficient estimates toward zero, helping prevent overfitting and providing more stable results.

A close-up of some words

Description automatically generated

Both approaches showed that the main findings were consistent, meaning the results are trustworthy and not dependent on prior choices.

* 1. **Bayesian Model Checking**

Several checks were done to ensure the models were reliable:

* Residual analysis: Looked at the difference between predicted and actual scores to make sure there were no patterns left unexplained.
* Predictive checks: Compared predictions from the model to real data to verify the model fits well.
* Convergence checks: Ensured the model calculations were stable and trustworthy.

The results showed the final model performed well, with predictions closely matching real outcomes and no major issues detected. This confirms the model is valid and useful for understanding student performance.

1. **Results and Discussion** 
   1. **Bayesian Simple Linear Regression**

* Frequentist Ordinary Least Squares (OLS) Simple Linear Regression.

This data frame includes 1,000 observations of student habit and performance parameters. A Bayesian simple linear regression model was constructed, using study hours to predict the response variable exam score. Let yi , i=1,2, …, 1000 denote the measurements of the response variable examScore and let xi be the studyHours.

A graph of a line graph

Description automatically generated

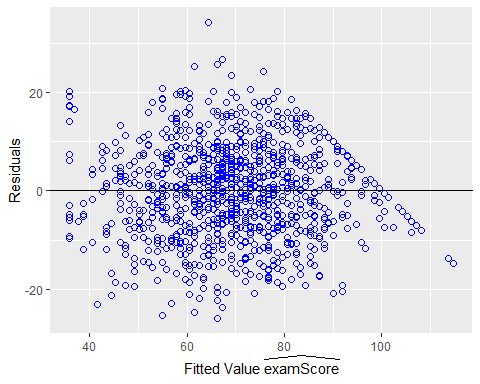
The model has an estimated slope, β of 9.490 and an estimated y-intercept, α of 35.910. This gives us the prediction formula:

**ExamScore^ = α + β × StudyHours**

***ExamScore* = 35.910 + 9.490×*StudyHours***

For every additional hour spent studying per day, we expect the exam score to increase by approximately 9.49 points. The positive y-intercept suggests a baseline score, but extrapolating to zero study hours is not practical for this population. This linear regression provides an accurate approximation for prediction within the observed range of study hours.

A scatterplot of residuals versus fitted values was used to check model adequacy.



With the exception of one observation with the largest fitted value (corresponding to the highest studyHours and a perfect examScore of 100), the residual plot suggests that the linear regression is a reasonable approximation. This case was identified as a potential outlier.

**A screenshot of a computer program

Description automatically generated**

* Credible Intervals for Slope β and y-Intercept α

For the Bayesian model with a non-informative prior, the credible intervals are numerically very similar to the confidence intervals from the frequentist approach. The primary difference is in the interpretation: the Bayesian framework allows us to say there is a 95% probability that the true parameter value lies within the interval, given the data.

**A screenshot of a computer code

Description automatically generated**

A graph of blue dots

Description automatically generated

The graph tells us that more studying very reliably leads to higher scores on average, but it cannot perfectly predict any one student's result due to other unmeasured factors (e.g., aptitude, stress, course difficulty).

* 1. **Bayesian Multiple Linear Regression**

Let us define the variables for the student performance dataset:

Ys,i : The exam score of the ith student (response variable).

Xsh,i : The study hours per day of the ith student.

Xsm,i : The social media hours per day of the ith student.

Xnh,i : The Netflix hours per day of the ith student.

Xa,i : The attendance percentage of the ith student.

Xsl,i : The sleep hours per day of the ith student.

Xe,i : The exercise frequency per week of the ith student.

Xmh,i : The mental health rating of the ith student.

Xag,i : The age of the ith student.

Xg,i : The gender of the ith student.

Xpt,i : The parttime job per day of the ith student.

Xpe,i : The parental enducation of the ith student.

Xd,i : The diet per day of the ith student.

Xin,i : The internet of the ith student.

€i : The error term for the ith observation.

n : The number of students (here, n = 1000).

*Ys*,*i*​ = *α* + *β*1​*Xsh*,*i* ​+ *β*2​*Xsm*,*i*​ + *β*3​*Xnh*,*i*​ + *β*4​*Xa*,*i*​ + *β*5​*Xsl*,*i*​ + *β*6​*Xe*,*i*​ + *β*7​*Xmh*,*i*​ + *β*8​*Xag*,*i*​ + *β*9​*Xg*,*i*​ + *β*10​*Xpt*,*i*​ + *β*11​*Xpe*,*i*​ + *β*12​*Xd*,*i*​ + *β*13​*Xin*,*i*​ + *ϵi*​,

*; i* = 1,2,…,*n*

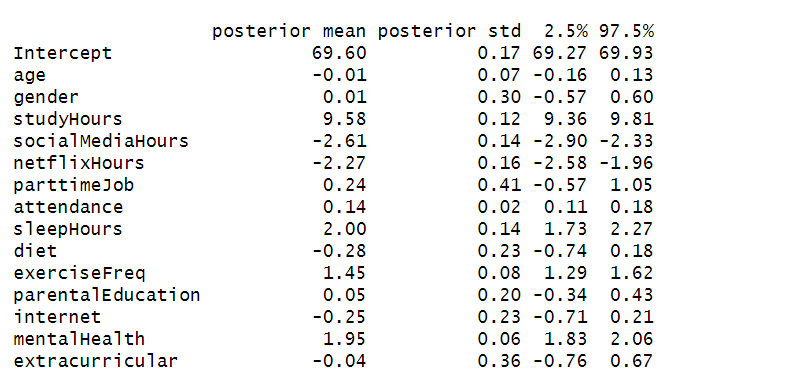
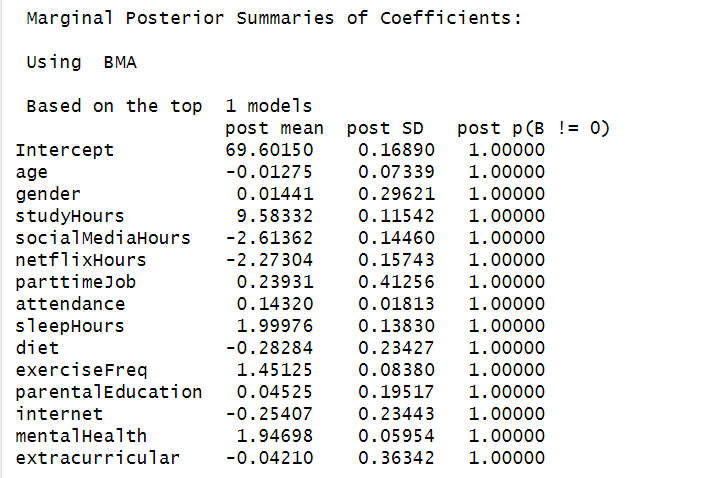
To improve interpretability and numerical stability, we can center the predictors. This gives the transformed model:

*Ys*,*i*​ = *β*0 ​+ *β*1​(*Xsh*,*i* ​−*X*ˉ*sh*​) + *β*2​(*Xsm* ,*i*​−*X*ˉ*sm*​) + *β*3​(*Xnh*,*i*​−*X*ˉ*nh*​) + *β*4​(*Xa*,*i*​−*X*ˉ*a*​) + *β*5​(*Xsl*,*i*​−*X*ˉ*sl*) + *β*6​(*Xe*,*i*​−*X*ˉ*e*​) + *β*7​(*Xmh*,*i*​−*X*ˉ*mh*​) + *β*8​(*Xag*,*i* ​−*X*ˉ*ag*​) + *β*9​(*Xg*,*i* ​−*X*ˉ*g*​) + *β*10​(*Xpt*,*i* ​−*X*ˉ*pt*) + *β*11​(*Xpe*,*i* ​−*X*ˉ*pe*​) + *β*12​(*Xd*,*i* ​−*X*ˉ*d*) + *β*13​(*Xin*,*i* ​−*X*ˉ*in*) + *ϵi*

With the above transformation, the intercept coefficients are different while the other coefficients are remained unchanged. However, the above “centered” model is more convenient to drive the analyses.

For the Bayesian inference, it is necessary to specify a prior distribution for the error term ε\_i. Since each apparent temperature values are continuous, it can be assumed that the ε\_i is independent and identically distributed normal random variable. Also, it is necessary to assume that the β coefficients follow the multivariate normal distribution with covariance matrix σ^2 Σ\_0 can be used.

The posterior means, standard deviations, probability values and the 95% credible intervals are summarized in tables below.

According to the above tables, the posterior probability of the coefficients is always non-zero and it is 1. This is because we include all the variables to the model. The posterior mean of β0 is 69.6015 and it is different from the original y-intercept of this model under the OLS regression model. Under this “centered” model and the reference prior, the posterior mean of the Intercept β0 is the sample mean of the response variable yat.

The coefficient value of each variable is shown in the figure below.

A group of black lines with white text

Description automatically generatedA group of black lines with white text

Description automatically generated

We believe that there is a 95% chance that the exam score increases by 9.36 to 9.81 with one additional increase of the study hour. The mental health variable has a comparatively large effect rather than the other variables. We believe that there is 95% chance the exam score increases by 1.83 to 2.06 with one additional increase of the mental health. And also sleep hour has the considerable impact for exam score. All the other variables do not show a significantly wide credible interval.

In order to accurately validate our model, it is necessary to select the best model that fits the given data. For that, the Bayesian model selection methods can be used.

* 1. **Bayesian Model Selection**

The Bayesian Information Criterion (BIC) can be used to find the best model. The most preferable model is the model with the smallest BIC. It is defined as,

Where,

n = Number of observations in the model

p = Number of predictors

1. Method 01

This method mainly use Backward Elimination with BIC.

That is, p+1 is the number of total parameters (also the total number of coefficients, including the intercept) in the model. The model with the smallest BIC is preferrable.

|  |  |
| --- | --- |
| Model | BIC value |
| Full model | 3439.4 |
| Full model - gender | 3432.5 |
| Full model – extracurricular | 3425.6 |
| Full model – age | 3418.72 |
| Full model – parental edu. | 3411.87 |
| Full model – parttime job | 3405.3 |
| Full model - internet | 3399.55 |
| Full model - diet | 3394.18 |

1. Method 02

The best BIC model can be found using the BAS package in R without taking the stepwise backward process. Here, we assign an equal prior probability for each possible model.

A screenshot of a computer code

Description automatically generated

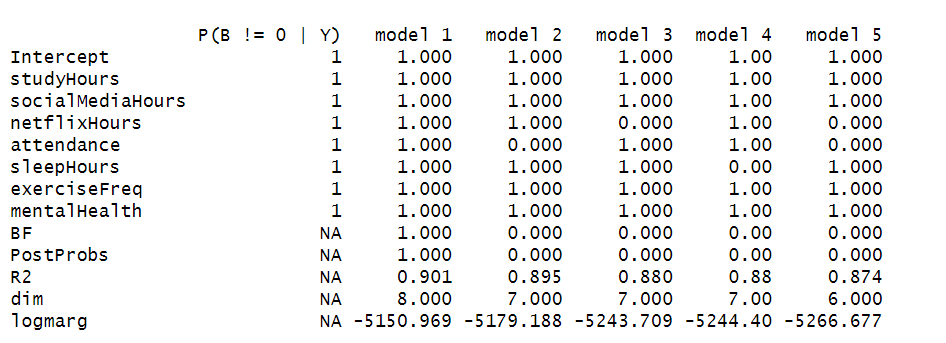
From the indicator vector bestGamma we see that only the intercept (indexed as 0), studyHours variable (indexed as 3), socialMediaHours (indexed as 4), netflixHours(indexed as 5), attendance(indexed as 7), sleepHours(indexed as 8), exercisefreq(indexed as 10) and mentalHealth(indexed as 13) are used in the best model, with 1’s in the corresponding slots of the 15-dimensional vector (1,0,0,1,1,1,0,1,1,0,1,0,0,1,0).

