Project Report - Analyzing Educational Data

Daniel Posmik, Aristofanis Rontogiannis, Jizhou Tian

2024-10-24

Table of contents

Introduction (Daniel)

For this project, we will be analyzing educational data. We are interested in understanding the predictors of student performance as measured by exam scores. We will be using a publicly available dataset from Kaggle that contains information about students and their exam scores.

Table 1: Variable Summary for the Educational Data

Variable Name	Variable Type	Description
Hours_Studied	numeric	Hours Studied
Attendance	numeric	Attendance
Parental_Involvement	factor	Parental Involvement
Access_to_Resources	factor	Access to Resources
Extracurricular_Activities	factor	Extracurricular Activities
Sleep_Hours	numeric	Sleep Hours
Previous_Scores	numeric	Previous Scores
Motivation_Level	factor	Motivation Level

Variable Name	Variable Type	Description
Internet_Access	factor	Internet Access
Tutoring_Sessions	numeric	Tutoring Sessions
Family_Income	factor	Family Income
Teacher_Quality	factor	Teacher Quality
School_Type	factor	School Type
Peer_Influence	factor	Peer Influence
Physical_Activity	numeric	Physical Activity
Learning_Disabilities	factor	Learning Disability
Parental_Education_Level	factor	Parental Education Level
Distance_from_Home	factor	Distance from Home
Gender	factor	Gender
Exam_Score	numeric	Exam Score

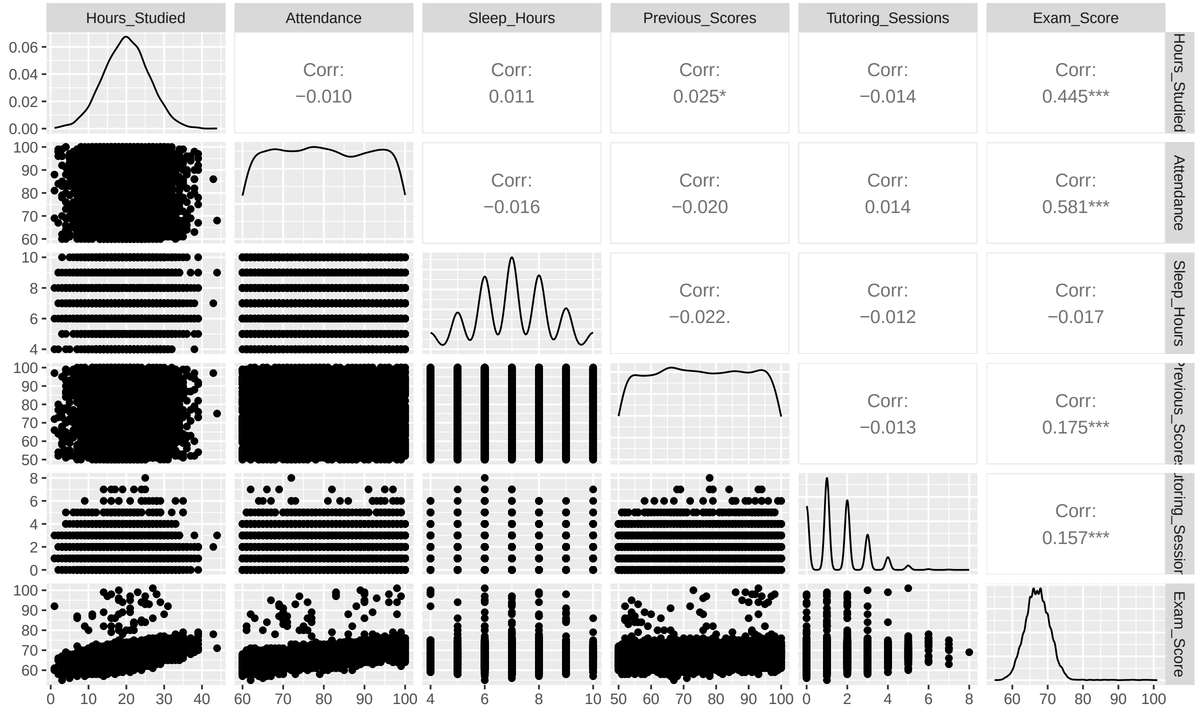
Now, we want to further explore a specific hypothesis about a subset of predictor variables. Suppose we maintain that the following variables are significant predictors:

