MKT Faculty Ratings Analysis

Ethan Chapman, Michael Okanta

2024-10-08

Executive Summary

This report investigates the relationships between professor performance, student behaviors, and overall student evaluations in marketing courses at Miami University. A mixed-effects model was used to analyze the influence of both instructor and student metrics on overall course ratings. Forward selection was employed to optimize the model, and multicollinearity was assessed using variance inflation factors (VIF). The analysis revealed that instructor preparedness, the challenge presented by the instructor, and demonstrating concern for student learning were significant predictors of course evaluations. However, strong correlation between the predictors make it difficult to separate out the effects of certain questions with a high degree of confidence.

Introduction

Understanding the drivers of student evaluations is critical for improving the educational experience and enhancing teaching effectiveness. This study explores the relationship between various instructor and student behaviors and overall course evaluations. The study addresses the following research questions:

Our research questions are:

1. What are the effects of professor performance measures on student evaluations of the instructor?

a. What implicit weights do students put on individual measures when making overall rating?

2. What are the effects of student behavior measures on their overall evaluation of the instructor?

a. To what extent and in what ways do student behaviors influence professor evaluations (or vice versa)?

3. Do these findings vary by the type of course?

a. How does the type of course affect the relationships found in 1 and 2?

The primary aim is to identify actionable insights for improving student ratings of courses, while accounting for variability between instructors and course types.

Data

The dataset includes student evaluations of marketing courses from 2013 to 2017 at Miami University. Each observation in the dataset corresponds to a section of a course taught in a particular semester. There were 587 observations collected during this period with key variables including:

- **Instructor Behaviors (i-variables):** Responses to questions regarding instructor behavior and performance. They were posed on a 0-4 Likert scale and averaged into a section mean in this dataset.

- **Student Behaviors (s-variables):** Responses to questions regarding student behavior. They were posed on a 0-4 Likert scale and averaged into a section mean in this dataset.

- **InstID**: Unique, anonymized identifier for each instructor

- **CourseType**: Categorical variable representing different types of courses. There are 4 categories of courses used: FSB Core, MKT Core, MKT Elective, and MKT Capstone.

- **Course Term**: The year and semester of the section.

- **Enrolled**: Number of students enrolled in the section.

- **Completed**: Number of students who completed the survey for that section.

We added another variable **responseRate**, which is the proportion of students who completed the evaluation for each course. Higher response rates might reflect positively on the instructor.

Exploratory Analysis

The dataset was cleaned by removing rows where no students completed the course evaluations. There were only two of these observations, which provided no useful information. Basic variable manipulation was done to prepare for statistical analysis.

A correlation matrix weighted by respondents, shown in Figure 1, was computed to explore relationships between student and instructor metrics. The matrix revealed strong correlations between student metrics as well as between instructor metrics. This shows that sections with students who rate themselves highly in one area tend to also mark themselves highly in other areas. Similarly, highly-rated instructors will tend to be highly-rated on all metrics, and vice versa. However, student and instructor metrics were only weakly correlated, suggesting that student and instructor operate somewhat independently. However, three of the student questions do exhibit correlation with instructor questions. Finally, the response rate has only small positive correlation with instructor metrics.