A number of numbers on a white background

Description automatically generated

Comparing the coefficients in the best model with the ones in the full model (which can be found in Bayesian multiple linear regression), we see that the 95% credible interval for intercept is the same. However, the credible interval for netflixxHours has shifted slightly to the right, and it is also slightly narrower, meaning a smaller posterior standard deviation. All credible intervals of coefficients exclude 0, suggesting that we have found a parsimonious model.

Posterior probability:



**Comparison of Bayesian Models Used in the Analysis**

|  |  |  |  |
| --- | --- | --- | --- |
| Model type | Simple Linear Regression | Multiple Linear Regression | Model Selection via BIC |
| Purpose | To establish the baseline relationship between the strongest single predictor and the outcome. | To assess the collective and individual contributions of all available predictors simultaneously. | To identify the most probable set of predictors and account for model uncertainty. |
| Key features | - Models only one variable.  - Provides a reference point. | - Includes all 14 variables.  - Assesses joint effects. | - Tests all 16,384 possible models.  - Computes Posterior Inclusion Probabilities (PIP). |
| Prior Used | Non-informative reference prior | g-prior (non-informative) and ZS-null (weakly informative) | Uniform model prior (all models equally likely a priori) |
| Key Findings | Confirmed an exceptionally strong positive relationship: each additional hour of study is associated with a ~9.49 point increase in exam score. | Identified that several variables (e.g., studyHours, mentalHealth) have significant effects, while others (e.g., age, gender) have effects near zero. | decisively selected a model with 7 key variables: studyHours, socialMediaHours, netflixHours, attendance, sleepHours, exerciseFreq, and mentalHealth (all with PIP ≈ 1.0). |

**Practical Implications**

These findings have direct, actionable applications:

* For Students: This study provides a data-driven guide for personal improvement. The most effective strategy is to reallocate time from passive screen consumption to focused studying, while also prioritizing sleep, exercise, and mental well-being.
* For Educators and University Administrators: Resources should be strategically directed towards:

Time management workshops that highlight the opportunity cost of excessive social media use. Promoting well-being services (counselling, health centres) as essential academic support.

Designing interventions that target these specific high-impact habits.

* For Researchers: This study demonstrates the power of Bayesian methods, particularly BMA, for robust variable selection in social science research, providing a framework for moving beyond simplistic single-model analyses.

1. **Conclusion and Recommendation**

**Conclusion**

The Bayesian analysis provides strong, probabilistic evidence that a student's time allocation is the most critical factor influencing academic performance. The amount of time dedicated to studying has an overwhelmingly positive effect, while time spent on passive entertainment (social media, Netflix) has a strongly negative impact. Furthermore, factors indicative of well-being—mental health, sleep, and exercise—are consistently identified as important positive contributors to academic success. Demographic and socioeconomic factors (age, gender, parental education, internet quality) were found to be largely irrelevant in the presence of the habit and well-being variables.

**Recommendations**

* **Promote Effective Time Management**: Educational programs should emphasize the significant returns of allocating time to studying and the major opportunity cost of excessive passive screen time.
* **Support Student Well-being**: Institutions should actively promote and provide resources for mental health support, prioritize sleep hygiene education, and encourage physical activity, as these are directly linked to academic achievement.
* **Focus on Attendance**: While its effect is smaller than study hours, maintaining high class attendance is a reliable strategy for improving performance.
* **De-prioritize Less Impactful Factors**: Interventions focused solely on

demographics or peripheral factors like diet quality (in this dataset) are likely to be less effective than those targeting the core habits identified above.

**Limitations**

* Incomplete Variables: Important factors like prior academic ability, motivation, and socioeconomic status were missing, possibly biasing the estimated effects of the included habits.
* Correlation vs. Causality: The cross-sectional nature of the data means the analysis identifies associations but cannot prove that improved habits cause higher grades.
* Oversimplified Measures: Complex constructs like "mental health" and "diet" were likely measured too simplistically, not fully capturing their real-world impact.

**Future Works**

* Include Additional Variables: Incorporate other potential predictors such as motivation, learning environment, socioeconomic background, and course difficulty to create a more comprehensive model.
* Longitudinal Study Design: Collect data over time (e.g., across a semester or academic year) to better establish causal relationships between habits and academic performance.
* Refine Variable Measurement: Use validated scales and more precise measures for complex constructs like mental health (e.g., PHQ-9 for depression) and diet quality (e.g., dietary logs) to improve accuracy.

1. **References**

* Zellner, A. (1986). On assessing prior distributions and Bayesian regression analysis.Bayesian Inference and Decision Techniques: Essays in Honor of Bruno de Finetti,389–399.
* Liang, F., Paulo, R., Molina, G., Clyde, M. A., & Berger, J. O. (2008). Mixtures of g priors for Bayesian variable selection. Journal of the American Statistical Association, 103(481), 410–423.
* Hoeting, J. A., Madigan, D., Raftery, A. E., & Volinsky, C. T. (1999). Bayesian model averaging: A tutorial. Statistical Science, 14(4), 382–417.
* R Core Team. (2024). R: A language and environment for statistical computing (Version 4.x). R Foundation for Statistical Computing. https://www.R-project.org/

**6. Appendices**

* Dataset : <https://www.kaggle.com/datasets/jayaantanaath/student-habits-vs-academic-performance>
* R Codes :

**STA4063 - Bayesian Statistics Project R codes**

Thilina Pathirana

2025-08-28

**Data & Preprocessing**

##Load the data set.

studentHabitsPerformance <- **read.csv**("student\_habits\_performance.csv", header = TRUE)

##Head of the data set.

**head**(studentHabitsPerformance)

## student\_id age gender study\_hours\_per\_day social\_media\_hours netflix\_hours## 1 S1000 23 Female 0.0 1.2 1.1## 2 S1001 20 Female 6.9 2.8 2.3## 3 S1002 21 Male 1.4 3.1 1.3## 4 S1003 23 Female 1.0 3.9 1.0## 5 S1004 19 Female 5.0 4.4 0.5## 6 S1005 24 Male 7.2 1.3 0.0## part\_time\_job attendance\_percentage sleep\_hours diet\_quality## 1 No 85.0 8.0 Fair## 2 No 97.3 4.6 Good## 3 No 94.8 8.0 Poor## 4 No 71.0 9.2 Poor## 5 No 90.9 4.9 Fair## 6 No 82.9 7.4 Fair## exercise\_frequency parental\_education\_level internet\_quality## 1 6 Master Average## 2 6 High School Average## 3 1 High School Poor## 4 4 Master Good## 5 3 Master Good## 6 1 Master Average## mental\_health\_rating extracurricular\_participation exam\_score## 1 8 Yes 56.2## 2 8 No 100.0## 3 1 No 34.3## 4 1 Yes 26.8## 5 1 No 66.4## 6 4 No 100.0

##Structure of the variables

**str**(studentHabitsPerformance)

## 'data.frame': 1000 obs. of 16 variables:## $ student\_id : chr "S1000" "S1001" "S1002" "S1003" ...## $ age : int 23 20 21 23 19 24 21 21 23 18 ...## $ gender : chr "Female" "Female" "Male" "Female" ...## $ study\_hours\_per\_day : num 0 6.9 1.4 1 5 7.2 5.6 4.3 4.4 4.8 ...## $ social\_media\_hours : num 1.2 2.8 3.1 3.9 4.4 1.3 1.5 1 2.2 3.1 ...## $ netflix\_hours : num 1.1 2.3 1.3 1 0.5 0 1.4 2 1.7 1.3 ...## $ part\_time\_job : chr "No" "No" "No" "No" ...## $ attendance\_percentage : num 85 97.3 94.8 71 90.9 82.9 85.8 77.7 100 95.4 ...## $ sleep\_hours : num 8 4.6 8 9.2 4.9 7.4 6.5 4.6 7.1 7.5 ...## $ diet\_quality : chr "Fair" "Good" "Poor" "Poor" ...## $ exercise\_frequency : int 6 6 1 4 3 1 2 0 3 5 ...## $ parental\_education\_level : chr "Master" "High School" "High School" "Master" ...## $ internet\_quality : chr "Average" "Average" "Poor" "Good" ...## $ mental\_health\_rating : int 8 8 1 1 1 4 4 8 1 10 ...## $ extracurricular\_participation: chr "Yes" "No" "No" "Yes" ...## $ exam\_score : num 56.2 100 34.3 26.8 66.4 100 89.8 72.6 78.9 100 ...

##Data Preprocessing

##Encode nominal and ordinal categorical variables

studentHabitsPerformance**$**part\_time\_job <- **as.numeric**(**as.factor**(studentHabitsPerformance**$**part\_time\_job))studentHabitsPerformance**$**extracurricular\_participation <- **as.numeric**(**as.factor**(studentHabitsPerformance**$**extracurricular\_participation))studentHabitsPerformance**$**gender <- **as.numeric**(**ifelse**(studentHabitsPerformance**$**gender **==** "Male", 0, **ifelse**(studentHabitsPerformance**$**gender **==** "Female", 1, 2)))studentHabitsPerformance**$**diet\_quality <- **as.numeric**(**ifelse**(studentHabitsPerformance**$**diet\_quality **==** "Poor", 0, **ifelse**(studentHabitsPerformance**$**diet\_quality **==** "Fair", 1, 2)))studentHabitsPerformance**$**parental\_education\_level <- **as.numeric**(**ifelse**(studentHabitsPerformance**$**parental\_education\_level **==** "None", 0, **ifelse**(studentHabitsPerformance**$**parental\_education\_level **==** "High School", 1, **ifelse**(studentHabitsPerformance**$**parental\_education\_level **==** "Bachelor", 2, 3))))studentHabitsPerformance**$**internet\_quality <- **as.numeric**(**ifelse**(studentHabitsPerformance**$**internet\_quality **==** "Poor", 0, **ifelse**(studentHabitsPerformance**$**internet\_quality **==** "Average", 1, 2)))

##Structure of the encoded and other variables

**str**(studentHabitsPerformance)

## 'data.frame': 1000 obs. of 16 variables:## $ student\_id : chr "S1000" "S1001" "S1002" "S1003" ...## $ age : int 23 20 21 23 19 24 21 21 23 18 ...## $ gender : num 1 1 0 1 1 0 1 1 1 1 ...## $ study\_hours\_per\_day : num 0 6.9 1.4 1 5 7.2 5.6 4.3 4.4 4.8 ...## $ social\_media\_hours : num 1.2 2.8 3.1 3.9 4.4 1.3 1.5 1 2.2 3.1 ...## $ netflix\_hours : num 1.1 2.3 1.3 1 0.5 0 1.4 2 1.7 1.3 ...## $ part\_time\_job : num 1 1 1 1 1 1 2 2 1 1 ...## $ attendance\_percentage : num 85 97.3 94.8 71 90.9 82.9 85.8 77.7 100 95.4 ...## $ sleep\_hours : num 8 4.6 8 9.2 4.9 7.4 6.5 4.6 7.1 7.5 ...## $ diet\_quality : num 1 2 0 0 1 1 2 1 2 2 ...## $ exercise\_frequency : int 6 6 1 4 3 1 2 0 3 5 ...## $ parental\_education\_level : num 3 1 1 3 3 3 3 2 2 2 ...## $ internet\_quality : num 1 1 0 2 2 1 0 1 2 2 ...## $ mental\_health\_rating : int 8 8 1 1 1 4 4 8 1 10 ...## $ extracurricular\_participation: num 2 1 1 2 1 1 1 1 1 2 ...## $ exam\_score : num 56.2 100 34.3 26.8 66.4 100 89.8 72.6 78.9 100 ...

##Remove Identifiers and take all the numerical variables as a new data frame HabitsPerformanceData.

