Who is Missing? Predicting Preschool Presence

Dustin Poon & Jerry Eiswerth

1. Introduction

Children's education is one of the most important aspects of life in modern society. Schools are an integral institution for the economy because they enable people to become productive citizens in our society. Most individuals in North America will spend at least 12 years of their lives in formal education settings, through standard elementary, middle, and high school. At the very least, school attendance is mandatory in most US states (National Center for Education Statistics, 2017) from ages six to 16. Preschool is a special type of optional education program, early childhood education, available to children ages three to five. These programs offer a chance for children to socialize and engage in a classroom setting before compulsory elementary school programs. Our study specifically focuses on children enrolled in Head Start preschool programs in Chicago. These programs are government subsidized and provide a variety of services including school lessons, free meals, and learning assessments (Office of Head Start, 2023).

Preschool in general has been found to greatly increase school outcomes for students in later years (Peisner-Feinberg, 2001). Of particular importance is the attendance rate for preschool students. When children attend preschool more often, they receive more benefits from the programs and have better outcomes. This is especially relevant economically when considering that today's children will become the economic workforce in the near future. By studying preschool attendance rates, we can better understand how to support parents so they can maximize their children's benefits from preschool programs. In addition to the costs to the children from being absent, there are also costs to schools with lower attendance rates in the form of penalties and sanctions (Kalil et. al, 2021). Our study focuses on attendance in order to understand what factors keep children from attending preschool. In particular, we utilize machine learning methods in order to predict the rate of attendance for preschool children in our sample based on their own characteristics and their parent's characteristics. We can then build a profile for the types of parents whose children have poor attendance rates, which can be used to inform policy.

Our results show two things: First, certain prediction methods were more effective at predicting preschool presence with our data. In particular, random forests performed significantly better than all other methods. Second, there are certain characteristics of parents that predicted worse attendance rates for their children. Those parents whose children had previous absences (including for illness) were more likely to be absent. Children whose parents had lower incomes, less predictable work schedules or who expected their children to miss more preschool also had lower attendance.

Furthermore, the presence of more adults in the household meant that children were less likely to attend preschool, possibly because they function as alternate caretakers.

2. Existing Literature

The importance of preschool in improving children's learning outcomes has been widely studied. Most literature in this field finds that attending preschool has a positive effect on potential outcomes for children. The Division of Early Childhood Education at the Ohio State Department of Education published three studies in 1992 that showed that children who attended preschool and full-day kindergarten were more likely to have higher elementary school grades (Ohio State Department of Education, 1992) than children who did not attend. Likewise, an Australian study used logistic regression models to study preschool attendance (Lim et. al, 2022). They reported much more developed language skills at the age of five in children who attended Early Child Education and Care centres at two years old. These centres provided daycare and preschooling. However, the number of hours of attendance was not associated with positive learning outcomes, only the fact that the children did attend at all was of importance (Lim et. al, 2022). These outcomes were especially pronounced for children facing adverse social situations, so interventions targeting these groups may be even more effective at improving language skills. Throndsen et. al (2019) also found a strong link between preschool attendance and children's later educational outcomes. They used a dataset with over 45,000 kindergarten children across Utah from 2017 to 2018 and measured math scores. Prior preschool attendance was a statistically significant predictor for higher math scores, especially for children from low socioeconomic status households.

As preschool has been shown to improve student outcomes, absenteeism in preschool has been the subject of many studies. Ehrlich et. al (2014) found that when parents did not consider attendance a priority, their children were less likely to attend preschool. They also showed that children who miss preschool more often will also be more likely to miss elementary school. This result was also found by Connolly & Olson (2012), who reported that half of chronically absent preschool and kindergarten children were likely to be chronically absent again the subsequent year. With our study, we hope to expand on these findings to better understand what types of students are absent. There is a gap in the literature wherein most studies use similar types of methods such as linear and logistic regression to determine preschool attendance. Machine learning methods may help to better profile the types of families whose children need intervention than using just simple methods. Many of these studies were undertaken in decades previous when current machine learning methods were not discovered, were less understood, or computing power was too expensive to use.