Results

Full Mixed-Effects Model

We normalized the predictors to have equal mean and standard deviation before running the model, which allows us to directly compare the coefficients of the predictors in the model as a measure of predictor strength. We fit a linear mixed-effects model to account for two important sources of variation in the model. First, as we are measuring the performance of the instructors, the instructor themselves will play an important role in the result, which may not be able to be accounted for with a simple adjustment to the intercept that a fixed effect would impose. Similarly, we want to explore if there is an effect in the course type more broadly than a fixed effect, as in, whether certain variables become more important for different course
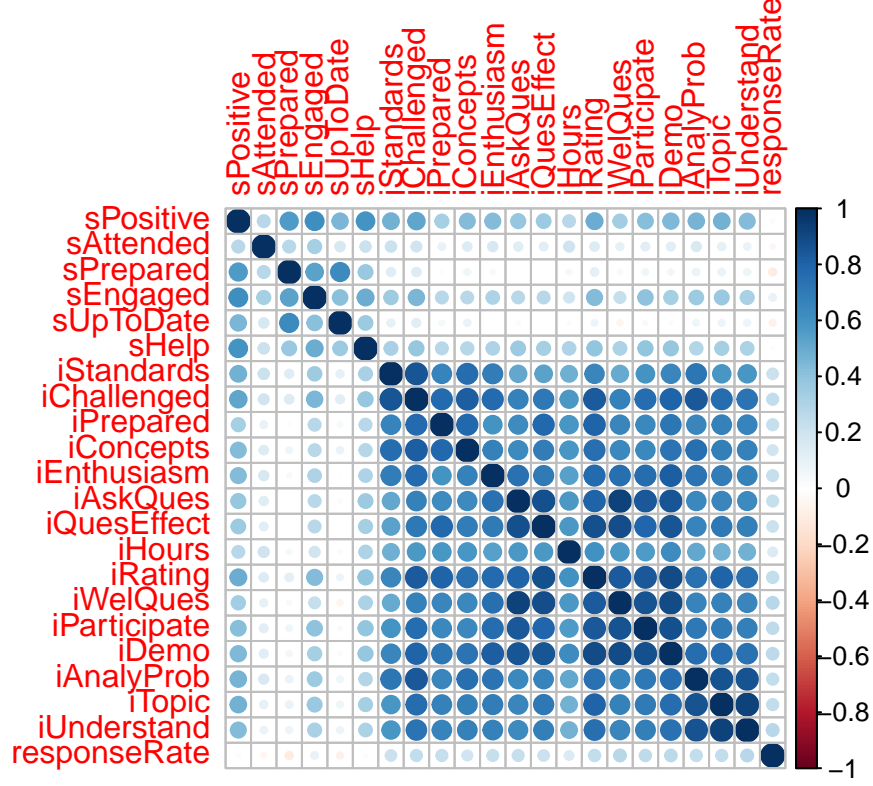
Figure 1: Weighted correlation matrix of predictors

types. Setting these two predictors to be random effects allows us to capture their variability more accurately. Additionally, it allows us to generalize our results across instructors and course types. Therefore, using a mixed-effects model with these as random effects and the questions and response rate as fixed effects is a straightforward choice of model. We also incorporated weights based on the number of respondents as a way to give more influence to sections with higher response rates, which provide greater confidence in their feedback. The top 4 predictors in this model were **iTopic, iDemo, iChallenged**, and **iPrepared**, with the other predictors significantly behind. Running the ANOVA on the mixed-effects model helped identify which predictors have significant effects on the overall instructor ratings. The predictors **sEngaged, iChallenged, iPrepared, iConcepts, iEnthusiasm, iQuesEffect, iWelQues, iParticipate, iDemo, iAnalyProb, iTopic,** and **iUnderstand** appeared to be statistically significant, with p-values below 0.05. However, with so many predictors, multicolinearity is a major concern. The multicolinearity effect, where two predictors are strongly correlated with each other, makes it more difficult for a model to distinguish which variable is causing an effect. This effect makes p-values an unreliable method of measuring variable importance. Multicolinearity can be assessed using Variance Inflation Factors (VIF), which measure the multicolinearity of a variable contextually in the model. The largest VIF in the model was for **iUnderstand** at 8.04, which is sufficiently large to warrant concern. One way to reduce multicolinearity is to reduce the number of other predictors that one can be correlated with, so we explored variable reduction techniques to help address this concern.

We then visualized the model's fit using diagnostic plots to assess whether the model assumptions hold. Importantly, the diagnostics plot does not suggest distributional errors. We previously wondered if the distributional assumption of normally distributed errors was reasonable for this model, as the actual responses are bounded between 0 and 4. However, the residuals do not suggest a need to switch to a different model response.

Forward Selection

We used forward selection based on AIC to help address multicolinearity, as well as generally help determine which predictors were significant. We did not include the instructor ID in selection, as not accounting for it in the model would cause signficant concern about the independence of observations from the same instructor. Furthermore, testing backwards selection found that removing the instructor IDs effect incurred a large AIC penalty. The forward selection did not use the course type as a random effect, or 13 of our fixed effects, including each student metric except for **sEngaged**, the **responseRate**, and several instructor metrics. As such, we can assume that they are not useful predictors of the instructor rating. The coefficients for the fixed effects in the final model are shown in Table 1. Again, **iDemo, iTopic, iChallenged, and iPrepared** are the strongest predictors. These are the same top four predictors as in our full model, which suggests a degree of robustness in the results. However, we also note a fairly strong negative coefficient for **iAnalyProb**.

