Class Seats Projection Modeling

Matthew Hefner and Julia Butler

February 26, 2019

Background

Projecting the class sizes for a semester given the data of previous semesters' class sizes presents several interesting challenges. The data reflect several patterns and phenomena; new courses are added, some courses are discontinued, and some are only offered in Fall or Spring semesters. To accurately predict class seats, a model must be developed that is capable of taking into account interactions between multiple courses over multiple semester transitions.

For the model developed in this paper, it is assumed that the conditions outside of this data to which the data are subject to remain constant; i.e., there are no significant changes in department or college policy outside of what is reflected within the data. A minimal class size must also be assumed, as a course below some minimum threshold will be unlikely to make the course schedule.

After a model is developed, it will be implemented in this paper on the course data for Appalachian State University's 1000-, 2000-, and 3000- level Mathematics courses.

Development

Let y_i^j represent the seats of class j in semester i. For simplification of notation, allow the previous semester's seats size to be denoted x_i^j. Explicitly put:

$$x_i^j = y_{i-1}^j$$

In the dataset provided, there are data for 8 semesters for 39 courses of Mathematics at the 1000, 2000, and 3000 course levels. Therefore, $i = 1...8$ and $j = 1...39$.

Because of the small number of observations, the relationship between courses' seat sizes and the previous semester's is assumed to be linear. Specifically, Let y_i^j be represented as a linear combination of x_i^k, $k = 1...39$ such that:

$$y_i^j = \sum_{k=1}^{39} \beta_k x_i^k$$

For any specific value y_i^j, a vector of the values of β_k, $\vec{\beta}$ may be determined such that there is no error. This, however, is not the objective of the model. Instead, the objective is to determine some $\vec{\beta}$ that minimizes the error vector $\vec{\epsilon}$ in the equation

$$\vec{y^j} = \mathbf{X}\vec{\beta} + \vec{\epsilon}$$

where:

$\vec{y^j}$ is the vector containing the class seat values of class j...

$$\vec{y^j} = \begin{bmatrix} y_1^j \\ y_2^j \\ \dots \\ y_8^j \end{bmatrix}$$

and $\mathbf{X}$ is a matrix containing all previous class seat sizes...

$$\mathbf{X} = \begin{bmatrix} x_1^1 & x_1^2 & \dots & x_1^{39} \\ x_2^1 & x_2^2 & \dots & x_2^{39} \\ \dots & & & \\ x_8^1 & x_8^2 & \dots & x_8^{39} \end{bmatrix}$$

This process should then be repeated for each of the 39 courses.

The solution to minimizing $\vec{\epsilon}$ for all of these error equations is to do exactly 39 multivariate linear regression models for each. The issue that arises, however, is that, given 39 courses and 8 observations, the system $\vec{y^j} = \mathbf{X}\vec{\beta}$ has infinitely many solutions $\vec{\beta}$. There are simply too many variables (courses) and not enough data to fit the data to a linear regression model by taking into account every course from the previous years.

Forward selection of courses is the remedy. To begin, a given course's model is null - there are no independent variables. Then, models are developed that add exactly one independent/predictor course. The quality of each model is then compared using a (generalized) Akaike Information Criterion (AIC) algorithm. This process is iterated, selecting exactly 6 courses that the AIC algorithm deems to be the most influential and developing associated coefficients (β) for each. Other, more hands-on approaches to the process of forward selection of courses are possible, but given the size of the dataset and the number of models, AIC is sufficient.

Once $\vec{\beta}$ has been determined, a given vector of 39 class seat values for a semester may be multiplied by the $\vec{\beta}$ of the model of a particular course to obtain the project class seats of that course for the next semester. Additionally, it is assumed that:

- The class seat size is the floor of the real number returned, and
- For a class to run, there must be at least 5 students in that class.

Results

Performing the above steps on the data given returns the projected class seat values for Fall of 2019 included in Appendix B. The code used to implement the model in R is included in Appendix A.

Discussion

Interestingly, in the cases of MAT - 1005, 2310, 3010, 3015, 3310, 3330, 3340, 3350, 3541, 3910 and 3920, the model was able to accurately predict a class seat size of 0 without explicit knowledge of these courses only being offered every-other semester.

Almost every number seems entirely reasonable, with the exception of MAT-3540. This course has clearly been discontinued or simply not been offered recently; however, the model predicts a class seat size of 25. This demonstrates the limitation of having "assumed that the conditions outside of this data to which the data are subject to remain constant." In the future, incorporating new data into the equation as predictors could improve these projections.

Additionally, a model with more semesters of data would likely provide better β coefficients. Given a dataset which contains *many* semesters of data, it is perhaps best to develop a model based on neural networks and machine learning to oust the assumption made that the relationship between a course and previous semesters is linear.