3. Data

Our data comes from OPENICPSR and was initially collected and used for the Show Up to Grow Up field experiment. Data was originally collected by surveying caregivers from nine subsidized preschools in Chicago from 2016 to 2017. The data collection period for each observation was 18

weeks long, during which the number of school absences was tracked. The original sample includes 741 participants and 88 variables. These observations each represent a single child and their caregiver (parent, grandparent, or other). However, the majority of caregivers were parents, so we will refer to them all as parents. Variables depict a range of demographic and other types of variables for the child/parent. All parents in the sample are lower income (less than $50,000 USD annual income). There are also sets of variables called time_pref that measure patience through future discounting scenarios. In addition, there are sets of variables where the parent ranks their child's perceived social and academic skills compared to other children.

We need to split our sample into training and test samples to test our methods' prediction power. However, since approximately 30 percent of values were missing in this dataset, the training and test samples were very small after omitting these values. We wanted to retain our sample size rather than omitting observations with missing values. Thus, multiple imputation using predictive mean matching was used to estimate the likely missing values (see Figure 1). Multiple imputation is a process that uses the distribution of and relationship between variables in the sample to generate values to fill in missing data. Predictive mean matching is a semi-parametric type of multiple imputation that draws many subsets (for our sample, we chose 100) where observations with missing values are compared to observations that contain no missing values but have similar dependent variable values (StataCorp, 2023). A random value is chosen from the set of closest neighbouring observations and imputed, or assigned to replace the missing value. One primary advantage to this imputation method is that it must choose values that are plausible as they are based on observed values from other observations (UCLA, 2021). Without multiple imputation, roughly two-thirds of our sample would have been dropped to remove missing values. Our methods were initially run on that much smaller sample (246 observations, 25 variables) and predictions were slightly worse overall than our current results using multiple imputation.

```
Data.imp <- mice(Data, m = 100, method = 'pmm')
```

Figure 1

Our outcome variable, Attendance_ratio, was created by dividing the number of attended preschool days by the number of potential preschool days during the period. Observations with attendance ratios of zero and unnecessary variables such as child_id were removed from the sample. Furthermore, variables with very little variation dropped as they would have negligible predictive power. 728 observations with 60 variables remain for our analysis. Summary statistics show a mean attendance ratio of 0.82 (See Figure 2).

For our prediction methods, we split our data into training and test samples. The training sample contains 485 observations of 60 variables and the test sample contains 243 observations of 60 variables. We chose a $\frac{2}{3}$ to $\frac{1}{3}$ split as the original sample is still relatively small.

	mean <dbl>	median <dbl>	sd <dbl>	min <dbl>	max <dbl>
Attendance_ratio	0.78	0.82	0.18	0.01	1
nadults	1.11	1.00	1.10	0.00	6
income	20303.71	19907.00	11981.19	0.00	50000
ndays_absentsick	1.78	1.00	2.26	0.00	15
other_takescare	3.12	3.00	1.36	1.00	5
ndays_absentother	1.17	1.00	1.68	0.00	12
commute_home_work	30.87	30.00	19.57	0.00	130
working_hours	29.92	32.00	14.89	0.00	80
safe	0.43	0.00	0.50	0.00	1
copayment_amount	101.56	67.00	104.99	0.00	670

Figure 2

4. Methods

This study utilizes a range of machine learning methods in order to predict children's attendance ratios. All methods are explained here, with results in the following section. Since our primary objective is a prediction of a non-categorical variable, our methods use regression-type models that focus on dealing with the bias-variance tradeoff. Every model was first trained on the full sample or training sample depending on how the method deals with the bias-variance tradeoff. Then we computed the test MSE of each model using our test sample. This allowed us to see how each model would perform when used on another sample since we are not interested in predicting our own data. Models were compared based on their test MSE (mean squared error), which measures how closely the model predicts test sample values. Lower test MSE values are better as they have less error.

$$MSE \equiv \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$$

We used a number of linear models with various model selection methods. This enabled us to see which covariates helped the most to explain attendance ratios. First, best subset selection was run using the full sample (Code: See Appendix 1). It works by choosing the best model for each different number of covariates and then finding the best model based on the chosen information criteria. The information criteria penalizes overfitting, which is why we can use the full sample. We chose the parameter adjusted R^2 because it is widely used in statistics and therefore easier to interpret. The model with the highest adjusted R^2 was chosen. Since the use of best subset selection estimates more models than forward stepwise and backward stepwise selection, we did not use either of those methods.