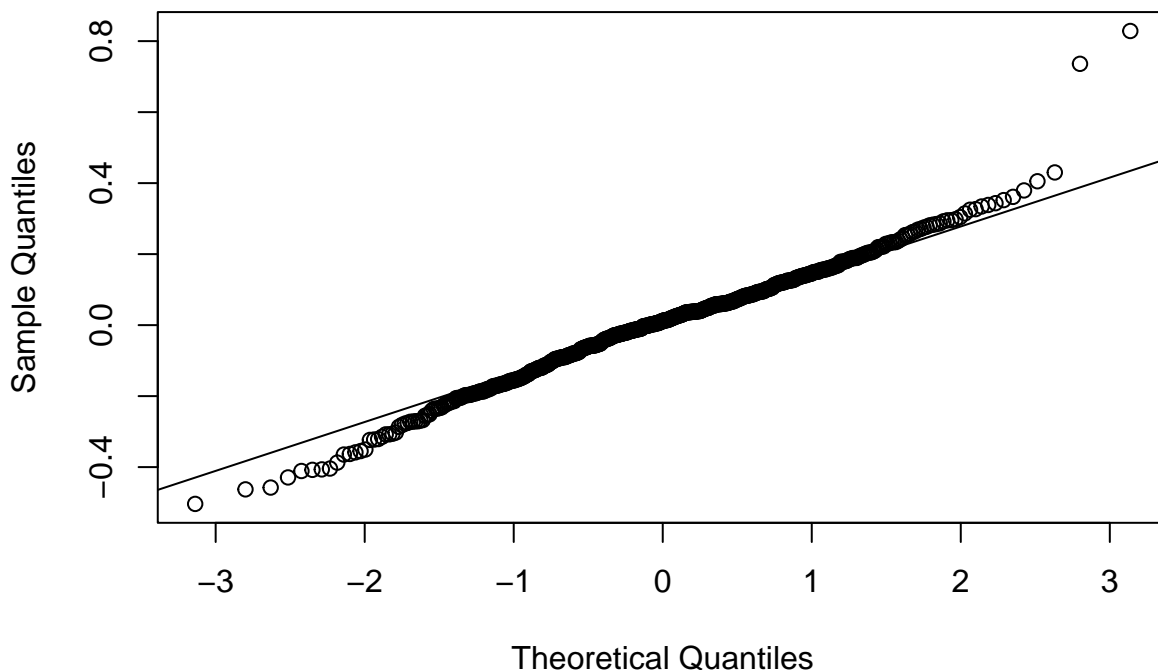|	Coefficient
(Intercept)	3.192
iDemo	0.130
iTopic	0.129
iChallenged	0.117
iPrepared	0.097
iQuesEffect	0.083
iWelQues	0.082
sEngaged	0.049
iAnalyProb	-0.097

Normal Q−Q Plot

Figure 2: Q-Q plot of model residuals

As shown in Figure 2, the normal error assumption of our model is reasonable, as the theoretical quantiles and observed quantiles generally line up quite well. There is a suggestion of two outliers, but their degree is

small and manual inspection shows no data issues with those points. The residual plots also show no signs of heteroscedasticity or violations of normality. Additionally, the VIF results show that the maximum VIF has decreased to 5.76 for **iDemo**, suggesting that multicolinearity is not a major issue and generally improved over our full model. However, it likely still has an impact on the model.

Summary of Model Results

- **iChallenged**, **iDemo**, **iPrepared**, and **iTopic** emerged as the strongest predictors of overall student evaluations.

- Multicolinearity was acceptable, but may impact the final results. Other diagnostics confirmed that the model's assumptions held, supporting the robustness of the findings.

Conclusion

The final model shows that the most significant predictors of overall student evaluations are certain key instructor behaviors. The top four predictors in the final model were also the top in the full model, increasing our confidence in the strength of these questions. They were:

- **iDemo**: "My instructor demonstrated concern for student learning."