- Hours Studied
- Attendance
- Sleep Hours
- Previous Scores
- Tutoring Sessions

We can formalize this question as follows:

- $H_0 : \left[1_{i=0}, \quad \cdots \quad 1_{i=p+1}, \quad 0_{i=p+2}, \quad \cdots \quad 0_{i=P},\right] \cdot \quad = \beta_0 + \cdots + \beta_{(p+1)} = 0$
- $H_A : \{\beta_1 \neq 0\} \cap \cdots \cap \{\beta_5 \neq 0\}$

Before we begin our analysis, let us take a look at the dependencies across these data:



Part 1: Linear Regression Analysis (Daniel)

The Least Squares Estimators

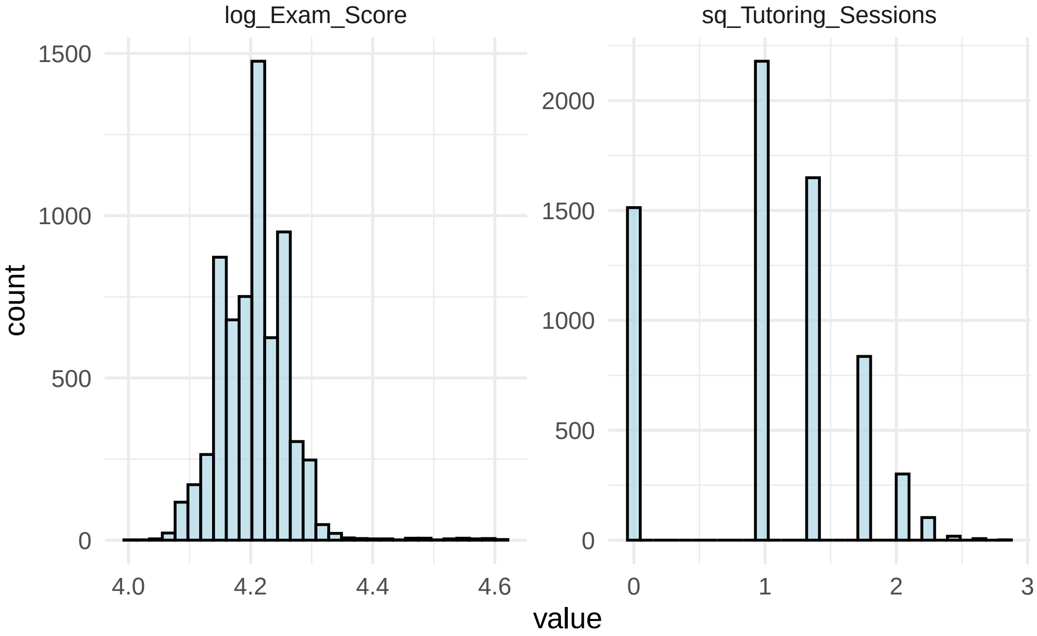
Let us begin by discussing the assumptions of linear regression model. In a Gauss-Markov setting, we assume that our linear model is of the form:

$$\mathbf{Y} = \mathbf{X}\beta + \epsilon, \quad \epsilon \sim N(0, \sigma^2 I)$$

where $\mathbb{E}[\epsilon] = 0$ and $\text{Var}[\epsilon] = \sigma^2 I$ denote the zero-mean and constant variance assumptions. In our case, we begin with $p = 5$, i.e. our design matrix has $p + 1$ columns, accounting for the intercept term. Then, we can write the model as matrices:

That being said, what the Gauss-Markov model boasts in theoretical simplicity, it often lacks in practical validity. If we refer to the exploratory analysis above, we can see that the assumptions may not hold. For one, we have one predictor variable, `Tutoring_Sessions`, and our dependent variable, `Exam_Score`, that are right-skewed. This violates the assumption of normality. Moreover, the Gauss-Markov model assumes constant variance with zeros on the off-diagonal elements of the covariance matrix. In practice, this is an assumption that is frequently violated. Interestingly, in our case the correlation between our predictor variables is indeed close to 0. If we had more substantial correlations on the off-diagonal elements, we could have solved our estimation problem with the generalized least squares estimator.