Next, we used validation sets (Code: See Appendix 2). It works by estimating all possible linear models on the training sample. Then the test MSE for each model is computed using the test sample. The model that has the lowest test MSE is chosen. For this method, we had too many variables and not enough computing power to perform the operations required. Thus, we dropped variables that we felt were less important based on our best subset and principal component analysis results. Hence, we used a reduced sample of 18 variables for this method only.

We then used cross-validation which also estimates all possible models on the training sample and tests their predictive power (using test MSE) against the training sample (Code: See Appendix 3).

However, this method uses a different method for choosing the training and test samples. The full sample is split into K folds (we chose 10), then one fold is put aside as the test sample and the rest becomes the training sample. The $K-1$ training sample is used to estimate the models and test predictive power against the first fold. Then these steps are followed again as each fold takes its turn as the test sample. After acquiring K different test MSE, we choose the model with the lowest average test MSE.

The above three methods were used and the results were compiled into a table (See Figure 3). This table gives a general overview of the covariates that these three linear models chose, many of which are the same.

Table 1: Results

	Dependent variable:		
	Attendance_ratio		
	Best	ValSet	CV
	(1)	(2)	(3)
income	0.00000*	0.00000***	0.00000**
nadults	−0.028***	−0.031***	
ndays_absentsick	−0.023***	−0.019***	−0.024***
ndays_absentother	−0.015**	−0.020***	−0.020**
commute_home_work		0.001*	
work_predictable	0.018**	0.020**	0.018**
rank_aca_skills	0.015***		
rank_soc_change3	−0.027***		−0.022***
rank_soc_change5	0.021***		0.021***
daysexpecttomiss		−0.013*	
timepref1_5		−0.035**	
timepref2_3		0.020	0.004
timepref3_2	0.009	0.005	0.005
timepref3_3	−0.018		−0.021
parent_drops	−0.047		
parent_takescare	0.018*		0.023**
other_takescare		−0.017**	
single		0.065**	
Constant	0.712***	0.742***	0.695***
Observations	243	243	243
R^2	0.277	0.283	0.224
Adjusted R^2	0.239	0.246	0.191
Note:			*p<0.1; **p<0.05; ***p<0.01

Figure 3

Three shrinkage methods were used to estimate full linear models using our full sample. They work by putting a constraint on the value of coefficients to decrease variance and reduce overfitting. The three methods are ridge regression, Least Absolute Shrinkage and Selection Operator (LASSO), and elastic nets. They all use a penalty term λ, the optimal value of which can be found using cross validation. Ridge regression can set some estimator ($\hat{\beta}$) values to low values, but cannot set them to

0. It uses the squares of the coefficients in its calculation.

$$\hat{\beta}_{Ridge} \equiv \underset{\{\beta\}}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} (y_i - x_i'\beta)^2 + \lambda \sum_{j=1}^{p} \beta_j^2$$

LASSO however, can set the value of coefficients to 0 and uses the absolute value of the coefficient in its calculation.

$$\hat{\beta}_{LASSO} \equiv \underset{\{\beta\}}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} (y_i - x_i'\beta)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

Elastic nets combine the other two methods and use parameter α which determines the best combination of ridge and LASSO to use. If $\alpha = 1$, then it is identical to LASSO and if $\alpha = 0$, it is identical to ridge. We used all values of α in 0.1 increments from 0.1 to 0.9 to pick the best value, with $\alpha = 0.5$ producing the best predictions.

Two more supervised learning methods that this paper used were regression trees (Code: See Appendix 5) and random forests (Code: See Appendix 6). With regression trees, binary splits are made in the data based on the value of covariates. A threshold value of each covariate is chosen at which the observations are split into pairs. Different regions are created from the different splits, with each observation falling into one region. Within the region, the prediction for each observation equals the average of all values of the outcome variable of observations in the same region. Our tree is shown in Figure 4.

