Analysis of FCQ Data

Alex Ojemann

2023-12-13

Introduction

For my final project, I explored CU FCQ data. The FCQ data set contains the average score for several questions regarding the effectiveness of a given instructor in a certain area on a discrete numeric scale. Before 2020, there were nine of these questions with values between 1 and 6, including an 'instructor overall' question. As of 2020 several more questions were added but the 'instructor overall' question was removed and the values are between 1 and 5.

One of the goals of my project is to determine the factors that are most related to each other and which best predict the instructor overall before 2020 and the degree to which the professor challenged students to develop knowledge and understanding since 2020. Of the questions in the survey since 2020, the degree to which the professor challenged students to develop knowledge and understanding makes the most sense as a response variable because it can be a product of the other variables which are more granular, such as providing feedback on students' work that helped them improve and considering diverse perspectives during class or in assignments. In addition, challenging students to develop knowledge and understanding is often the goal of a class. In addition to determining the factors that are most influential in prediction, I want to build models that obtain a high degree of predictive accuracy. This could be valuable in particular for the data before 2020 because building a predictive model with high accuracy on that data to predict the instructor overall variable could be useful to infer how an instructor is doing overall on the data since 2020, which does not contain an instructor overall variable. The third and final goal of the project is to explore the relationships between all the variables in the data since 2020 using a correlation matrix. This could reveal many new insights as the data since 2020 includes many different variables for which we may not understand how they're related. All of these directions of exploration are designed to help professors and educators in general better understand how students learn.

Methodology and Results

To begin, some exploratory data analysis and preprocessing was conducted on both data sets. Rows with NA values in any of the predictor or response variables were omitted. All variables that were based on student responses were analyzed to ensure that all of the values were within the specified range, which was between 1 and 6 for the data before 2020 and between 1 and 5 for the data since 2020. In addition, all features were standardized.

The methods used for prediction were a linear regression model with stepwise feature selection and a random forest. This provided a variety of techniques that allowed for varying degrees of complexity and interpretability. Random forests are more complex than linear regression but they can still provide feature importance values that allow for insight into the factors that best predict the response. Success was measured using R^2 because it allows for straightforward comparison between the differently scaled response variables and gives an absolute sense of how the model is performing relative to a model with no parameters.

For the data before 2020, the predictors to be considered for the models included all of the student repsonse questions that were on the scale of 1 to 6 and other than instructor overall along with the number of hours spent per week and the student response rate.

The linear regression model from the data before 2020 had the following formula:

$$\begin{aligned}\text{Instr} = 5.17041 \\ &+ 0.42936 \times \text{Effect} \\ &+ 0.12573 \times \text{Avail} \\ &+ 0.08202 \times \text{Respect} \\ &- 0.09861 \times \text{Interest} \\ &+ 0.15073 \times \text{Course} \\ &- 0.01558 \times \text{HrsPerWk} \\ &- 0.00586 \times \text{Resp Rate} \\ &- 0.00636 \times \text{Learned}\end{aligned}$$

Forward stepwise feature selection was performed using BIC as the criterion for adding a feature. The stepwise selection process resulted in all potential predictors other than 'Challenge' being included in the model. The R^2 value of this model was 0.887. All predictors were statistically significant at the 0.001 level. From this formula, we can gather information about how each predictor related to the instructor overall variable given the coefficients. We can see that the instructor effectiveness, instructor availability, instructor respect and course overall have positive relationships with the instructor overall score when all other variables in the model are held constant, while the prior interest in the course, number of hours per week spent on course material, response rate to the FCQ and how much the student felt they learned have negative relationships with the instructor overall score when all other variables in the model are held constant.

The random forest model was trained using 100 estimators. The R^2 of the model was 0.980. The features as ranked by feature importance are shown below. The measure of feature importance is called IncNodePurity, which expresses the change in the homogeneity of the of the groups created by the decision trees.

Table 1: Feature Importances Ranked by Increase in Node Purity

	Feature	IncNodePurity
7	Effect	21533.3391
6	Course	12752.7490
8	Avail	9552.2214
5	Learned	4002.2437
9	Respect	3382.0209
3	Interest	1442.1780
2	HrsPerWk	790.6565
1	Resp_Rate	752.1036
4	Challenge	698.8159

These feature importances are largely in accordance with the insights generated from the linear regression parameters. Instructor effectiveness and course overall are intuitively highly correlated with instructor overall. However, it is worth noting that availability was the next most important feature by a large margin and had a positive coefficient in the linear regression model, reflecting that being available to students is one of the most important aspects of instructor performance.