References

Arnholt, Alan. *Misc Regression*. 9 Oct. 2018, https://alanarnholt.github.io/STT3850/Rmarkdown/MiscRegression.html. Accessed 26 Feb. 2019.

Venables, W. N. and Ripley, B. D. *Modern Applied Statistics with S*. 4th ed., Springer, 2002.

Appendix A

```
library(dplyr)
library(ggplot2)
library(moderndive)
library(MASS)

#load dataset csv
Course_Sizes
<- read_csv("D:/Schoolwork/Mathematical Models/Project 3/Course Size Estimates.csv")

#preserve original, use only MAT 1, 2, and 3 thousand level courses
data <- Course_Sizes %>% slice(c(1:9, 12:17, 20:43))

#first remember the names
n <- data$X1

#transpose all but the first column (which is the names)
data <- as.data.frame(t(data[,-1]))
colnames(data) <- n

#Create dataframe of only actual class seat numbers
r <- rownames(data)
actual <- data %>% slice(c(1,2,3,4,5,6,8,10))
rownames(actual) <- c(r[1:6], "Fall_2018", "Spring_2019")

#create future cols
future <- actual
colnames(future) <- paste("Future_", colnames(actual), sep = "")
future <- future %>% slice(2:8)

#create training data set
train <- actual %>% slice(1:7)
#train <- bind_cols(train_actual, future)

#create predict data set
new_Values <- actual %>% slice(8)

#create lm models for each course based on forward selection
class_models <- vector(mode="list", length=length(new_Values))
for (i in 1:length(new_Values)) {
  class_models[[i]] <- lm(future[[i]] ~ 1, data = train)
  class_models[[i]] <- stepAIC(class_models[[i]], scope = (~MAT_0010 + MAT_1005 + MAT_1010
+ MAT_1020 + MAT_1025 + MAT_1035 + MAT_1110 + MAT_1120 + MAT_1530 + MAT_2030 + MAT_2110
+ MAT_2130 + MAT_2240 + MAT_2310 + MAT_2510 + MAT_3010 + MAT_3015 + MAT_3030 + MAT_3110
+ MAT_3130 + MAT_3220 + MAT_3310 + MAT_3330 + MAT_3340 + MAT_3350 + MAT_3500
+ MAT_3500 + MAT_3500 + MAT_3500 + MAT_3510 + MAT_3520 + MAT_3530 + MAT_3530 + MAT_3540
+ MAT_3541 + MAT_3543 + MAT_3610 + MAT_3910 + MAT_3920),
direction = "forward", test = "F")
}

Projected_Fall_2019 <- numeric(length(new_Values))
```

```r
for (i in 1:length(new_Values)) {
  Projected_Fall_2019[i] <- predict(class_models[[i]], newdata = new_Values)
  if (Projected_Fall_2019[i] < 5) {
    Projected_Fall_2019[i] <- 0
  }
  Projected_Fall_2019[i] <- floor(Projected_Fall_2019[i])
}
pcs <- rbind(Projected_Fall_2019)
colnames(pcs) <- colnames(actual)
actual <- rbind(actual, pcs)
```

Appendix B

	Fall_2017	Spring_2018	Fall_2018	Spring_2019	Projected_Fall_2019
MAT_1005	0	75	0	75	0
MAT_1010	619	390	454	435	402
MAT_1020	468	288	523	261	390
MAT_1025	276	215	323	179	249
MAT_1035	572	472	518	485	585
MAT_1110	411	276	403	282	417
MAT_1120	149	203	157	208	156
MAT_1530	0	25	26	16	6
MAT_2030	94	112	94	102	105
MAT_2110	34	31	32	27	35
MAT_2130	77	60	71	52	60
MAT_2240	114	114	107	95	98
MAT_2310	34	0	33	0	31
MAT_2510	0	0	0	0	0
MAT_3010	19	0	11	0	14
MAT_3015	0	13	0	7	0
MAT_3030	0	0	16	24	10
MAT_3110	11	27	9	25	12
MAT_3130	41	26	35	30	48
MAT_3220	29	25	20	13	19
MAT_3310	28	0	22	0	21
MAT_3330	24	0	31	0	36
MAT_3340	0	22	0	18	0
MAT_3350	0	18	0	21	0
MAT_3500	0	0	3	0	0
MAT_3510	0	17	8	9	0
MAT_3520	8	11	8	7	6
MAT_3530	2	0	0	0	0
MAT_3540	0	0	0	0	25
MAT_3541	0	4	0	14	0
MAT_3543	0	0	0	4	0
MAT_3610	11	9	0	16	0
MAT_3910	12	0	20	0	26
MAT_3920	0	6	0	7	0