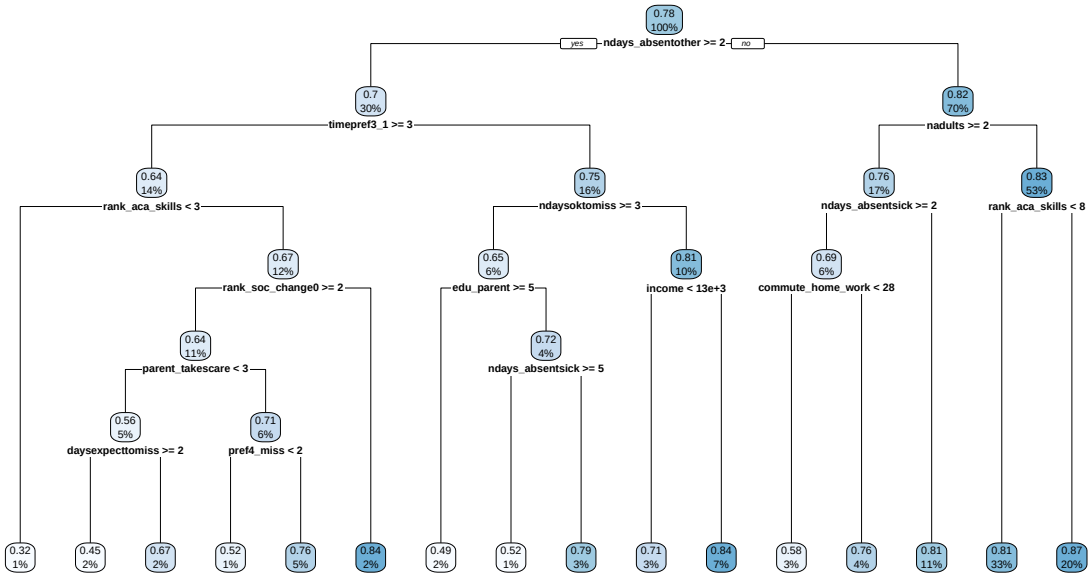

Figure 4

Notable features include variables that appear more than once as they are better predictors of attendance ratio. For example, the number of days previously missed because of sickness has two splits and how the parent ranks their children's social skills compared to other children also appears twice. We can also see that the right-most region contains the observations with the highest predicted attendance ratio of 0.9 (20% of the sample). These children have had less than two previous absences for non-sick reasons, less than two adults live in their homes, and their parents rank their social skills as the same as or better than other children in the USA.

Random forests are essentially an extension of regression trees that uses bootstrapping and model averaging to take the average predictions of B amount of trees. We chose the default B of 500 trees. This method is better for prediction than regression trees because it reduces the variance but has the same bias which lets us make better out of sample predictions. Figure 5 shows a comparison between our regression tree predictions and random forest predictions. The difference is striking as the random forest predictions are mostly very close to the 45 degree line, meaning they predict the actual data from the test sample quite well.