For the data since 2020, the predictors to be considered for the models included all of the student repsonse questions that were on the scale of 1 to 5 and other than the 'Challenge' variable along with the student response rate.

The linear regression model from the data before 2020 had the following formula:

$$\begin{aligned}\text{Challenge} = {} & 4.4932 \\ & + 0.1116 \times \text{Reflect} \\ & + 0.1031 \times \text{Creative} \\ & + 0.0389 \times \text{Questions} \\ & + 0.0667 \times \text{Eval} \\ & + 0.0623 \times \text{Respect} \\ & + 0.0498 \times \text{Feedback} \\ & + 0.0493 \times \text{Discuss} \\ & - 0.0474 \times \text{Diverse} \\ & + 0.0613 \times \text{Synth} \\ & + 0.0354 \times \text{Tech} \\ & + 0.0197 \times \text{Connect} \\ & - 0.0137 \times \text{Interact} \\ & - 0.0082 \times \text{Collab} \\ & + 0.0056 \times \text{Resp\_Rate}\end{aligned}$$

Forward stepwise feature selection was performed using BIC as the criterion for adding a feature. The stepwise selection process resulted in 14 of the possible predictors being included in the model. The R^2 value of this model was 0.763. All of these predictors were significant at the 0.001 level. We can see that most of these predictors have a positive relationship with the degree to which the course challenged students to develop knowledge and understanding, with the degree to which the professor encouraged students to reflect on what they were learning and the degree to which the professor gave assignments that required original or creative thinking having the strongest positive relationships. Interestingly, the degree to which the instructor considered diverse perspectives had the strongest negative relationship with the degree to which the course challenged students to develop knowledge and understanding of the predictors in the model.

The random forest model was trained using 100 estimators. The R^2 of the model was 0.957. The features as ranked by feature importance are shown below. The measure of feature importance is again IncNodePurity.

Table 2: Feature Importances in Random Forest Model

	Feature	IncNodePurity
3	Reflect	2468.3409
8	Synth	1667.1237
11	Creative	1589.4841
12	Discuss	816.4794
15	Questions	700.0985
13	Feedback	646.0412
7	Eval	632.1452
16	Tech	376.2196
10	Respect	299.8690
6	Contrib	254.0592
1	Resp_Rate	240.3149
14	Grading	222.4231
4	Connect	180.5287
9	Diverse	170.6143
5	Collab	162.6794
2	Interact	134.6221

These feature importances once again are in relative agreement with our insights generated from the linear regression model. However, one significant difference is that the degree to which the instructor encouraged students to evaluate arguments, evidence, assumptions, and conclusions appears to be more significant than in the linear model, suggesting that it could also be one of the most influential factors in predicting the the degree to which the course challenged students to develop knowledge and understanding along with the degree to which the professor encouraged students to reflect on what they were learning and the degree to which the professor gave assignments that required original or creative thinking.

Below is the correlation matrix of all the features from the data since 2020. It contains all of the predictors and the response variable.