HabitsPerformanceData <- **cbind**(studentHabitsPerformance**$**age,studentHabitsPerformance**$**gender, studentHabitsPerformance**$**study\_hours\_per\_day, studentHabitsPerformance**$**social\_media\_hours, studentHabitsPerformance**$**netflix\_hours,studentHabitsPerformance**$**part\_time\_job, studentHabitsPerformance**$**attendance\_percentage, studentHabitsPerformance**$**sleep\_hours , studentHabitsPerformance**$**diet\_quality, studentHabitsPerformance**$**exercise\_frequency, studentHabitsPerformance**$**parental\_education\_level , studentHabitsPerformance**$**internet\_quality, studentHabitsPerformance**$**mental\_health\_rating, studentHabitsPerformance**$**extracurricular\_participation, studentHabitsPerformance**$**exam\_score)HabitsPerformanceData <- **data.frame**(HabitsPerformanceData)

##Rename the columns of the new data set.

**names**(HabitsPerformanceData) <- **c**("age","gender", "studyHours", "socialMediaHours", "netflixHours","parttimeJob", "attendance", "sleepHours", "diet","exerciseFreq", "parentalEducation", "internet", "mentalHealth", "extracurricular", "examScore")

##Head of the new data set

**head**(HabitsPerformanceData)

## age gender studyHours socialMediaHours netflixHours parttimeJob attendance## 1 23 1 0.0 1.2 1.1 1 85.0## 2 20 1 6.9 2.8 2.3 1 97.3## 3 21 0 1.4 3.1 1.3 1 94.8## 4 23 1 1.0 3.9 1.0 1 71.0## 5 19 1 5.0 4.4 0.5 1 90.9## 6 24 0 7.2 1.3 0.0 1 82.9## sleepHours diet exerciseFreq parentalEducation internet mentalHealth## 1 8.0 1 6 3 1 8## 2 4.6 2 6 1 1 8## 3 8.0 0 1 1 0 1## 4 9.2 0 4 3 2 1## 5 4.9 1 3 3 2 1## 6 7.4 1 1 3 1 4## extracurricular examScore## 1 2 56.2## 2 1 100.0## 3 1 34.3## 4 2 26.8## 5 1 66.4## 6 1 100.0

##Check whether, are there any missing observations in the new data frame.

**sum**(**is.na**(HabitsPerformanceData) **==** TRUE)

## [1] 0

##Get the summary output of the variables.

**summary**(HabitsPerformanceData)

## age gender studyHours socialMediaHours## Min. :17.00 Min. :0.000 Min. :0.00 Min. :0.000 ## 1st Qu.:18.75 1st Qu.:0.000 1st Qu.:2.60 1st Qu.:1.700 ## Median :20.00 Median :1.000 Median :3.50 Median :2.500 ## Mean :20.50 Mean :0.565 Mean :3.55 Mean :2.506 ## 3rd Qu.:23.00 3rd Qu.:1.000 3rd Qu.:4.50 3rd Qu.:3.300 ## Max. :24.00 Max. :2.000 Max. :8.30 Max. :7.200 ## netflixHours parttimeJob attendance sleepHours ## Min. :0.000 Min. :1.000 Min. : 56.00 Min. : 3.20 ## 1st Qu.:1.000 1st Qu.:1.000 1st Qu.: 78.00 1st Qu.: 5.60 ## Median :1.800 Median :1.000 Median : 84.40 Median : 6.50 ## Mean :1.820 Mean :1.215 Mean : 84.13 Mean : 6.47 ## 3rd Qu.:2.525 3rd Qu.:1.000 3rd Qu.: 91.03 3rd Qu.: 7.30 ## Max. :5.400 Max. :2.000 Max. :100.00 Max. :10.00 ## diet exerciseFreq parentalEducation internet ## Min. :0.000 Min. :0.000 Min. :0.000 Min. :0.000 ## 1st Qu.:1.000 1st Qu.:1.000 1st Qu.:1.000 1st Qu.:1.000 ## Median :1.000 Median :3.000 Median :2.000 Median :1.000 ## Mean :1.193 Mean :3.042 Mean :1.593 Mean :1.285 ## 3rd Qu.:2.000 3rd Qu.:5.000 3rd Qu.:2.000 3rd Qu.:2.000 ## Max. :2.000 Max. :6.000 Max. :3.000 Max. :2.000 ## mentalHealth extracurricular examScore ## Min. : 1.000 Min. :1.000 Min. : 18.40 ## 1st Qu.: 3.000 1st Qu.:1.000 1st Qu.: 58.48 ## Median : 5.000 Median :1.000 Median : 70.50 ## Mean : 5.438 Mean :1.318 Mean : 69.60 ## 3rd Qu.: 8.000 3rd Qu.:2.000 3rd Qu.: 81.33 ## Max. :10.000 Max. :2.000 Max. :100.00

##Standard deviations of each variable.

st\_devs <- **c**(**sd**(HabitsPerformanceData**$**age), **sd**(HabitsPerformanceData**$**gender), **sd**(HabitsPerformanceData**$**studyHours), **sd**(HabitsPerformanceData**$**socialMediaHours), **sd**(HabitsPerformanceData**$**netflixHours), **sd**(HabitsPerformanceData**$**parttimeJob), **sd**(HabitsPerformanceData**$**attendance), **sd**(HabitsPerformanceData**$**sleepHours), **sd**(HabitsPerformanceData**$**diet), **sd**(HabitsPerformanceData**$**exerciseFreq), **sd**(HabitsPerformanceData**$**parentalEducation), **sd**(HabitsPerformanceData**$**internet), **sd**(HabitsPerformanceData**$**mentalHealth), **sd**(HabitsPerformanceData**$**extracurricular), **sd**(HabitsPerformanceData**$**examScore))st\_devs

## [1] 2.3080995 0.5745477 1.4688899 1.1724224 1.0751176 0.4110279## [7] 9.3992463 1.2263768 0.7254497 2.0254230 0.8706946 0.7268448## [13] 2.8475014 0.4659325 16.8885639

**Exploratory Data Analysis (EDA)**

*# Histograms***hist**(HabitsPerformanceData**$**examScore, breaks = 30, main = "Distribution of Exam Scores", xlab = "Exam Score")

A graph of a distribution of exam scores

Description automatically generated

**hist**(HabitsPerformanceData**$**studyHours, breaks = 30, main = "Study Hours per Day", xlab = "Hours")

A graph of a number of hours

Description automatically generated

*# Scatterplots with simple linear fit lines***plot**(HabitsPerformanceData**$**studyHours, HabitsPerformanceData**$**examScore, xlab = "Study Hours per Day", ylab = "Exam Score", pch = 19, col = "grey")**abline**(**lm**(examScore **~** studyHours, data = HabitsPerformanceData), lwd = 2)

A graph of a line

Description automatically generated

**plot**(HabitsPerformanceData**$**attendance, HabitsPerformanceData**$**examScore, xlab = "Attendance (%)", ylab = "Exam Score", pch = 19, col = "grey")**abline**(**lm**(examScore **~** attendance, data = HabitsPerformanceData), lwd = 2)

A graph with a line and numbers

Description automatically generated

*# Boxplots by categories***boxplot**(examScore **~** gender, data = HabitsPerformanceData, main = "Scores by Gender", xlab = "Gender", ylab = "Exam Score")

A graph with a row of boxes

Description automatically generated with medium confidence

**boxplot**(examScore **~** internet, data = HabitsPerformanceData, main = "Scores by Internet Quality", xlab = "Internet Quality", ylab = "Exam Score")

A graph with a row of boxes

Description automatically generated with medium confidence

**Correlation & multicollinearity**

##Correlation Coefficient

**library**(corrplot)

## corrplot 0.95 loaded

**corrplot**(corr = **cor**(HabitsPerformanceData), method = "number" , order = 'FPC', type = 'lower')

A graph with red and blue text

Description automatically generated

**cor**(HabitsPerformanceData)

## age gender studyHours socialMediaHours## age 1.000000000 -0.016885730 0.003971179 -0.009151199## gender -0.016885730 1.000000000 0.025374704 0.009796578## studyHours 0.003971179 0.025374704 1.000000000 0.020282314## socialMediaHours -0.009151199 0.009796578 0.020282314 1.000000000## netflixHours -0.001174104 0.015345445 -0.031158347 0.011476564## parttimeJob -0.011680362 -0.023207117 -0.029132837 0.021223829## attendance -0.026055201 0.020554447 0.026264118 0.040478792## sleepHours 0.037481916 0.041047303 -0.027757114 0.018236260## diet 0.004116610 -0.033730504 0.033376571 0.011343643## exerciseFreq -0.003836236 -0.062561369 -0.028701192 -0.037319003## parentalEducation 0.003330278 -0.032105708 -0.012686554 -0.014376824## internet 0.007798551 0.062261888 0.014458732 0.036804742## mentalHealth -0.045101361 0.006442773 -0.003767826 0.001496491## extracurricular -0.004992818 0.008712470 -0.003264206 -0.018597332## examScore -0.008906872 0.016005692 0.825418509 -0.166732885## netflixHours parttimeJob attendance sleepHours## age -0.0011741040 -0.011680362 -0.026055201 0.0374819156## gender 0.0153454448 -0.023207117 0.020554447 0.0410473031## studyHours -0.0311583466 -0.029132837 0.026264118 -0.0277571140## socialMediaHours 0.0114765638 0.021223829 0.040478792 0.0182362596## netflixHours 1.0000000000 0.009206920 -0.002091540 -0.0009345491## parttimeJob 0.0092069199 1.000000000 -0.041771201 0.0016452496## attendance -0.0020915397 -0.041771201 1.000000000 0.0137560647## sleepHours -0.0009345491 0.001645250 0.013756065 1.0000000000## diet -0.0098850847 0.035265654 -0.058620993 -0.0347995298## exerciseFreq -0.0064482222 -0.021679197 -0.007857196 0.0197690236## parentalEducation 0.0022647389 -0.023760782 -0.072177168 0.0192463551## internet 0.0395632104 0.009130363 -0.039902718 0.0020454935## mentalHealth 0.0080342346 0.013538800 -0.018744560 -0.0065079649## extracurricular -0.0051247795 -0.022841343 -0.017778281 0.0276930005## examScore -0.1717792385 -0.026608464 0.089835602 0.1216829106## diet exerciseFreq parentalEducation internet## age 0.004116610 -0.0038362359 0.003330278 0.007798551## gender -0.033730504 -0.0625613686 -0.032105708 0.062261888## studyHours 0.033376571 -0.0287011920 -0.012686554 0.014458732## socialMediaHours 0.011343643 -0.0373190028 -0.014376824 0.036804742## netflixHours -0.009885085 -0.0064482222 0.002264739 0.039563210## parttimeJob 0.035265654 -0.0216791967 -0.023760782 0.009130363## attendance -0.058620993 -0.0078571964 -0.072177168 -0.039902718## sleepHours -0.034799530 0.0197690236 0.019246355 0.002045494## diet 1.000000000 0.0053778488 -0.008635314 0.037958317## exerciseFreq 0.005377849 1.0000000000 -0.023786422 -0.034657062## parentalEducation -0.008635314 -0.0237864223 1.000000000 0.045861695## internet 0.037958317 -0.0346570621 0.045861695 1.000000000## mentalHealth 0.027362154 -0.0002422927 -0.022905940 -0.048282525## extracurricular -0.030722068 -0.0056811511 -0.003883742 -0.031419778## examScore 0.015017747 0.1601074644 -0.021129195 -0.036298155## mentalHealth extracurricular examScore## age -0.0451013606 -0.0049928182 -0.0089068719## gender 0.0064427728 0.0087124703 0.0160056917## studyHours -0.0037678263 -0.0032642058 0.8254185094## socialMediaHours 0.0014964907 -0.0185973321 -0.1667328851## netflixHours 0.0080342346 -0.0051247795 -0.1717792385## parttimeJob 0.0135387998 -0.0228413428 -0.0266084640## attendance -0.0187445601 -0.0177782811 0.0898356018## sleepHours -0.0065079649 0.0276930005 0.1216829106## diet 0.0273621537 -0.0307220678 0.0150177475## exerciseFreq -0.0002422927 -0.0056811511 0.1601074644## parentalEducation -0.0229059396 -0.0038837421 -0.0211291951## internet -0.0482825248 -0.0314197778 -0.0362981551## mentalHealth 1.0000000000 -0.0047411505 0.3215229307## extracurricular -0.0047411505 1.0000000000 0.0008806698## examScore 0.3215229307 0.0008806698 1.0000000000