In our case, we will rememedy the normality assumption by transforming our data. We will use logaritmic transformations on the Exam score variable to achieve a normal distribution (`log_Exam_Score`) and a square root transformation of the Tutoring sessions variable to achieve a distribution that more closely resembles a normal distribution (`log_Tutoring_Sessions`). We chose the square root transformation for the tutoring sessions variable because it contains a lot of 0s, making the logarithmic transformation less suitable.



$$Y = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} = \begin{bmatrix} 1 & X_{12} & X_{13} & \cdots & X_{1(p+1)} \\ 1 & X_{22} & X_{23} & \cdots & X_{2(p+1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & X_{n2} & X_{n3} & \cdots & X_{n(p+1)} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix}$$

Note that we are using the generalized matrix inverse in case the design matrix is not of full rank. The canonical matrix inverse of the form X^{-1} exists iff X is of full rank. Next, we can solve for $\widehat{\beta}$ via the normal equations:

$$
\begin{aligned}
\hat{\beta} =& (X^T X)^g X^T Y \\
=& \left(\begin{bmatrix} 1 & 1 & \cdots & 1 \\ X_{12} & X_{22} & \cdots & X_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ X_{1(p+1)} & X_{2(p+1)} & \cdots & X_{n(p+1)} \end{bmatrix} \begin{bmatrix} 1 & X_{12} & \cdots & X_{1(p+1)} \\ 1 & X_{22} & \cdots & X_{2(p+1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & X_{n2} & \cdots & X_{n(p+1)} \end{bmatrix} \right)^g \cdot \\
& \begin{bmatrix} 1 & 1 & \cdots & 1 \\ X_{12} & X_{22} & \cdots & X_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ X_{1(p+1)} & X_{2(p+1)} & \cdots & X_{n(p+1)} \end{bmatrix} \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix}
\end{aligned}
$$

Using the R `lm()` function, we can estimate the coefficients of the linear model:

```
Call:
lm(formula = log_Exam_Score ~ Hours_Studied + Attendance + Sleep_Hours + 
    Previous_Scores + sq_Tutoring_Sessions, data = educ_dta)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.08391 -0.01675 -0.00173  0.01336  0.37210 

Coefficients:
                       Estimate Std. Error t value Pr(>|t|)    
(Intercept)           3.818e+00  4.460e-03 856.076   <2e-16 ***
Hours_Studied         4.350e-03  6.916e-05  62.899   <2e-16 ***
Attendance            2.951e-03  3.588e-05  82.256   <2e-16 ***
Sleep_Hours          -2.646e-04  2.822e-04  -0.938    0.348    
Previous_Scores       7.146e-04  2.878e-05  24.827   <2e-16 ***
sq_Tutoring_Sessions  1.339e-02  6.417e-04  20.865   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.03366 on 6601 degrees of freedom
Multiple R-squared:  0.6387,	Adjusted R-squared:  0.6385 
F-statistic:  2334 on 5 and 6601 DF,  p-value: < 2.2e-16
```

We can see that all variables except for `Sleep_Hours` are significant predictors of exam scores, even at a 1% significance level. So what does this tell us about our hypothesis? We will further examine this question in the next subsection.

Hypothesis Testing

Our estimation question is essentially a hypothesis testing problem. Namely, we want to test whether a subset of our predictors (see above) is jointly significant in the prediction of exam scores. We can formalize this as a hypothesis test by introducing the scalar vector K such that:

$$K^T\beta = \begin{bmatrix}1_{i=0} & \dots & 1_{i=p+1} & 0_{i=p+2} & \dots & 0_{i=P}\end{bmatrix} \begin{bmatrix}\beta_0 \\ \vdots \\ \beta_{p+1} \\ \beta_{p+2} \\ \vdots \\ \beta_P\end{bmatrix} = \begin{bmatrix}\beta_0 \\ \vdots \\ \beta_{p+1}\end{bmatrix} = M_{1,(p+1)}$$

where $\{\beta_0, \dots, \beta_{p+1}\}$ are the coefficients of the predictors we are interested in and $\{\beta_{p+2}, \dots, \beta_{P^*}\}$ are the coefficients of the remaining predictors, with $p \leq P$.