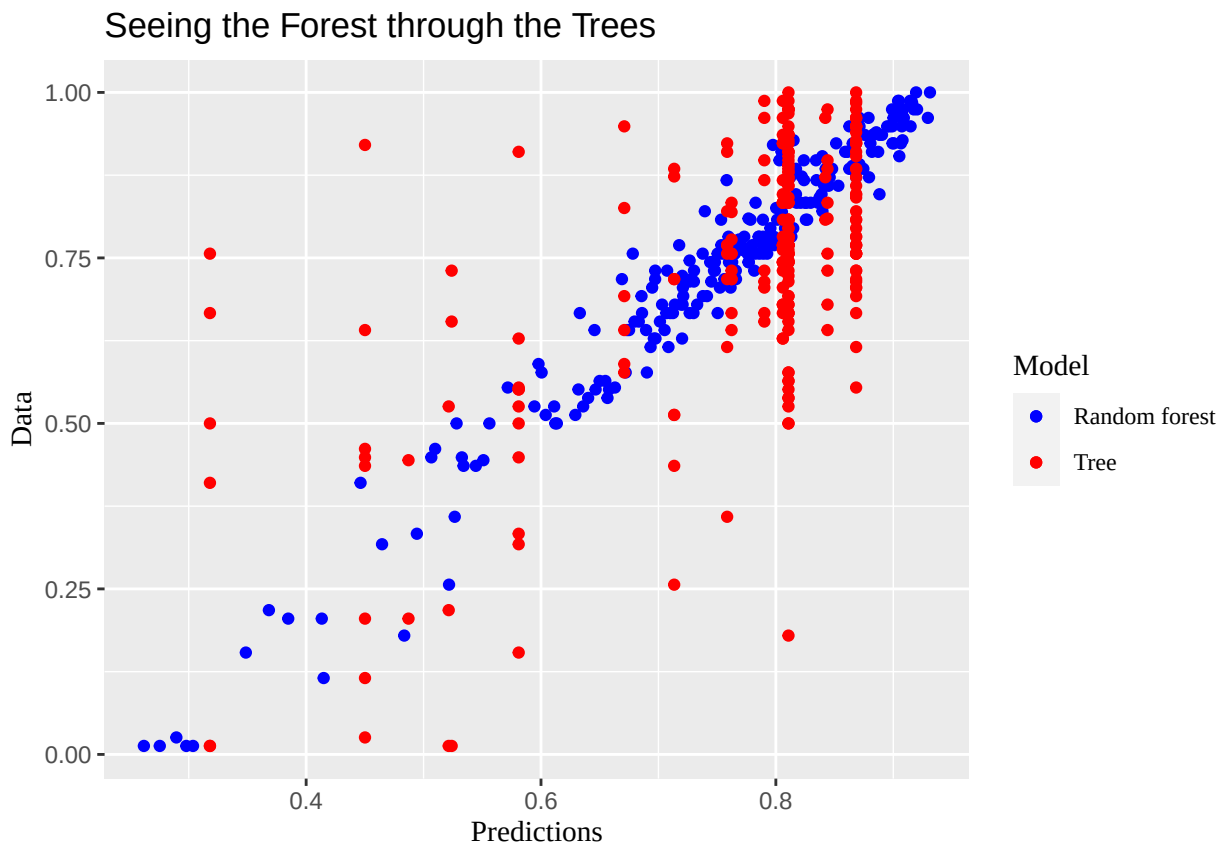

Figure 5

For this paper, we used two nonparametric methods that both have linear estimators, K nearest neighbours (KNN) (Code: See Appendix 7) and kernel estimators (Code: See Appendix 8). KNN works by taking the average of the outcome variable values y_i whose covariate values x_i are similar to those of the observation we are trying to predict. K is the tuning parameter for the number of neighbours selected to average over, where K is chosen by leave-one-out cross-validation (a modified version of K-fold). Kernels put weights on observations based on their distance from the observation we are trying to predict. Its tuning parameter is bandwidth h, which we also choose using leave-one-out cross-validation. The bandwidth determines the width of a range on either side of the observation that we want to predict, where any observations in that range are used to predict the value of our observation. For kernels, an estimator and kernel function type must also be selected. We used a local linear estimator which helps deal with boundary bias when trying to estimate values close to the boundary of the support for the data. For the kernel function type, Epanechnikov was selected as it is generally considered the best.

We also used neural networks in our paper (Code: See Appendix 9). This method interconnects our variables through layers similar to how a brain's neurons are connected. Each covariate is an input node in the input layer, and the dependent variable is the output layer. Between these layers are hidden layers that contain nodes, each of which is a linear combination of the nodes from previous layers. Multiple layers increase the flexibility of the model which will make better predictions.

Nonlinear least squares is used to estimate the parameters. We used the covariates selected from our best subset selection as our input nodes. Then we created three hidden layers with three nodes each. This produced a model that estimates 327 parameters.

Last, we performed Principal Component Analysis (PCA), an unsupervised method, to evaluate variance and correlation patterns (Code: See Appendix 10). This is a dimension reduction method that gives a summary of the correlation and variance patterns of the original covariates but reduces the dimension. First we standardize the data to deal with scaling issues. Principal components are linear combinations of the original covariates that then we create to capture the patterns of those covariates. We want to find coefficients that maximize the variance of the principal component scores. We then get the scores and loading vectors for each component. Scores for each component must be uncorrelated with each other. Loading vectors contain coefficients that account for the weights of each variable within that component. We can use a biplot (See Figure 6) to represent the main results.