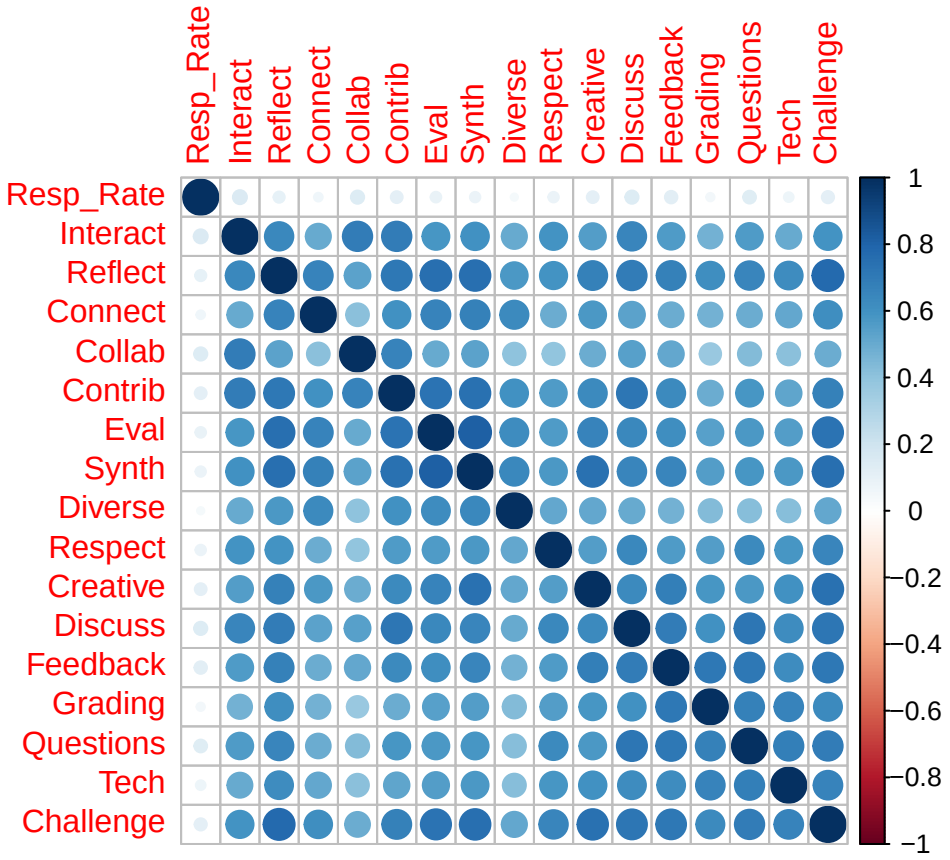

Table 3: Top 5 Most Correlated Feature Pairs

	Feature1	Feature2	Correlation
110	Synth	Eval	0.8177175
51	Challenge	Reflect	0.7741284
41	Eval	Reflect	0.7566999
136	Challenge	Synth	0.7517026
42	Synth	Reflect	0.7508692

Many interesting insights can be found in this matrix, as there are many other variables to consider other than the the degree to which the course challenged students to develop knowledge and understanding as was predicted by the prior models. Each of the top five pairs of features as shown contain synth, eval, reflect, and challenge, suggesting that those four features are highly interconnected.

Discussion and Future Directions

In summary, this project explored CU FCQ data from many different viewpoints, generated a number of useful insights into which factors best predict how effective an instructor is and the degree to which the course challenged students to develop knowledge and understanding, and allowed for further insight into how the many response questions since 2020 are related to one another.

One aspect of this report that could be extended is the models used for prediction. The original plan was to also train a feed forward neural network on both of the data sets, before and since 2020, but this was omitted due to the length of time required to train the neural network given the high volume of data. Neural networks are even more powerful than the methods used here and could obtain even higher predictive accuracy.

In addition, another potential direction for future students is to apply the models trained to predict instructor effectiveness on the data before 2020 to predict the effectiveness of instructors since 2020 since this question is no longer on the FCQ. This could have immense value in helping instructors determine how effective they have been.

Appendix

Code for models on 2010-2019 data:

```
# Load necessary libraries
library(MASS)
library(randomForest)
library(neuralnet)
library(caret)
library(readxl)

# Read the data (assuming the data is in a file named 'FCQ_Data.csv')
data <- read_excel("FCQ_Data_2010-2019.xlsx")

View(data)

# Standardize the predictor variables
standardized_data <- data
standardized_data[c("Resp Rate", "HrsPerWk", "Interest", "Challenge", "Learned", "Course", "Effect", "A

# Prepare data for modeling
predictors <- standardized_data[c("Resp Rate", "HrsPerWk", "Interest", "Challenge", "Learned", "Course"
response <- standardized_data[["Instr"]]
significant_vars <- c("Resp Rate", "HrsPerWk", "Interest", "Challenge", "Learned", "Course", "Effect",
standardized_data <- standardized_data[complete.cases(standardized_data[significant_vars]), ]


# Perform stepwise regression
standardized_data <- standardized_data[significant_vars]
full.model <- lm(Instr ~ ., data=standardized_data)
null.model <- lm(Instr ~ 1, data=standardized_data)
step.model <- step(null.model, scope=list(lower=null.model, upper=full.model), direction="both", k=log(
summary(step.model)

# Random Forest Model
#Space in a column name was causing an error in the random forest
names(standardized_data)[names(standardized_data) == "Resp Rate"] <- "Resp_Rate"
```

```
fit.rf <- randomForest(Instr ~ ., data=standardized_data, ntree=100)
summary(fit.rf)
importance(fit.rf)

# Model Evaluation: R^2 values
# For Linear Regression
r2.lm <- summary(step.model)$r.squared

# For Random Forest
rf.pred <- predict(fit.rf, standardized_data)
r2.rf <- cor(standardized_data$Instr, rf.pred)^2


# Print R^2 values
print(paste("R^2 for Linear Regression:", r2.lm))
print(paste("R^2 for Random Forest:", r2.rf))
```