- **iTopic**: "My appreciation for this topic has increased as a result of this course."

- **iChallenged**: "The instructor effectively challenged me to think and to learn."

- **iPrepared**: "My instructor offered opportunities for active participation to understand course content."

Student engagement, represented by **sEngaged**, has a positive but smaller impact on overall ratings compared to instructor behaviors. No other student behaviors were present in the final model. Although student preparedness and participation are important for the learning environment, the results suggest that they have less direct influence on how students rate their instructors. This reinforces the idea that students' evaluations are more strongly shaped by instructor behavior rather than their own engagement levels.

The instructor metric **iAnalyProb** has a negative coefficient, suggesting that, after accounting for the other questions, it negatively impacts an instructor's rating. It is possible that the question ("In this course I learned to analyze complex problems or think about complex issues") is used as a proxy for indicating a difficult course, which may result from unnecessarily difficult instruction rather than challenging topics or material. Alternatively, students may tend to dislike more difficult courses in general, and rate their instructors lower because of it.

The random effect for **CourseType** was not found to significantly affect the relationship between the fixed effects and the overall rating. This indicates that the impact of instructor behaviors is consistent across different types of courses. Hence, course type does not meaningfully influence the factors that drive overall student evaluations in the context of marketing courses.

Multicollinearity was assessed using variance inflation factors (VIFs), with all values falling below 10, confirming that multicollinearity did not significantly impact the interpretation of the model. Forward selection helped reduce any potential multicollinearity by removing variables that did not contribute meaningfully to the model. The final model includes only those variables that significantly impact student evaluations while maintaining interpretability and robustness.

The positive coefficients for **iWelQues** and **iQuesEffect** indicate that students appreciate instructors who welcome questions and engage effectively with student inquiries. Interestingly, although their effect on the model is very similar, the forward selection process found it meaningful to include both as predictors in the model.

Implications and Recommendations

Based on the final model, it is recommended that instructors focus on the following areas to improve their overall evaluations:

1. **Demonstrating concern for student learning:** This factor was the overall most important in the model. The added importance of welcoming and answering questions suggests one way that instructors can demonstrate concern for student learning.

2. **Challenging students intellectually:** Encouraging critical thinking and providing challenging material can help improve instructor rating. Additionally, they can help keep students engaged, another key factor in instructor ratings.

3. **Being well-prepared for class:** Organizational and preparedness skills are important for creating a positive learning experience.

4. **Drive student appreciation:** Students growing to appreciate a topic better reflects positively on their instructor's ratings.

These strategies can be applied across various course types, as the model indicates that these behaviors are consistent drivers of positive student evaluations in marketing courses.

Bibliography

Brown, Christopher. 2018. *Formula.tools: Programmatic Utilities for Manipulating Formulas, Expressions, Calls, Assignments and Other r Objects.* https://CRAN.R-project.org/package=formula.tools.

Clark, Michael. 2022. *Mixed Models with r.* https://m-clark.github.io/mixed-models-with-R/.

Fox, John, and Sanford Weisberg. 2019. *An r Companion to Applied Regression.* Third. Thousand Oaks CA: Sage. https://socialsciences.mcmaster.ca/jfox/Books/Companion/.

Kuznetsova, Alexandra, Per B. Brockhoff, and Rune H. B. Christensen. 2017. "lmerTest Package: Tests in Linear Mixed Effects Models." *Journal of Statistical Software* 82 (13): 1–26. https://doi.org/10.18637/jss.v082.i13.

Pasek, Josh, and Alex Tahk. 2021. *Weights: Weighting and Weighted Statistics.* https://CRAN.R-project.org/package=weights.

R Core Team. 2024. *R: A Language and Environment for Statistical Computing.* Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Voeten, Cesko C. 2023. *Buildmer: Stepwise Elimination and Term Reordering for Mixed-Effects Regression.* https://CRAN.R-project.org/package=buildmer.

Wei, Taiyun, and Viliam Simko. 2024. *R Package 'Corrplot': Visualization of a Correlation Matrix.* https://github.com/taiyun/corrplot.