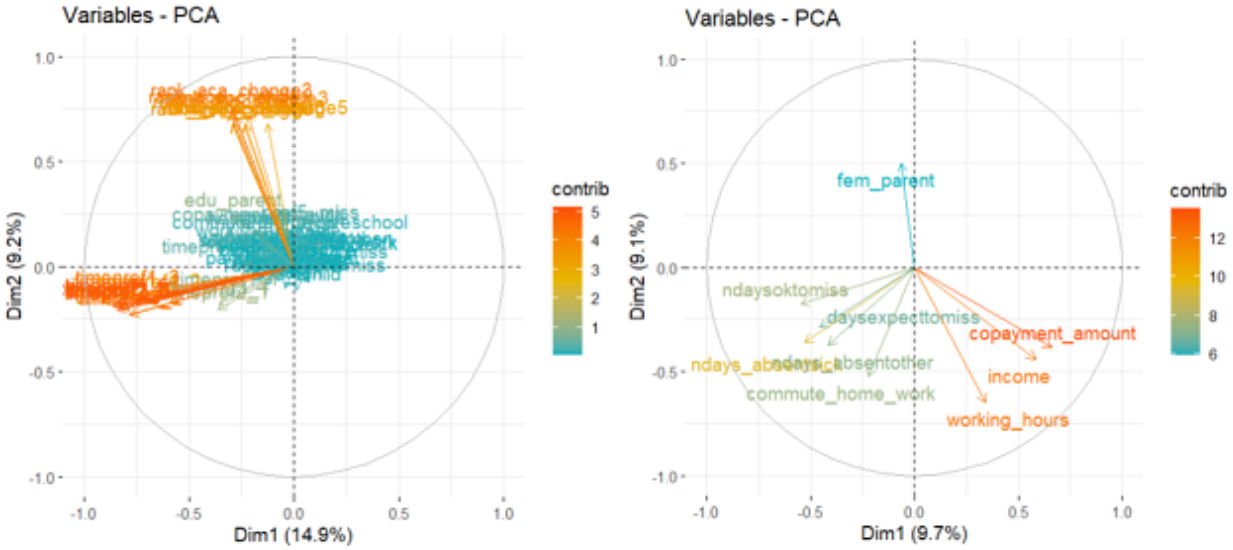

Figure 6

By analyzing the length, direction, and angle between loading vectors we can see how the original variables are related to each other and how they contribute to the first two principal components. Our initial biplot (left) showed us that the social and academic ranking variables are all very highly correlated, and the time preference variables are all very highly correlated. The rest of the variables are overlapping and hard to read, so we removed most of the preference variables. We kept the 5 variables that contribute the most to principal components one and two respectively and then reran the PCA (right). This plot was more helpful for visualizing the relationship between our covariates. The result shows that copayment_amount, income and working_hours are closely correlated because the angle between their vectors is small. We can identify them as a "money related considerations of absences" group. To the left we can see ndaysoktomiss, ndayabsent, etc. are closely correlated. These

we can refer to as the "time related considerations of absences" group. Finally, fem_parent stands alone at the top and has a negative correlation to the other variables. We also used an extension of PCA called Principal Components Regression (Code: See Appendix 11). It uses cross validation to pick the number of principal components taken from PCA and runs a regression using them.

5. Results

Method and Test MSE	
Random forest	0.07722575
Tree	0.15799146
Best Subset	0.16373008
Cross Validation (10 Fold)	0.16668510
Elastic net	0.16705713
LASSO	0.16734067
Ridge	0.16846070
OLS (Validation sets)	0.17027958
Neural networks	0.17938563
Principle Components Regression	0.20196166
K-nn	0.20832025
Kernel	0.76240062

Figure 7

Overall, our results obtained from applying various machine learning methods provided us with some meaningful insights (See Figure 7). We can see that among the methods used, random forest yields the best results in terms of the accuracy of prediction. Its ability to reduce the variance while retaining the same bias as regression trees might be the reason for its success in out-of-sample predictions. The regression tree method performed much worse than the random forest, which makes sense because trees tend to overfit the data. Our linear methods all performed fairly similarly and did much better than many of the more complicated methods. Elastic nets, LASSO, and Ridge regression performed similarly as they should, with elastic nets performing slightly better. They likely performed decently because of their ability to limit overfitting using the penalty term λ. PCR and KNN were worse compared to most other methods, but only by a slight amount. It is possible that with more fine tuning, they could be on par with the methods that performed better. On the other hand, kernel methods produced the worst result in our analysis. The limitation of this method could be from inaccuracies caused by the curse of dimensionality. The kernel formula suggests that as the number of covariates increases, our sample size needs to increase faster in order to maintain accurate statistical results. Since our sample size is relatively small given the number of variables, it makes sense that kernels did not perform as well as other methods. Interestingly, in our initial results before multiple imputation, kernels performed much better. When we doubled the number of variables but only tripled our observations, kernels became much less useful for prediction.