Code for models on 2020-present data:

```
# Load necessary libraries
library(MASS)
library(randomForest)
library(neuralnet)
library(caret)
library(readxl)

# Read the data (assuming the data is in a file named 'FCQ_Data.xlsx')
data <- read_excel("FCQ_Data_Since_2020.xlsx")
names(data)[names(data) == "Resp Rate"] <- "Resp_Rate"


# List of predictors (excluding 'Challenge' as it is now the response)
predictors <- c("Resp_Rate", "Interact", "Reflect", "Connect", "Collab", "Contrib", "Eval",
                "Synth", "Diverse", "Respect", "Creative", "Discuss",
                "Feedback", "Grading", "Questions", "Tech")

# Standardize the predictor variables
standardized_data <- data
standardized_data[predictors] <- scale(data[predictors])

# Prepare data for modeling
response <- standardized_data[["Challenge"]]
significant_vars <- c(predictors, "Challenge")
standardized_data <- standardized_data[complete.cases(standardized_data[significant_vars]), ]

# Perform stepwise regression
standardized_data <- standardized_data[significant_vars]
full.model <- lm(Challenge ~ ., data=standardized_data)
null.model <- lm(Challenge ~ 1, data=standardized_data)
step.model <- step(null.model, scope=list(lower=null.model, upper=full.model), direction="both", k=log(
summary(step.model)

# Random Forest Model
```

```
fit.rf <- randomForest(Challenge ~ ., data=standardized_data, ntree=100)
summary(fit.rf)
importance(fit.rf)


# Model Evaluation: R^2 values
# For Linear Regression
r2.lm <- summary(step.model)$r.squared

# For Random Forest
rf.pred <- predict(fit.rf, standardized_data)
r2.rf <- cor(standardized_data$Challenge, rf.pred)^2


# Print R^2 values
print(paste("R^2 for Linear Regression:", r2.lm))
print(paste("R^2 for Random Forest:", r2.rf))
```

Code for correlation matrix and analysis of 2020-present data:

```
# Assuming 'standardized_data' is your data frame and contains all the predictors and the response vari

# List of predictors and the response variable
variables <- c("Resp_Rate", "Interact", "Reflect", "Connect", "Collab", "Contrib", "Eval",
               "Synth", "Diverse", "Respect", "Creative", "Discuss",
               "Feedback", "Grading", "Questions", "Tech", "Challenge")

# Calculate the correlation matrix
correlation_matrix <- cor(standardized_data[variables], use = "complete.obs")

# Display the correlation matrix
print(correlation_matrix)

# Optionally, you can also visualize this matrix using a package like 'corrplot'
if("corrplot" %in% rownames(installed.packages()) == FALSE) {
    install.packages("corrplot")
}
library(corrplot)
corrplot(correlation_matrix, method = "circle")


# Flatten the correlation matrix
cor_flat <- as.data.frame(as.table(correlation_matrix))

# Name the columns appropriately
names(cor_flat) <- c("Feature1", "Feature2", "Correlation")

# Remove self-correlations and duplicate pairs
cor_flat <- subset(cor_flat, Feature1 != Feature2)
cor_flat <- cor_flat[!duplicated(t(apply(cor_flat[,1:2], 1, sort))),]

# Sort by absolute correlation value
cor_flat <- cor_flat[order(-abs(cor_flat$Correlation)), ]
```

```r
# Select top 5 most correlated pairs
top_correlations <- head(cor_flat, 5)

kable(top_correlations, caption = "Top 5 Most Correlated Feature Pairs")
```