Wickham, Hadley, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, et al. 2019. "Welcome to the Tidyverse." *Journal of Open Source Software* 4 (43): 1686. https://doi.org/10.21105/joss.01686.

Appendix: R Code

```r
library(tidyverse)
library(corrplot)
library(weights)
library(formula.tools)
library(lmerTest)
library(car)
library(buildmer)

# Data cleaning
data <- read_csv("CombinedData.csv") |>
  filter(Completed > 0) |>
  mutate(InstID = as.factor(InstID),
         CourseType = factor(CourseType,
                             levels=c("1", "2", "3", "4"),
                             labels=c("FSB Core", "MKT Core",
                                      "MKT Elective", "MKT Capstone")),
         responseRate = Completed/Enrolled)

# Weighted correlation plot
corrplot(wtd.cors(as.matrix(data[,4:27]), weight=data$Completed))

X <- data |>
  select(sPositive:responseRate) |>
  select(-iRating)

# PCA
pca <- prcomp(scale(X))

# Normalize data for regression
data_norm <- data |>
  mutate(sPositive=scale(sPositive),
         sAttended=scale(sAttended),
         sPrepared=scale(sPrepared),
         sEngaged=scale(sEngaged),
         sUpToDate=scale(sUpToDate),
         sHelp=scale(sHelp),
         iStandards=scale(iStandards),
         iChallenged=scale(iChallenged),
         iPrepared=scale(iPrepared),
         iConcepts=scale(iConcepts),
         iEnthusiasm=scale(iEnthusiasm),
         iAskQues=scale(iAskQues),
         iQuesEffect=scale(iQuesEffect),
         iHours=scale(iHours),
         iWelQues=scale(iWelQues),
         iParticipate=scale(iParticipate),
         iDemo=scale(iDemo),
         iAnalyProb=scale(iAnalyProb),
         iTopic=scale(iTopic),
         iUnderstand=scale(iUnderstand))
```

```r
# Full model
rand_int_model <- lmer(iRating ~ sPositive + sAttended + sPrepared + sEngaged +
                        sUpToDate + sHelp + iStandards + iChallenged + iPrepared +
                        iConcepts + iEnthusiasm + iAskQues + iQuesEffect + iHours +
                        iWelQues + iParticipate + iDemo + iAnalyProb + iTopic +
                        iUnderstand + responseRate + (1 | InstID) + (1 | CourseType),
                        weights=Completed, data=data_norm)
summary(rand_int_model)
# Model analysis
anova(rand_int_model)
vif(rand_int_model)
plot(rand_int_model)

# Forwards selection
curr_model <- formula(iRating ~ (1 | InstID))
full_model <- formula(iRating ~ sPositive + sAttended + sPrepared + sEngaged +
                        sUpToDate + sHelp + iStandards + iChallenged + iPrepared +
                        iConcepts + iEnthusiasm + iAskQues + iQuesEffect + iHours +
                        iWelQues + iParticipate + iDemo + iAnalyProb + iTopic +
                        iUnderstand + responseRate + (1 | InstID) + (1 | CourseType))
aic_val <- AIC(lmer(curr_model, weights=Completed, data=data_norm))
full_vars <- rhs.vars(full_model)
while(TRUE) {
  aics <- rep(NA, length=length(full_vars))
  # Determine AICs for all
  for(i in 1:length(rhs.vars(full_model))) {
    if(!(full_vars[i] %in% rhs.vars(curr_model))) {
      test_model <- add.terms(curr_model, full_vars[i])
      aics[i] <- AIC(lmer(test_model, weights=Completed, data=data_norm))
    }
  }
  if(min(aics, na.rm=TRUE) >= aic_val) {
    break
  }
  toAdd = which.min(aics)
  print(paste0("Adding ",full_vars[toAdd]," (AIC=",aics[toAdd],")"))
  curr_model <- add.terms(curr_model, full_vars[toAdd])
  aic_val <- AIC(lmer(curr_model, weights=Completed, data=data_norm))
}

forward_model <- lmer(curr_model, weights=Completed, data=data_norm)
summary(forward_model)
plot(forward_model)
vif(forward_model)

qqnorm(resid(forward_model))
qqline(resid(forward_model))
```