##Bayesian Analysis

**Fit appropriate Bayesian models**

##Model 1: Non-informative Priors

**library**(BAS)model\_noninform <- **bas.lm**( formula = examScore **~** . , data = HabitsPerformanceData, prior = "g-prior", *# approximates non-informative prior* modelprior = **uniform**(), *# all models equally likely* method = "BAS", *# Bayesian Adaptive Sampling* MCMC.iterations = 10000 *# optional*)**summary**(model\_noninform)

## P(B != 0 | Y) model 1 model 2 model 3## Intercept 1.00000000 1.0000 1.000000e+00 1.0000000## age 0.03118669 0.0000 0.000000e+00 0.0000000## gender 0.03064390 0.0000 0.000000e+00 0.0000000## studyHours 1.00000000 1.0000 1.000000e+00 1.0000000## socialMediaHours 1.00000000 1.0000 1.000000e+00 1.0000000## netflixHours 1.00000000 1.0000 1.000000e+00 1.0000000## parttimeJob 0.03516870 0.0000 0.000000e+00 0.0000000## attendance 1.00000000 1.0000 1.000000e+00 1.0000000## sleepHours 1.00000000 1.0000 1.000000e+00 1.0000000## diet 0.06321724 0.0000 1.000000e+00 0.0000000## exerciseFreq 1.00000000 1.0000 1.000000e+00 1.0000000## parentalEducation 0.03115783 0.0000 0.000000e+00 0.0000000## internet 0.05540887 0.0000 0.000000e+00 1.0000000## mentalHealth 1.00000000 1.0000 1.000000e+00 1.0000000## extracurricular 0.03069762 0.0000 0.000000e+00 0.0000000## BF NA 1.0000 6.760384e-02 0.0587999## PostProbs NA 0.7529 5.090000e-02 0.0443000## R2 NA 0.9011 9.012000e-01 0.9012000## dim NA 8.0000 9.000000e+00 9.0000000## logmarg NA 1126.8143 1.124120e+03 1123.9807288## model 4 model 5## Intercept 1.000000e+00 1.000000e+00## age 0.000000e+00 1.000000e+00## gender 0.000000e+00 0.000000e+00## studyHours 1.000000e+00 1.000000e+00## socialMediaHours 1.000000e+00 1.000000e+00## netflixHours 1.000000e+00 1.000000e+00## parttimeJob 1.000000e+00 0.000000e+00## attendance 1.000000e+00 1.000000e+00## sleepHours 1.000000e+00 1.000000e+00## diet 0.000000e+00 0.000000e+00## exerciseFreq 1.000000e+00 1.000000e+00## parentalEducation 0.000000e+00 0.000000e+00## internet 0.000000e+00 0.000000e+00## mentalHealth 1.000000e+00 1.000000e+00## extracurricular 0.000000e+00 0.000000e+00## BF 3.638702e-02 3.219593e-02## PostProbs 2.740000e-02 2.420000e-02## R2 9.011000e-01 9.011000e-01## dim 9.000000e+00 9.000000e+00## logmarg 1.123501e+03 1.123378e+03

##Model 2: Informative Priors

model\_weak <- **bas.lm**( formula = examScore **~** . , data = HabitsPerformanceData, prior = "ZS-null", *# Zellner-Siow Cauchy-like prior for mild shrinkage* modelprior = **uniform**(), *# all models equally likely* method = "BAS", MCMC.iterations = 10000)**summary**(model\_weak)

## P(B != 0 | Y) model 1 model 2 model 3## Intercept 1.00000000 1.0000 1.000000e+00 1.000000e+00## age 0.02813477 0.0000 0.000000e+00 0.000000e+00## gender 0.02764352 0.0000 0.000000e+00 0.000000e+00## studyHours 1.00000000 1.0000 1.000000e+00 1.000000e+00## socialMediaHours 1.00000000 1.0000 1.000000e+00 1.000000e+00## netflixHours 1.00000000 1.0000 1.000000e+00 1.000000e+00## parttimeJob 0.03173350 0.0000 0.000000e+00 0.000000e+00## attendance 1.00000000 1.0000 1.000000e+00 1.000000e+00## sleepHours 1.00000000 1.0000 1.000000e+00 1.000000e+00## diet 0.05716285 0.0000 1.000000e+00 0.000000e+00## exerciseFreq 1.00000000 1.0000 1.000000e+00 1.000000e+00## parentalEducation 0.02810779 0.0000 0.000000e+00 0.000000e+00## internet 0.05007748 0.0000 0.000000e+00 1.000000e+00## mentalHealth 1.00000000 1.0000 1.000000e+00 1.000000e+00## extracurricular 0.02769191 0.0000 0.000000e+00 0.000000e+00## BF NA 1.0000 6.004999e-02 5.222226e-02## PostProbs NA 0.7756 4.660000e-02 4.050000e-02## R2 NA 0.9011 9.012000e-01 9.012000e-01## dim NA 8.0000 9.000000e+00 9.000000e+00## logmarg NA 1125.5152 1.122703e+03 1.122563e+03## model 4 model 5## Intercept 1.0000000 1.000000e+00## age 0.0000000 1.000000e+00## gender 0.0000000 0.000000e+00## studyHours 1.0000000 1.000000e+00## socialMediaHours 1.0000000 1.000000e+00## netflixHours 1.0000000 1.000000e+00## parttimeJob 1.0000000 0.000000e+00## attendance 1.0000000 1.000000e+00## sleepHours 1.0000000 1.000000e+00## diet 0.0000000 0.000000e+00## exerciseFreq 1.0000000 1.000000e+00## parentalEducation 0.0000000 0.000000e+00## internet 0.0000000 0.000000e+00## mentalHealth 1.0000000 1.000000e+00## extracurricular 0.0000000 0.000000e+00## BF 0.0323007 2.857671e-02## PostProbs 0.0251000 2.220000e-02## R2 0.9011000 9.011000e-01## dim 9.0000000 9.000000e+00## logmarg 1122.0825448 1.121960e+03

##Model Comparison and Selection

**plot**(model\_noninform)

A graph of a graph with numbers and a line

Description automatically generated with medium confidenceA graph with a line and a line

Description automatically generated with medium confidenceA graph of a model

Description automatically generatedA graph with text and numbers

Description automatically generated

**plot**(model\_weak)

A graph of a graph with numbers and a line

Description automatically generated with medium confidenceA graph with a number of numbers and a line

Description automatically generated with medium confidenceA graph of a model

Description automatically generatedA graph with text and numbers

Description automatically generated

##Posterior Summaries

**Non-informative prior**

coef\_noninform <- **coef**(model\_noninform) *# Extract posterior mean, SD***print**(coef\_noninform)

## ## Marginal Posterior Summaries of Coefficients: ## ## Using BMA ## ## Based on the top 16384 models ## post mean post SD post p(B != 0)## Intercept 6.960e+01 1.686e-01 1.000e+00 ## age -4.370e-04 1.316e-02 3.119e-02 ## gender 4.016e-05 5.159e-02 3.064e-02 ## studyHours 9.565e+00 1.150e-01 1.000e+00 ## socialMediaHours -2.617e+00 1.441e-01 1.000e+00 ## netflixHours -2.275e+00 1.569e-01 1.000e+00 ## parttimeJob 7.723e-03 8.706e-02 3.517e-02 ## attendance 1.445e-01 1.797e-02 1.000e+00 ## sleepHours 2.002e+00 1.376e-01 1.000e+00 ## diet -1.817e-02 9.125e-02 6.322e-02 ## exerciseFreq 1.450e+00 8.335e-02 1.000e+00 ## parentalEducation 1.126e-03 3.489e-02 3.116e-02 ## internet -1.436e-02 8.076e-02 5.541e-02 ## mentalHealth 1.947e+00 5.923e-02 1.000e+00 ## extracurricular -6.875e-04 6.360e-02 3.070e-02

**Weak-informative prior**

coef\_weak <- **coef**(model\_weak)**print**(coef\_weak)

## ## Marginal Posterior Summaries of Coefficients: ## ## Using BMA ## ## Based on the top 16384 models ## post mean post SD post p(B != 0)## Intercept 6.960e+01 1.686e-01 1.000e+00 ## age -3.944e-04 1.250e-02 2.813e-02 ## gender 3.554e-05 4.900e-02 2.764e-02 ## studyHours 9.567e+00 1.150e-01 1.000e+00 ## socialMediaHours -2.617e+00 1.441e-01 1.000e+00 ## netflixHours -2.275e+00 1.569e-01 1.000e+00 ## parttimeJob 6.967e-03 8.273e-02 3.173e-02 ## attendance 1.445e-01 1.797e-02 1.000e+00 ## sleepHours 2.003e+00 1.376e-01 1.000e+00 ## diet -1.643e-02 8.694e-02 5.716e-02 ## exerciseFreq 1.451e+00 8.336e-02 1.000e+00 ## parentalEducation 1.016e-03 3.314e-02 2.811e-02 ## internet -1.298e-02 7.691e-02 5.008e-02 ## mentalHealth 1.947e+00 5.923e-02 1.000e+00 ## extracurricular -6.183e-04 6.041e-02 2.769e-02

**WAIC/DIC Calculation for BAS Models**

***## Model Comparison using BAS*** *# Calculate log marginal likelihoods for model comparison*log\_marginals <- **c**(model\_noninform**$**logmarg[**which.max**(model\_noninform**$**logmarg)], model\_weak**$**logmarg[**which.max**(model\_weak**$**logmarg)]) *# Approximate Bayes Factor (using log marginal likelihoods)*bf <- **exp**(log\_marginals[1] **-** log\_marginals[2])**cat**("Bayes Factor (Non-informative vs Weak):", **round**(bf, 3), "**\n**")

## Bayes Factor (Non-informative vs Weak): 3.666

*# Model probabilities***cat**("Model probabilities:**\n**")

## Model probabilities:

**cat**("Non-informative prior model:", **round**(**exp**(log\_marginals[1])**/sum**(**exp**(log\_marginals)), 3), "**\n**")

## Non-informative prior model: NaN

**cat**("Weak prior model:", **round**(**exp**(log\_marginals[2])**/sum**(**exp**(log\_marginals)), 3), "**\n**")

## Weak prior model: NaN

##Model Diagnostics Section

*# Check variable inclusion probabilities***cat**("**\n**Variable Inclusion Probabilities (Non-informative prior):**\n**")

## ## Variable Inclusion Probabilities (Non-informative prior):

**print**(model\_noninform**$**probne0[**-**1]) *# exclude intercept*

## [1] 0.03118669 0.03064390 1.00000000 1.00000000 1.00000000 0.03516870## [7] 1.00000000 1.00000000 0.06321724 1.00000000 0.03115783 0.05540887## [13] 1.00000000 0.03069762

**cat**("**\n**Variable Inclusion Probabilities (Weak prior):**\n**")

## ## Variable Inclusion Probabilities (Weak prior):

**print**(model\_weak**$**probne0[**-**1])

## [1] 0.02813477 0.02764352 1.00000000 1.00000000 1.00000000 0.03173350## [7] 1.00000000 1.00000000 0.05716285 1.00000000 0.02810779 0.05007748## [13] 1.00000000 0.02769191

*# Check model size distribution***cat**("**\n**Model Size Distribution:**\n**")