6. Conclusion

Finally, Based on the variables selected by our methods, we can build a rough profile of the parents whose children are the most likely to miss school. These parents rank their child's academic skills as lower than other children's, they expect their children to miss more classes in the future, and their children have multiple previous absences for various reasons. They also have lower incomes, longer commute times from home to work, it's easier for them to stay home to take care of their child, and more adults live in their households. These parents also think that it's okay for their children to miss multiple days of preschool for various reasons including if the parent doesn't have to go to work that day. Lastly, they are less patient (based on future discounting questions).

We are at a turning point in the world of economic research where we now have enough computing power to harness machine learning methods. This allows us to use methods that a decade ago would have been either prohibitively expensive, or downright impossible. Future studies of preschool attendance should use all available tools to obtain better results. Researchers commonly choose covariates on a whim, when there are powerful tools that can pick objectively better models to use. Rather than employing methods that have already been used extensively in this field, there is much to gain from machine learning applications. If, as we believe, children are the future, then we should be using the tool of the future, artificial intelligence, to find ways to make their lives better.

7. References

Connolly, F., & Olson, L. S. (2012). Early elementary performance and attendance in Baltimore City Schools' pre-kindergarten and kindergarten. Baltimore Education Research Consortium. https://files.eric.ed.gov/fulltext/ED535768.pdf

Kalil, A., Mayer, S.E, Gallegos, S. (2021). Using behavioral insights to increase attendance at subsidized preschool programs: The Show Up to Grow Up intervention, Organizational Behavior and Human Decision Processes, 163, 65-79, https://doi.org/10.1016/j.obhdp.2019.11.002.

Lim, S., Levickis, P., & Eadie, P. (2022). Associations between Early Childhood Education and Care (ECEC) attendance, adversity and language outcomes of 2-year-olds. Journal of Early Childhood Research, 20(4), 565–579. https://doi-org.proxy.lib.sfu.ca/10.1177/1476718X221087078

National Center for Education Statistics (2017). Table 5.1. Compulsory school attendance laws, minimum and maximum age limits for required free education, by state: 2017. https://nces.ed.gov/programs/statereform/tab5_1.asp

Office of Head Start. (2023). Head Start Services. An Office of the Administration for Children and Families. https://www.acf.hhs.gov/ohs/about/head-start

Ohio State Department of Education (1999). The effects of preschool attendance & kindergarten schedule: Kindergarten through grade four. https://files.eric.ed.gov/fulltext/ED400038.pdf

Peisner-Feinberg, E. S., Burchinal, M. R., Clifford, R. M., Culkin, M. L., Howes, C., Kagan, S. L., & Yazejian, N. (2001). The relation of preschool child-care quality to children's cognitive and social developmental trajectories through second grade. Child development, 72(5), 1534-1553.

StataCorp. (2023). Stata 18 Base Reference Manual. College Station, TX: Stata Press.

Throndsen, J.E., Shumway, J.F. & Moyer-Packenham, P.S. (2019). The Relationship Between Mathematical Literacy at Kindergarten Entry and Public Preschool Attendance, Type, and Quality. Early Childhood Education Journal, 48, 473–483. https://doi.org/10.1007/s10643-019-01014-7

UCLA (2021). How do I perform multiple imputation using predictive mean matching in R? R FAQ. UCLA Advanced Research Computing. https://stats.oarc.ucla.edu/r/faq/how-do-i-perform-multiple-imputation-using-predictive-mean-matching-in-r/

8. Appendix

```r
# Best Subset Selection
subset_selection <- regsubsets(Attendance_ratio ~ .,
                               data = Data, nvmax = 12, really.big=T)
# Show the different measures after all the models are estimated
subset_sum <- summary(subset_selection)
data.frame( Adj.R2 = which.max(subset_sum$adjr2),
            CP = which.min(subset_sum$cp),
            BIC = which.min(subset_sum$bic))

# Let us do best subset selection according to adjusted R squared
subset_adjr2_model <- data.frame(selected =
              as.matrix(subset_sum$which[which.max(subset_sum$adjr2), ]))
# Look at the selected variables
subset_adjr2_model <- dplyr::filter(subset_adjr2_model, selected == TRUE)
subset_adjr2_model$variable <- row.names(subset_adjr2_model)
subset_adjr2_model$variable
```