We call $\beta^T M$ estimable if $K_i \in \mathbb{C}(X^T)$ or in other words $\exists\ A$ s.t. $X^T A = K^T$, i.e. the rows of K are linearly dependent on the rows of X.

Since $K_I^T\beta$ is estimable, its BLUE is given by:

$$K^T\hat{\beta} = K_i^T(X^T X)^g X^T Y$$

In our case, we can test our hypothesis with the help of a F-test. **How do we arrive at F test statistic?????**

Part 2: Principal Component Analysis (Aristofanis)

Part 3: Non-linear Regression Analysis (Jizhou)

Conclusion

Code Appendix

```
# Set up knit environment
knitr::opts_chunk$set(echo = F)
knitr::opts_chunk$set(error = F)
knitr::opts_chunk$set(warning = F)
knitr::opts_chunk$set(message = F)

# Load necessary packages
library(tidyverse)
library(kableExtra)
library(knitr)
library(broom)
library(ggplot2)
library(naniar)
library(gtsummary)
library(GGally)

# Load the data
educ_dta <- read_csv("student_performance.csv") %>%
  mutate(
    Parental_Involvement = as.factor(Parental_Involvement),
    Access_to_Resources = as.factor(Access_to_Resources),
    Extracurricular_Activities = as.factor(Extracurricular_Activities),
    Motivation_Level = as.factor(Motivation_Level),
    Internet_Access = as.factor(Internet_Access),
    Family_Income = as.factor(Family_Income),
    Teacher_Quality = as.factor(Teacher_Quality),
    School_Type = as.factor(School_Type),
    Peer_Influence = as.factor(Peer_Influence),
    Learning_Disabilities = as.factor(Learning_Disabilities),
    Parental_Education_Level = as.factor(Parental_Education_Level),
    Distance_from_Home = as.factor(Distance_from_Home),
    Gender = as.factor(Gender)
  )

# Summary table
table_summary <- tibble(
  "Variable Name" = colnames(educ_dta),
  "Variable Type" = sapply(educ_dta, class),
  "Description" = c(
    "Hours Studied", #1
    "Attendance", #2
    "Parental Involvement", #3
    "Access to Resources", #4
    "Extracurricular Activities", #5
    "Sleep Hours", #6
    "Previous Scores", #7
    "Motivation Level", #8
    "Internet Access", #9
    "Tutoring Sessions", #10
    "Family Income", #11
    "Teacher Quality", #12
    "School Type", #13
    "Peer Influence", #14
    "Physical Activity", #15
    "Learning Disability", #16
    "Parental Education Level", #17
    "Distance from Home", #18
    "Gender", #19
    "Exam Score" #20
  )
)

# Display the table
knitr::kable(table_summary,
  caption = "Variable Summary for the Educational Data")

pred_var <- c("Hours_Studied",
              "Attendance",
              "Sleep_Hours",
              "Previous_Scores",
              "Tutoring_Sessions")

pred_dta <- educ_dta %>%
  dplyr::select(all_of(pred_var), "Exam_Score")

GGally::ggpairs(pred_dta)
# Log-transform skewed variables
educ_dta <- educ_dta %>%
  mutate(
    log_Exam_Score = ifelse(Exam_Score == 0, 0, log(Exam_Score)),
    sq_Tutoring_Sessions = sqrt(Tutoring_Sessions)
    )
```

```r
# Display histograms next to each other
educ_dta %>%
  dplyr::select(log_Exam_Score, sq_Tutoring_Sessions) %>%
  gather() %>%
  ggplot(aes(value)) +
  geom_histogram(bins = 30, color = "black",
    fill = "lightblue", alpha = 0.7) +
  facet_wrap(~key, scales = "free") +
  theme_minimal()
# Fit the linear model
lm_model <-
  lm(log_Exam_Score ~ Hours_Studied + Attendance +
                      Sleep_Hours + Previous_Scores + sq_Tutoring_Sessions,
                      data = educ_dta)

# Summary
summary(lm_model)
```