## ## Model Size Distribution:

**table**(model\_noninform**$**size)

## ## 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 ## 1 14 91 364 1001 2002 3003 3432 3003 2002 1001 364 91 14 1

##Posterior Predictive Checks

***## Posterior Predictive Checks - FIXED*** *# Simulate data from the best model and compare to observed*best\_model\_idx <- **which.max**(model\_noninform**$**postprobs)best\_model\_vars <- model\_noninform**$**which[[best\_model\_idx]] **+** 1 *# +1 to account for intercept  
  
# Extract the variables included in the best model (excluding intercept)*included\_vars <- best\_model\_vars[**-**1] **-** 1 *# -1 to adjust back to column indices***cat**("Variables in best model:", **colnames**(HabitsPerformanceData)[included\_vars], "**\n**")

## Variables in best model: studyHours socialMediaHours netflixHours attendance sleepHours exerciseFreq mentalHealth

*# Create design matrix for the best model***if** (**length**(included\_vars) **>** 0) { X\_best <- **as.matrix**(**cbind**(Intercept = 1, HabitsPerformanceData[, included\_vars]))} **else** { X\_best <- **matrix**(1, nrow = **nrow**(HabitsPerformanceData), ncol = 1) *# Intercept only*} *# Get coefficient estimates - FIXED ACCESS METHOD*beta\_hat <- model\_noninform**$**mle[[best\_model\_idx]] *# This is already the coefficient vector*y\_pred <- X\_best **%\*%** beta\_hat *# Compare observed vs predicted***par**(mfrow = **c**(1, 2))**plot**(HabitsPerformanceData**$**examScore, y\_pred,  xlab = "Observed Exam Scores", ylab = "Predicted Exam Scores", main = "Posterior Predictive Check", pch = 19, col = "blue")**abline**(0, 1, col = "red", lwd = 2) *# Residual plot*residuals <- HabitsPerformanceData**$**examScore **-** y\_pred**plot**(y\_pred, residuals,  xlab = "Predicted Values", ylab = "Residuals", main = "Residual Plot", pch = 19, col = "red")**abline**(h = 0, col = "blue", lwd = 2)

A red and blue graphs

Description automatically generated

*# Add some diagnostic statistics***cat**("**\n**Posterior Predictive Check Diagnostics:**\n**")

## ## Posterior Predictive Check Diagnostics:

**cat**("Mean Absolute Error:", **mean**(**abs**(residuals)), "**\n**")

## Mean Absolute Error: 63.44428

**cat**("Root Mean Squared Error:", **sqrt**(**mean**(residuals**^**2)), "**\n**")

## Root Mean Squared Error: 63.66605

**cat**("Correlation (Observed vs Predicted):", **cor**(HabitsPerformanceData**$**examScore, y\_pred), "**\n**")

## Correlation (Observed vs Predicted): 0.9492476

##Formal Model Comparison Table

***## ROBUST Model Comparison with Error Handling***model\_comparison <- **data.frame**( Model = **c**("Non-informative Prior", "Weak Prior")) *# Safely extract values with error handling*safe\_extract <- **function**(model, value\_name) { **tryCatch**({ **if** (value\_name **==** "logmarg") { val <- **max**(model[[value\_name]], na.rm = TRUE) **if** (**is.infinite**(val)) **return**(NA) **else** **return**(val) } **else** **if** (value\_name **==** "BIC") { *# Get BIC of the best model* best\_idx <- **which.max**(model**$**postprobs) **return**(model**$**BIC[best\_idx]) } **else** **if** (value\_name **==** "size") { best\_idx <- **which.max**(model**$**postprobs) **return**(model**$**size[best\_idx]) } **else** **if** (value\_name **==** "postprob") { **return**(**max**(model**$**postprobs)) } }, error = **function**(e) { **return**(NA) })} *# Fill comparison table safely*model\_comparison**$**Log\_Marginal <- **c**( **safe\_extract**(model\_noninform, "logmarg"), **safe\_extract**(model\_weak, "logmarg"))model\_comparison**$**BIC <- **c**( **safe\_extract**(model\_noninform, "BIC"), **safe\_extract**(model\_weak, "BIC"))model\_comparison**$**Size <- **c**( **safe\_extract**(model\_noninform, "size"), **safe\_extract**(model\_weak, "size"))model\_comparison**$**Posterior\_Prob <- **c**( **safe\_extract**(model\_noninform, "postprob"), **safe\_extract**(model\_weak, "postprob"))**cat**("**\n**=== ROBUST MODEL COMPARISON TABLE ===**\n**")

## ## === ROBUST MODEL COMPARISON TABLE ===

**print**(model\_comparison)

## Model Log\_Marginal Size Posterior\_Prob## 1 Non-informative Prior 1126.814 8 0.7528962## 2 Weak Prior 1125.515 8 0.7755590

*# Determine best model***if** (**!any**(**is.na**(model\_comparison**$**Log\_Marginal))) { best\_idx <- **which.max**(model\_comparison**$**Log\_Marginal) **cat**("**\n**Best model based on marginal likelihood:", model\_comparison**$**Model[best\_idx], "**\n**")} **else** **if** (**!any**(**is.na**(model\_comparison**$**Posterior\_Prob))) { best\_idx <- **which.max**(model\_comparison**$**Posterior\_Prob) **cat**("**\n**Best model based on posterior probability:", model\_comparison**$**Model[best\_idx], "**\n**")} **else** **if** (**!any**(**is.na**(model\_comparison**$**BIC))) { best\_idx <- **which.min**(model\_comparison**$**BIC) **cat**("**\n**Best model based on BIC:", model\_comparison**$**Model[best\_idx], "**\n**")} **else** { **cat**("**\n**Cannot determine best model due to missing values**\n**")}

## ## Best model based on marginal likelihood: Non-informative Prior

##Frequentist linear regression(p-values and confidence intervals)

lmScore <- **lm**(formula = examScore **~** . , data = HabitsPerformanceData)**summary**(lmScore)

## ## Call:## lm(formula = examScore ~ ., data = HabitsPerformanceData)## ## Residuals:## Min 1Q Median 3Q Max ## -21.8035 -3.4559 0.0299 3.6161 15.5633 ## ## Coefficients:## Estimate Std. Error t value Pr(>|t|) ## (Intercept) 6.88566 2.64573 2.603 0.00939 \*\* ## age -0.01275 0.07339 -0.174 0.86213 ## gender 0.01441 0.29621 0.049 0.96121 ## studyHours 9.58332 0.11542 83.027 < 2e-16 \*\*\*## socialMediaHours -2.61362 0.14460 -18.075 < 2e-16 \*\*\*## netflixHours -2.27304 0.15743 -14.438 < 2e-16 \*\*\*## parttimeJob 0.23931 0.41256 0.580 0.56200 ## attendance 0.14320 0.01813 7.900 7.41e-15 \*\*\*## sleepHours 1.99976 0.13830 14.459 < 2e-16 \*\*\*## diet -0.28284 0.23427 -1.207 0.22760 ## exerciseFreq 1.45125 0.08380 17.318 < 2e-16 \*\*\*## parentalEducation 0.04525 0.19517 0.232 0.81669 ## internet -0.25407 0.23443 -1.084 0.27873 ## mentalHealth 1.94698 0.05954 32.701 < 2e-16 \*\*\*## extracurricular -0.04210 0.36342 -0.116 0.90780 ## ---## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1## ## Residual standard error: 5.341 on 985 degrees of freedom## Multiple R-squared: 0.9014, Adjusted R-squared: 0.9 ## F-statistic: 643.1 on 14 and 985 DF, p-value: < 2.2e-16

##The scatter plots and the fitted simple linear regression lines of the selected explanatory variables versus exam score variable

##Load ggplot2 library.

**library**(ggplot2)

scPlot1 <- **ggplot**(data = HabitsPerformanceData , mapping = **aes**(x = age , y = examScore)) **+** **geom\_point**(color="blue") **+** **xlab**("Age") **+** **ylab**("Exam Score") **+** **geom\_smooth**(method=lm, color="red") scPlot1

## `geom\_smooth()` using formula = 'y ~ x'

A graph with numbers and lines

Description automatically generated

scPlot2 <- **ggplot**(data = HabitsPerformanceData , mapping = **aes**(x = gender , y = examScore)) **+** **geom\_point**(color="blue") **+** **xlab**("Gender") **+** **ylab**("Exam Score") **+** **geom\_smooth**(method=lm, color="red") scPlot2

## `geom\_smooth()` using formula = 'y ~ x'

A graph with a line graph and numbers

Description automatically generated

scPlot3 <- **ggplot**(data = HabitsPerformanceData , mapping = **aes**(x = studyHours , y = examScore)) **+** **geom\_point**(color="blue") **+** **xlab**("Study Hours") **+** **ylab**("Exam Score") **+** **geom\_smooth**(method=lm, color="red")scPlot3

## `geom\_smooth()` using formula = 'y ~ x'

A graph of a line graph

Description automatically generated

scPlot4 <- **ggplot**(data = HabitsPerformanceData , mapping = **aes**(x = socialMediaHours , y = examScore)) **+** **geom\_point**(color="blue") **+** **xlab**("Social Media Hours") **+** **ylab**("Exam Score") **+** **geom\_smooth**(method=lm, color="red")scPlot4

*A graph with blue dots and a red line

Description automatically generated*## `geom\_smooth()` using formula = 'y ~ x'

scPlot5 <- **ggplot**(data = HabitsPerformanceData , mapping = **aes**(x = netflixHours , y = examScore)) **+** **geom\_point**(color="blue") **+** **xlab**("Netflix Hours") **+** **ylab**("Exam Score") **+** **geom\_smooth**(method=lm, color="red")scPlot5

## `geom\_smooth()` using formula = 'y ~ x'

A graph showing a line going down

Description automatically generated with medium confidence

scPlot6 <- **ggplot**(data = HabitsPerformanceData , mapping = **aes**(x = parttimeJob , y = examScore)) **+** **geom\_point**(color="blue") **+** **xlab**("Parttime Job") **+** **ylab**("Exam Score") **+** **geom\_smooth**(method=lm, color="red") scPlot6

## `geom\_smooth()` using formula = 'y ~ x'

A graph of a line graph

Description automatically generated

scPlot7 <- **ggplot**(data = HabitsPerformanceData , mapping = **aes**(x = attendance , y = examScore)) **+** **geom\_point**(color="blue") **+** **xlab**("Attendance") **+** **ylab**("Exam Score") **+** **geom\_smooth**(method=lm, color="red")scPlot7

## `geom\_smooth()` using formula = 'y ~ x'

A graph with blue dots and a red line

Description automatically generated

scPlot8 <- **ggplot**(data = HabitsPerformanceData , mapping = **aes**(x = sleepHours , y = examScore)) **+** **geom\_point**(color="blue") **+** **xlab**("Sleep Hours") **+** **ylab**("Exam Score") **+** **geom\_smooth**(method=lm, color="red")scPlot8

## `geom\_smooth()` using formula = 'y ~ x'

A graph with blue dots

Description automatically generated

scPlot9 <- **ggplot**(data = HabitsPerformanceData , mapping = **aes**(x = diet , y = examScore)) **+** **geom\_point**(color="blue") **+** **xlab**("Diet") **+** **ylab**("Exam Score") **+** **geom\_smooth**(method=lm, color="red") scPlot9

## `geom\_smooth()` using formula = 'y ~ x'

A graph with a line and a red line

Description automatically generated

scPlot10 <- **ggplot**(data = HabitsPerformanceData , mapping = **aes**(x = exerciseFreq , y = examScore)) **+** **geom\_point**(color="blue") **+** **xlab**("Exercise frequency") **+** **ylab**("Exam Score") **+** **geom\_smooth**(method=lm, color="red") scPlot10

## `geom\_smooth()` using formula = 'y ~ x'

A graph of exercise frequency

Description automatically generated

scPlot11 <- **ggplot**(data = HabitsPerformanceData , mapping = **aes**(x = parentalEducation , y = examScore)) **+** **geom\_point**(color="blue") **+** **xlab**("parental Education Level") **+** **ylab**("Exam Score") **+** **geom\_smooth**(method=lm, color="red") scPlot11

## `geom\_smooth()` using formula = 'y ~ x'

A graph with a red line

Description automatically generated

scPlot12 <- **ggplot**(data = HabitsPerformanceData , mapping = **aes**(x = internet , y = examScore)) **+** **geom\_point**(color="blue") **+** **xlab**("Internet") **+** **ylab**("Exam Score") **+** **geom\_smooth**(method=lm, color="red") scPlot12

## `geom\_smooth()` using formula = 'y ~ x'

A graph with a line and a line

Description automatically generated

scPlot13 <- **ggplot**(data = HabitsPerformanceData , mapping = **aes**(x = mentalHealth , y = examScore)) **+** **geom\_point**(color="blue") **+** **xlab**("Mental Health") **+** **ylab**("Exam Score") **+** **geom\_smooth**(method=lm, color="red") scPlot13

## `geom\_smooth()` using formula = 'y ~ x'

A graph of a graph with a line and a red line

Description automatically generated with medium confidence

scPlot14 <- **ggplot**(data = HabitsPerformanceData , mapping = **aes**(x = extracurricular , y = examScore)) **+** **geom\_point**(color="blue") **+** **xlab**("Extracurricular Participation") **+** **ylab**("Exam Score") **+** **geom\_smooth**(method=lm, color="red") scPlot14

## `geom\_smooth()` using formula = 'y ~ x'

A graph with a line

Description automatically generated

**Bayesian Simple Linear Regression**

**We can also use the ‘BAS’ package to find the best BIC HabitsPerformanceData without taking the stepwise backward process.**

##Load BAS Library

**library**(BAS)

##Get the summary output of the above HabitsPerformanceData.

score.lm1 <- **bas.lm**(formula = examScore **~** . , data = HabitsPerformanceData, prior="BIC", modelprior=**uniform**()) *# Coefficients averaged across models (Bayesian Model Averaging)***coef**(score.lm1, estimator = "BMA")

## ## Marginal Posterior Summaries of Coefficients: ## ## Using BMA ## ## Based on the top 16384 models ## post mean post SD post p(B != 0)## Intercept 6.960e+01 1.686e-01 1.000e+00 ## age -4.378e-04 1.317e-02 3.121e-02 ## gender 4.028e-05 5.163e-02 3.066e-02 ## studyHours 9.575e+00 1.150e-01 1.000e+00 ## socialMediaHours -2.619e+00 1.442e-01 1.000e+00 ## netflixHours -2.277e+00 1.570e-01 1.000e+00 ## parttimeJob 7.747e-03 8.721e-02 3.524e-02 ## attendance 1.446e-01 1.798e-02 1.000e+00 ## sleepHours 2.004e+00 1.377e-01 1.000e+00 ## diet -1.834e-02 9.169e-02 6.375e-02 ## exerciseFreq 1.452e+00 8.340e-02 1.000e+00 ## parentalEducation 1.128e-03 3.492e-02 3.118e-02 ## internet -1.448e-02 8.110e-02 5.580e-02 ## mentalHealth 1.949e+00 5.926e-02 1.000e+00 ## extracurricular -6.889e-04 6.365e-02 3.071e-02

*# Coefficients from the single best model (highest posterior probability)***coef**(score.lm1, estimator = "HPM")

## ## Marginal Posterior Summaries of Coefficients: ## ## Using HPM ## ## Based on the top 1 models ## post mean post SD post p(B != 0)## Intercept 69.60150 0.16857 1.00000 ## age 0.00000 0.00000 0.03121 ## gender 0.00000 0.00000 0.03066 ## studyHours 9.57456 0.11503 1.00000 ## socialMediaHours -2.61978 0.14413 1.00000 ## netflixHours -2.27708 0.15697 1.00000 ## parttimeJob 0.00000 0.00000 0.03524 ## attendance 0.14473 0.01797 1.00000 ## sleepHours 2.00462 0.13764 1.00000 ## diet 0.00000 0.00000 0.06375 ## exerciseFreq 1.45187 0.08338 1.00000 ## parentalEducation 0.00000 0.00000 0.03118 ## internet 0.00000 0.00000 0.05580 ## mentalHealth 1.94891 0.05924 1.00000 ## extracurricular 0.00000 0.00000 0.03071

##Fit a simple linear regression HabitsPerformanceData of examScores versus studyHours.

score.lm1 <- **lm**(formula = examScore **~** studyHours , data = HabitsPerformanceData)**summary**(score.lm1)

## ## Call:## lm(formula = examScore ~ studyHours, data = HabitsPerformanceData)## ## Residuals:## Min 1Q Median 3Q Max ## -25.979 -6.626 0.236 6.537 34.319 ## ## Coefficients:## Estimate Std. Error t value Pr(>|t|) ## (Intercept) 35.9102 0.7893 45.50 <2e-16 \*\*\*## studyHours 9.4903 0.2055 46.19 <2e-16 \*\*\*## ---## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1## ## Residual standard error: 9.539 on 998 degrees of freedom## Multiple R-squared: 0.6813, Adjusted R-squared: 0.681 ## F-statistic: 2134 on 1 and 998 DF, p-value: < 2.2e-16

##Obtain residuals and n.(Residual analysis checks model accuracy and assumptions. A smaller MSE = better fit.)

resid <- **residuals**(score.lm1) n <- **length**(resid)n

## [1] 1000

##Calculate MSE

MSE <- 1**/**(n-2) **\*** **sum**((resid **^** 2)) MSE

## [1] 90.98735

##Combine residuals and fitted values into a data frame.

result <- **data.frame**(fitted\_values = **fitted.values**(score.lm1) , residuals = **residuals**(score.lm1))

##Load library and plot residuals versus fitted values.

**library**(ggplot2)**ggplot**(data = result , **aes**(x = fitted\_values , y = residuals)) **+** **geom\_point**(color = "blue" , pch = 1 , size = 2) **+** **geom\_abline**(intercept = 0 , slope = 0) **+** **xlab**(**expression**(**paste**("Fitted Value " , **widehat**(examScore)))) **+** **ylab**("Residuals")

A graph showing a number of blue dots

Description automatically generated

##Find the observation with the largest fitted value.

**which.max**(**as.vector**(**fitted.values**(score.lm1)))

## [1] 456

HabitsPerformanceData**$**studyHours[456] ***##model predicts the highest study hours per day***

## [1] 8.3

##Shows this observation has the maximum studyHours.

**which.max**(HabitsPerformanceData**$**studyHours)

## [1] 456

HabitsPerformanceData**$**studyHours[456] ***##the highest actual study hours per day***

## [1] 8.3

##Normal probability plot of the residuals.(to check normality assumption)

**plot**(score.lm1, which = 2)

A graph of a line

Description automatically generated

##Credible Intervals for Slope Beta and y-Intercept alpha.

output <- **summary**(score.lm1)**$**coef[, 1**:**2] out <- **cbind**(output, **confint**(score.lm1)) **colnames**(out) <- **c**("Posterior Mean", "Posterior Std", "2.5", "97.5") **round**(out, 3)

## Posterior Mean Posterior Std 2.5 97.5## (Intercept) 35.91 0.789 34.361 37.459## studyHours 9.49 0.205 9.087 9.893

**library**(ggplot2)

##Construct current prediction.

alpha <- score.lm1**$**coefficients[1]alpha

## (Intercept) ## 35.91016

beta <- score.lm1**$**coefficients[2]beta

## studyHours ## 9.49025

new\_x <- **seq**(**min**(HabitsPerformanceData**$**studyHours) , **max**(HabitsPerformanceData**$**studyHours) , length.out = 100)y\_hat <- alpha **+** beta**\***new\_x

##Get lower and upper bounds for mean.

ymean <- **data.frame**(**predict**(score.lm1 , newdata = **data.frame**(studyHours = new\_x) , interval = "confidence" , level = 0.95))

##Get lower and upper bounds for prediction.

ypred <- **data.frame**(**predict**(score.lm1 , newdata = **data.frame**(studyHours = new\_x) , interval = "prediction" , level = 0.95))output <- **data.frame**(x = new\_x ,  y\_hat = **pmin**(**pmax**(y\_hat, 0), 100), ymean\_lwr = **pmin**(**pmax**(ymean**$**lwr, 0), 100), ymean\_upr = **pmin**(**pmax**(ymean**$**upr, 0), 100), ypred\_lwr = **pmin**(**pmax**(ypred**$**lwr, 0), 100), ypred\_upr = **pmin**(**pmax**(ypred**$**upr, 0), 100))output

## x y\_hat ymean\_lwr ymean\_upr ypred\_lwr ypred\_upr## 1 0.00000000 35.91016 34.36128 37.45904 17.12792 54.69240## 2 0.08383838 36.70581 35.18811 38.22351 17.92612 55.48550## 3 0.16767677 37.50146 36.01482 38.98809 18.72425 56.27867## 4 0.25151515 38.29710 36.84142 39.75279 19.52232 57.07189## 5 0.33535354 39.09275 37.66788 40.51762 20.32033 57.86517## 6 0.41919192 39.88840 38.49420 41.28260 21.11828 58.65852## 7 0.50303030 40.68405 39.32038 42.04771 21.91617 59.45192## 8 0.58686869 41.47969 40.14639 42.81299 22.71400 60.24538## 9 0.67070707 42.27534 40.97224 43.57844 23.51177 61.03891## 10 0.75454545 43.07099 41.79791 44.34406 24.30948 61.83250## 11 0.83838384 43.86663 42.62338 45.10988 25.10713 62.62614## 12 0.92222222 44.66228 43.44865 45.87591 25.90471 63.41985## 13 1.00606061 45.45793 44.27369 46.64217 26.70224 64.21362## 14 1.08989899 46.25358 45.09849 47.40866 27.49970 65.00745## 15 1.17373737 47.04922 45.92303 48.17542 28.29711 65.80134## 16 1.25757576 47.84487 46.74729 48.94245 29.09445 66.59529## 17 1.34141414 48.64052 47.57125 49.70979 29.89174 67.38930## 18 1.42525253 49.43617 48.39488 50.47745 30.68896 68.18337## 19 1.50909091 50.23181 49.21815 51.24548 31.48612 68.97751## 20 1.59292929 51.02746 50.04104 52.01388 32.28322 69.77170## 21 1.67676768 51.82311 50.86351 52.78270 33.08026 70.56595## 22 1.76060606 52.61875 51.68553 53.55198 33.87724 71.36027## 23 1.84444444 53.41440 52.50706 54.32175 34.67416 72.15465## 24 1.92828283 54.21005 53.32805 55.09205 35.47101 72.94908## 25 2.01212121 55.00570 54.14845 55.86294 36.26781 73.74358## 26 2.09595960 55.80134 54.96822 56.63446 37.06455 74.53814## 27 2.17979798 56.59699 55.78730 57.40668 37.86122 75.33276## 28 2.26363636 57.39264 56.60562 58.17965 38.65783 76.12744## 29 2.34747475 58.18829 57.42312 58.95345 39.45439 76.92218## 30 2.43131313 58.98393 58.23973 59.72813 40.25088 77.71699## 31 2.51515152 59.77958 59.05537 60.50379 41.04731 78.51185## 32 2.59898990 60.57523 59.86995 61.28051 41.84368 79.30678## 33 2.68282828 61.37087 60.68339 62.05836 42.63999 80.10176## 34 2.76666667 62.16652 61.49560 62.83745 43.43623 80.89681## 35 2.85050505 62.96217 62.30648 63.61786 44.23242 81.69192## 36 2.93434343 63.75782 63.11594 64.39969 45.02855 82.48708## 37 3.01818182 64.55346 63.92389 65.18304 45.82461 83.28231## 38 3.10202020 65.34911 64.73023 65.96799 46.62062 84.07761## 39 3.18585859 66.14476 65.53489 66.75463 47.41656 84.87296## 40 3.26969697 66.94041 66.33778 67.54303 48.21244 85.66837## 41 3.35353535 67.73605 67.13885 68.33326 49.00826 86.46384## 42 3.43737374 68.53170 67.93803 69.12537 49.80402 87.25938## 43 3.52121212 69.32735 68.73531 69.91938 50.59972 88.05497## 44 3.60505051 70.12299 69.53066 70.71533 51.39536 88.85063## 45 3.68888889 70.91864 70.32408 71.51320 52.19093 89.64635## 46 3.77272727 71.71429 71.11560 72.31298 52.98645 90.44213## 47 3.85656566 72.50994 71.90525 73.11462 53.78190 91.23797## 48 3.94040404 73.30558 72.69310 73.91807 54.57730 92.03387## 49 4.02424242 74.10123 73.47920 74.72326 55.37263 92.82983## 50 4.10808081 74.89688 74.26365 75.53011 56.16790 93.62585## 51 4.19191919 75.69252 75.04651 76.33854 56.96311 94.42194## 52 4.27575758 76.48817 75.82789 77.14845 57.75826 95.21808## 53 4.35959596 77.28382 76.60788 77.95976 58.55335 96.01429## 54 4.44343434 78.07947 77.38658 78.77236 59.34838 96.81055## 55 4.52727273 78.87511 78.16407 79.58616 60.14335 97.60688## 56 4.61111111 79.67076 78.94044 80.40108 60.93825 98.40327## 57 4.69494949 80.46641 79.71579 81.21702 61.73310 99.19972## 58 4.77878788 81.26206 80.49019 82.03392 62.52788 99.99623## 59 4.86262626 82.05770 81.26372 82.85168 63.32260 100.00000## 60 4.94646465 82.85335 82.03645 83.67025 64.11727 100.00000## 61 5.03030303 83.64900 82.80845 84.48955 64.91187 100.00000## 62 5.11414141 84.44464 83.57977 85.30952 65.70641 100.00000## 63 5.19797980 85.24029 84.35047 86.13011 66.50089 100.00000## 64 5.28181818 86.03594 85.12061 86.95127 67.29531 100.00000## 65 5.36565657 86.83159 85.89022 87.77296 68.08966 100.00000## 66 5.44949495 87.62723 86.65935 88.59512 68.88396 100.00000## 67 5.53333333 88.42288 87.42804 89.41773 69.67820 100.00000## 68 5.61717172 89.21853 88.19632 90.24074 70.47237 100.00000## 69 5.70101010 90.01418 88.96422 91.06413 71.26648 100.00000## 70 5.78484848 90.80982 89.73179 91.88786 72.06054 100.00000## 71 5.86868687 91.60547 90.49903 92.71191 72.85453 100.00000## 72 5.95252525 92.40112 91.26597 93.53626 73.64846 100.00000## 73 6.03636364 93.19676 92.03264 94.36089 74.44233 100.00000## 74 6.12020202 93.99241 92.79906 95.18576 75.23614 100.00000## 75 6.20404040 94.78806 93.56524 96.01087 76.02989 100.00000## 76 6.28787879 95.58371 94.33121 96.83621 76.82358 100.00000## 77 6.37171717 96.37935 95.09697 97.66174 77.61721 100.00000## 78 6.45555556 97.17500 95.86253 98.48747 78.41078 100.00000## 79 6.53939394 97.97065 96.62793 99.31337 79.20428 100.00000## 80 6.62323232 98.76630 97.39315 100.00000 79.99773 100.00000## 81 6.70707071 99.56194 98.15822 100.00000 80.79112 100.00000## 82 6.79090909 100.00000 98.92315 100.00000 81.58444 100.00000## 83 6.87474747 100.00000 99.68794 100.00000 82.37771 100.00000## 84 6.95858586 100.00000 100.00000 100.00000 83.17091 100.00000## 85 7.04242424 100.00000 100.00000 100.00000 83.96405 100.00000## 86 7.12626263 100.00000 100.00000 100.00000 84.75714 100.00000## 87 7.21010101 100.00000 100.00000 100.00000 85.55016 100.00000## 88 7.29393939 100.00000 100.00000 100.00000 86.34312 100.00000## 89 7.37777778 100.00000 100.00000 100.00000 87.13602 100.00000## 90 7.46161616 100.00000 100.00000 100.00000 87.92886 100.00000## 91 7.54545455 100.00000 100.00000 100.00000 88.72164 100.00000## 92 7.62929293 100.00000 100.00000 100.00000 89.51436 100.00000## 93 7.71313131 100.00000 100.00000 100.00000 90.30702 100.00000## 94 7.79696970 100.00000 100.00000 100.00000 91.09962 100.00000## 95 7.88080808 100.00000 100.00000 100.00000 91.89216 100.00000## 96 7.96464646 100.00000 100.00000 100.00000 92.68464 100.00000## 97 8.04848485 100.00000 100.00000 100.00000 93.47706 100.00000## 98 8.13232323 100.00000 100.00000 100.00000 94.26942 100.00000## 99 8.21616162 100.00000 100.00000 100.00000 95.06172 100.00000## 100 8.30000000 100.00000 100.00000 100.00000 95.85396 100.00000

##Extract potential outlier data point.

outlier <- **data.frame**(x = HabitsPerformanceData**$**studyHours[456] , y = HabitsPerformanceData**$**examScore[456])outlier

## x y## 1 8.3 100

##Scatter plot of original.

plot1 <- **ggplot**(data = HabitsPerformanceData , **aes**(x = studyHours , y = examScore)) **+** **geom\_point**(color = "blue")

##Add bounds of mean and prediction.

plot2 <- plot1 **+**  **geom\_line**(data = output , **aes**(x = new\_x , y = y\_hat , color = "first") , lty = 1) **+**  **geom\_line**(data = output , **aes**(x = new\_x , y = ymean\_lwr , lty = "second")) **+**  **geom\_line**(data = output , **aes**(x = new\_x , y = ymean\_upr , lty = "second")) **+**  **geom\_line**(data = output , **aes**(x = new\_x , y = ypred\_upr , lty = "third")) **+**  **geom\_line**(data = output , **aes**(x = new\_x , y = ypred\_lwr , lty = "third")) **+**  **scale\_colour\_manual**(values = **c**("orange") , labels = "Posterior mean" , name = "") **+**  **scale\_linetype\_manual**(values = **c**(2,3) , labels = **c**("95% CI for mean" , "95% CI for predictions") , name = "") **+**  **theme\_bw**() **+** **theme**(legend.position = **c**(1,0) , legend.justification = **c**(1.5,0))

## Warning: A numeric `legend.position` argument in `theme()` was deprecated in ggplot2## 3.5.0.## ℹ Please use the `legend.position.inside` argument of `theme()` instead.## This warning is displayed once every 8 hours.## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was## generated.

##Identify potential outlier.

plot2 **+** **geom\_point**(data = outlier , **aes**(x = x , y = y) , color = "orange" , pch = 1 , cex = 5)

A graph of blue dots

Description automatically generated

**Bayesian Multiple Linear Regression**

##Import library.

**library**(BAS)

##Use *bas.lm* to run regression HabitsPerformanceData.

score.bas = **lm**(examScore **~** . , data = HabitsPerformanceData)**summary**(score.bas)

## ## Call:## lm(formula = examScore ~ ., data = HabitsPerformanceData)## ## Residuals:## Min 1Q Median 3Q Max ## -21.8035 -3.4559 0.0299 3.6161 15.5633 ## ## Coefficients:## Estimate Std. Error t value Pr(>|t|) ## (Intercept) 6.88566 2.64573 2.603 0.00939 \*\* ## age -0.01275 0.07339 -0.174 0.86213 ## gender 0.01441 0.29621 0.049 0.96121 ## studyHours 9.58332 0.11542 83.027 < 2e-16 \*\*\*## socialMediaHours -2.61362 0.14460 -18.075 < 2e-16 \*\*\*## netflixHours -2.27304 0.15743 -14.438 < 2e-16 \*\*\*## parttimeJob 0.23931 0.41256 0.580 0.56200 ## attendance 0.14320 0.01813 7.900 7.41e-15 \*\*\*## sleepHours 1.99976 0.13830 14.459 < 2e-16 \*\*\*## diet -0.28284 0.23427 -1.207 0.22760 ## exerciseFreq 1.45125 0.08380 17.318 < 2e-16 \*\*\*## parentalEducation 0.04525 0.19517 0.232 0.81669 ## internet -0.25407 0.23443 -1.084 0.27873 ## mentalHealth 1.94698 0.05954 32.701 < 2e-16 \*\*\*## extracurricular -0.04210 0.36342 -0.116 0.90780 ## ---## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1## ## Residual standard error: 5.341 on 985 degrees of freedom## Multiple R-squared: 0.9014, Adjusted R-squared: 0.9 ## F-statistic: 643.1 on 14 and 985 DF, p-value: < 2.2e-16

**par**(mfrow = **c**(2,2))**plot**(score.bas)

A group of graphs showing different values

Description automatically generated

##Use *bas.lm* to run regression HabitsPerformanceData.

score.bas2 <- **bas.lm**(examScore **~** . , data = HabitsPerformanceData ,  prior = "BIC" ,  modelprior = **Bernoulli**(1) , include.always = **~** . ,  n.models = 1)***##Posterior Means and Posterior Standard Deviations.***score.coef = **coef**(score.bas2)score.coef

## ## Marginal Posterior Summaries of Coefficients: ## ## Using BMA ## ## Based on the top 1 models ## post mean post SD post p(B != 0)## Intercept 69.60150 0.16890 1.00000 ## age -0.01275 0.07339 1.00000 ## gender 0.01441 0.29621 1.00000 ## studyHours 9.58332 0.11542 1.00000 ## socialMediaHours -2.61362 0.14460 1.00000 ## netflixHours -2.27304 0.15743 1.00000 ## parttimeJob 0.23931 0.41256 1.00000 ## attendance 0.14320 0.01813 1.00000 ## sleepHours 1.99976 0.13830 1.00000 ## diet -0.28284 0.23427 1.00000 ## exerciseFreq 1.45125 0.08380 1.00000 ## parentalEducation 0.04525 0.19517 1.00000 ## internet -0.25407 0.23443 1.00000 ## mentalHealth 1.94698 0.05954 1.00000 ## extracurricular -0.04210 0.36342 1.00000

##visualization of the coefficients.

**par**(mfrow = **c**(2, 4))**plot**(score.coef , ask = F)

A group of black lines with white text

Description automatically generatedA group of black lines with white text

Description automatically generated

##Summary Table.

out <- **confint**(score.coef)[, 1**:**2]***## Extract the upper and lower bounds of the credible intervals***names = **c**("posterior mean", "posterior std", **colnames**(out))out = **cbind**(score.coef**$**postmean, score.coef**$**postsd, out)**colnames**(out) = names**round**(out, 2)

## posterior mean posterior std 2.5% 97.5%## Intercept 69.60 0.17 69.27 69.93## age -0.01 0.07 -0.16 0.13## gender 0.01 0.30 -0.57 0.60## studyHours 9.58 0.12 9.36 9.81## socialMediaHours -2.61 0.14 -2.90 -2.33## netflixHours -2.27 0.16 -2.58 -1.96## parttimeJob 0.24 0.41 -0.57 1.05## attendance 0.14 0.02 0.11 0.18## sleepHours 2.00 0.14 1.73 2.27## diet -0.28 0.23 -0.74 0.18## exerciseFreq 1.45 0.08 1.29 1.62## parentalEducation 0.05 0.20 -0.34 0.43## internet -0.25 0.23 -0.71 0.21## mentalHealth 1.95 0.06 1.83 2.06## extracurricular -0.04 0.36 -0.76 0.67

**Bayesian Model Selection**

*# Total num of observations*n <- **nrow**(HabitsPerformanceData)n

## [1] 1000

sco.lm1 <- **lm**(examScore **~** . , data = HabitsPerformanceData)sco.step <- **step**(sco.lm1, k = **log**(n))

## Start: AIC=3439.4## examScore ~ age + gender + studyHours + socialMediaHours + netflixHours + ## parttimeJob + attendance + sleepHours + diet + exerciseFreq + ## parentalEducation + internet + mentalHealth + extracurricular## ## Df Sum of Sq RSS AIC## - gender 1 0 28100 3432.5## - extracurricular 1 0 28101 3432.5## - age 1 1 28101 3432.5## - parentalEducation 1 2 28102 3432.5## - parttimeJob 1 10 28110 3432.8## - internet 1 34 28134 3433.7## - diet 1 42 28142 3434.0## <none> 28100 3439.4## - attendance 1 1781 29881 3493.9## - netflixHours 1 5947 34048 3624.5## - sleepHours 1 5965 34065 3625.0## - exerciseFreq 1 8556 36656 3698.3## - socialMediaHours 1 9321 37421 3718.9## - mentalHealth 1 30507 58607 4167.6## - studyHours 1 196658 224759 5511.7## ## Step: AIC=3432.5## examScore ~ age + studyHours + socialMediaHours + netflixHours + ## parttimeJob + attendance + sleepHours + diet + exerciseFreq + ## parentalEducation + internet + mentalHealth + extracurricular## ## Df Sum of Sq RSS AIC## - extracurricular 1 0 28101 3425.6## - age 1 1 28101 3425.6## - parentalEducation 1 2 28102 3425.6## - parttimeJob 1 10 28110 3425.9## - internet 1 33 28134 3426.8## - diet 1 42 28142 3427.1## <none> 28100 3432.5## - attendance 1 1781 29882 3487.1## - netflixHours 1 5948 34048 3617.6## - sleepHours 1 5977 34078 3618.4## - exerciseFreq 1 8585 36686 3692.2## - socialMediaHours 1 9321 37421 3712.0## - mentalHealth 1 30511 58611 4160.7## - studyHours 1 196777 224878 5505.4## ## Step: AIC=3425.6## examScore ~ age + studyHours + socialMediaHours + netflixHours + ## parttimeJob + attendance + sleepHours + diet + exerciseFreq + ## parentalEducation + internet + mentalHealth## ## Df Sum of Sq RSS AIC## - age 1 1 28102 3418.7## - parentalEducation 1 2 28102 3418.7## - parttimeJob 1 10 28111 3419.0## - internet 1 33 28134 3419.9## - diet 1 42 28142 3420.2## <none> 28101 3425.6## - attendance 1 1783 29884 3480.2## - netflixHours 1 5947 34048 3610.7## - sleepHours 1 5979 34080 3611.6## - exerciseFreq 1 8587 36688 3685.3## - socialMediaHours 1 9321 37422 3705.2## - mentalHealth 1 30513 58614 4153.9## - studyHours 1 196778 224879 5498.5## ## Step: AIC=3418.72## examScore ~ studyHours + socialMediaHours + netflixHours + parttimeJob + ## attendance + sleepHours + diet + exerciseFreq + parentalEducation + ## internet + mentalHealth## ## Df Sum of Sq RSS AIC## - parentalEducation 1 2 28103 3411.9## - parttimeJob 1 10 28111 3412.2## - internet 1 33 28135 3413.0## - diet 1 42 28143 3413.3## <none> 28102 3418.7## - attendance 1 1787 29889 3473.5## - netflixHours 1 5947 34049 3603.8## - sleepHours 1 5982 34084 3604.8## - exerciseFreq 1 8588 36690 3678.5## - socialMediaHours 1 9320 37422 3698.2## - mentalHealth 1 30590 58691 4148.3## - studyHours 1 196779 224881 5491.6## ## Step: AIC=3411.87## examScore ~ studyHours + socialMediaHours + netflixHours + parttimeJob + ## attendance + sleepHours + diet + exerciseFreq + internet + ## mentalHealth## ## Df Sum of Sq RSS AIC## - parttimeJob 1 10 28113 3405.3## - internet 1 33 28136 3406.1## - diet 1 42 28145 3406.4## <none> 28103 3411.9## - attendance 1 1789 29892 3466.7## - netflixHours 1 5947 34050 3596.9## - sleepHours 1 5989 34092 3598.1## - exerciseFreq 1 8588 36691 3671.6## - socialMediaHours 1 9325 37428 3691.5## - mentalHealth 1 30595 58698 4141.5## - studyHours 1 196793 224896 5484.7## ## Step: AIC=3405.3## examScore ~ studyHours + socialMediaHours + netflixHours + attendance + ## sleepHours + diet + exerciseFreq + internet + mentalHealth## ## Df Sum of Sq RSS AIC## - internet 1 33 28145 3399.6## - diet 1 41 28153 3399.8## <none> 28113 3405.3## - attendance 1 1781 29894 3459.8## - netflixHours 1 5944 34057 3590.2## - sleepHours 1 5990 34103 3591.5## - exerciseFreq 1 8579 36692 3664.7## - socialMediaHours 1 9317 37429 3684.6## - mentalHealth 1 30612 58725 4135.0## - studyHours 1 196888 225001 5478.3## ## Step: AIC=3399.55## examScore ~ studyHours + socialMediaHours + netflixHours + attendance + ## sleepHours + diet + exerciseFreq + mentalHealth## ## Df Sum of Sq RSS AIC## - diet 1 43 28189 3394.2## <none> 28145 3399.6## - attendance 1 1804 29949 3454.8## - sleepHours 1 5987 34132 3585.5## - netflixHours 1 5989 34134 3585.6## - exerciseFreq 1 8624 36770 3659.9## - socialMediaHours 1 9369 37514 3680.0## - mentalHealth 1 30791 58937 4131.7## - studyHours 1 196856 225001 5471.4## ## Step: AIC=3394.18## examScore ~ studyHours + socialMediaHours + netflixHours + attendance + ## sleepHours + exerciseFreq + mentalHealth## ## Df Sum of Sq RSS AIC## <none> 28189 3394.2## - attendance 1 1843 30032 3450.6## - netflixHours 1 5980 34169 3579.7## - sleepHours 1 6027 34216 3581.0## - exerciseFreq 1 8616 36805 3654.0## - socialMediaHours 1 9388 37577 3674.7## - mentalHealth 1 30752 58941 4124.9## - studyHours 1 196883 225072 5464.8

`

**library**(BAS)

##Model

basModel <- **bas.lm**(formula = examScore **~** . , data = HabitsPerformanceData , prior = "BIC" , modelprior = **uniform**()) *# equal prior to the model*

##bas\_model

basCoeff <- **coef**(basModel) basCoeff

## ## Marginal Posterior Summaries of Coefficients: ## ## Using BMA ## ## Based on the top 16384 models ## post mean post SD post p(B != 0)## Intercept 6.960e+01 1.686e-01 1.000e+00 ## age -4.378e-04 1.317e-02 3.121e-02 ## gender 4.028e-05 5.163e-02 3.066e-02 ## studyHours 9.575e+00 1.150e-01 1.000e+00 ## socialMediaHours -2.619e+00 1.442e-01 1.000e+00 ## netflixHours -2.277e+00 1.570e-01 1.000e+00 ## parttimeJob 7.747e-03 8.721e-02 3.524e-02 ## attendance 1.446e-01 1.798e-02 1.000e+00 ## sleepHours 2.004e+00 1.377e-01 1.000e+00 ## diet -1.834e-02 9.169e-02 6.375e-02 ## exerciseFreq 1.452e+00 8.340e-02 1.000e+00 ## parentalEducation 1.128e-03 3.492e-02 3.118e-02 ## internet -1.448e-02 8.110e-02 5.580e-02 ## mentalHealth 1.949e+00 5.926e-02 1.000e+00 ## extracurricular -6.889e-04 6.365e-02 3.071e-02

##Best model

best <- **which.max**(basModel**$**logmarg)bestmodel <- basModel**$**which[[best]]bestmodel

## [1] 0 3 4 5 7 8 10 13

bestGamma <- **rep**(0,basModel**$**n.vars)bestGamma[bestmodel **+** 1] <- 1bestGamma

## [1] 1 0 0 1 1 1 0 1 1 0 1 0 0 1 0

##Fit the best BIC model by imposing which variables to be used using the indicators.

bas\_bestmodel <- **bas.lm**(examScore **~** studyHours**+**socialMediaHours**+**netflixHours**+**attendance**+**sleepHours**+**exerciseFreq**+**mentalHealth , data = HabitsPerformanceData, prior = "BIC", n.models = 1, bestmodel = bestGamma, modelprior = **uniform**())

*Coefficient Estimates Under Reference Prior for Best BIC model*

##Retreat coefficients information.

score.coeff <- **coef**(bas\_bestmodel)

##Retreat bounds of credible intervals.

out <- **confint**(score.coeff)[,1**:**2]

##Combine results and construct summary table.

basSummary <- **cbind**(score.coeff**$**postmean , score.coeff**$**postsd , out) names <- **c**("post mean" , "post sd" , **colnames**(out))**colnames**(basSummary) <- namesbasSummary

## post mean post sd 2.5% 97.5%## Intercept 69.6015000 0.49108007 68.63782852 70.565171## studyHours 0.0000000 0.00000000 0.00000000 0.000000## socialMediaHours 0.0000000 0.00000000 0.00000000 0.000000## netflixHours -2.7346034 0.45701305 -3.63142341 -1.837783## attendance 0.1687005 0.05228704 0.06609498 0.271306## sleepHours 1.6848516 0.40067790 0.89858096 2.471122## exerciseFreq 0.0000000 0.00000000 0.00000000 0.000000## mentalHealth 1.9304120 0.17258549 1.59173873 2.269085

*Calculating Posterior Probability*

##Use ‘bas.lm’ for regression

basModel <- **bas.lm**(examScore **~** studyHours **+** socialMediaHours **+** netflixHours **+** attendance **+** sleepHours **+** exerciseFreq **+** mentalHealth , data = HabitsPerformanceData , prior = "BIC" , modelprior = **uniform**())**round**(**summary**(basModel) , 3)

## P(B != 0 | Y) model 1 model 2 model 3 model 4 model 5## Intercept 1 1.000 1.000 1.000 1.00 1.000## studyHours 1 1.000 1.000 1.000 1.00 1.000## socialMediaHours 1 1.000 1.000 1.000 1.00 1.000## netflixHours 1 1.000 1.000 0.000 1.00 0.000## attendance 1 1.000 0.000 1.000 1.00 0.000## sleepHours 1 1.000 1.000 1.000 0.00 1.000## exerciseFreq 1 1.000 1.000 1.000 1.00 1.000## mentalHealth 1 1.000 1.000 1.000 1.00 1.000## BF NA 1.000 0.000 0.000 0.00 0.000## PostProbs NA 1.000 0.000 0.000 0.00 0.000## R2 NA 0.901 0.895 0.880 0.88 0.874## dim NA 8.000 7.000 7.000 7.00 6.000## logmarg NA -5150.969 -5179.188 -5243.709 -5244.40 -5266.677

*The marginal posterior inclusion probability (pip)*

**print**(basModel)

*##*   
*## Call:*  
*## bas.lm(formula = examScore ~ studyHours + socialMediaHours +*   
*## netflixHours + attendance + sleepHours + exerciseFreq + mentalHealth,*   
*## data = HabitsPerformanceData, prior = "BIC", modelprior = uniform())*  
*##*   
*##*   
*## Marginal Posterior Inclusion Probabilities:*   
*## Intercept studyHours socialMediaHours netflixHours*   
*## 1 1 1 1*   
*## attendance sleepHours exerciseFreq mentalHealth*   
*## 1 1*