Appendix 1

```r
# Validation Sets
# Function that generates all the possible models with a set of variables
all_models <- function(variables){
  # How many variables in "variables"?
  K <- length(variables)
  # Use binary representation
  bin_vec <- rep(list(0:1), K)
  # Makes vectors of 1 and 0
  # Consider all of the different combinations, except the empty model.
  # There will be 2^K - 1 combinations
  bin_mat <- expand.grid(bin_vec)[-1, ]

  # Initialize the results. The loop will fill that list
  list_of_RHS <- list()
  # Fill up the list by looping over all combinations
  for(i in 1:nrow(bin_mat)){
    list_of_RHS[[i]] <- name_from_bin(bin_mat[i, ], variables)
  }
  return(list_of_RHS) # Each row of that list is a combination of covariates
}
```

```r
# function that estimates all the possible models and computes the test MSE
all_subset_regression <- function(covariates_to_consider,
                                  y_var, train_dat, test_dat)
  { # Makes all the possible combos
  models_to_consider <- all_models(covariates_to_consider)

  results <- map(models_to_consider, MSEs,
                 Y_name = y_var, training_data = train_dat,
                 test_data = test_dat)
# Format the "results" nicely
  useful_results <- matrix(unlist(results), ncol = 3, byrow = TRUE)
  useful_results <- as_tibble(useful_results)
  names(useful_results) <- c(
    "num_vars",
    "training_error","test_error")
  return(useful_results)
}

max_X <- colnames(Training_sample_R)[-c(1,which(colnames(Training_sample_R) == "Attendance_ratio
max_X

library(modelr)  # needed for the add_residual function to work
performances <- all_subset_regression(covariates_to_consider = max_X,
                                      y_var = "Attendance_ratio",
                                      train_dat = Training_sample_R,
                                      test_dat = Test_sample_R)
# Smallest training error per number of covariates used
min_k_train <- performances %>%
  group_by(num_vars) %>%
  summarise(min_training_error = min(training_error))
min_k_train
# Smallest test error per number of covariates used
min_k_test <- performances %>%
  group_by(num_vars) %>%
  summarise(test_error = min(test_error))
min_k_test

which(performances$test_error == min(performances$test_error))
best <- which.min(performances$test_error)
```

```
performances[best, ]
all_models(max_X)[[best]]
```

Appendix 2

```
# K-fold cross-validation
# setting seed to generate a
# reproducible random sampling
set.seed(125)

# defining training control
# as cross-validation and
# value of K equal to 10
train_control <- trainControl(method = "cv",
                              number = 10)

# training the model by assigning sales column
# as target variable and rest other column
# as independent variable
cv_model <- train(Attendance_ratio ~., data = Data,
                  method = "lm",
                  trControl = train_control)

# printing model performance metrics
# along with other details
print(cv_model)
cv10_mse <- cv_model$results[[2]]
```

Appendix 3

```
# Ridge regression
# Run these on the whole dataset, but test only on the test data
cv_ridge <- cv.glmnet(x = data.matrix(Data[ , 2:K ]),
                      y = Data$Attendance_ratio, alpha = 0)
ridge_lambda <- cv_ridge$lambda.min # optimal lambda

ridge <- glmnet(y = Data$Attendance_ratio, x = Data[ , 2:K ],
                alpha = 0, family = "gaussian")
# make predictions
ridge_fit <- predict(ridge, newx = data.matrix(Test_sample[ , 2:K ]),
                     s = ridge_lambda )
```

```r
# LASSO regression
cv_lasso <- cv.glmnet(x = data.matrix(Data[ , 2:K ]),
                      y = Data$Attendance_ratio, alpha = 1)
lasso_lambda <- cv_lasso$lambda.min # optimal lambda

lasso <- glmnet(y = Data$Attendance_ratio, x = Data[ , 2:K ],
                alpha = 1, family="gaussian")
# make predictions
lasso_fit <- predict(lasso, newx = data.matrix(Test_sample[ , 2:K ]),
                     s = lasso_lambda)

# Elastic net
cv_net <- cv.glmnet(x = data.matrix(Data[ , 2:K ]),
                    y = Data$Attendance_ratio,
                    alpha = 0.5, family = "gaussian")
net_lambda <- cv_net$lambda.min  # optimal lambda for elastic nets
elastic_net <- glmnet(y = Data$Attendance_ratio, x = Data[ , 2:K ],
                      alpha = 0.5, family = "gaussian"  )
# make predictions
elastic_fit <- predict(elastic_net,
                       newx = data.matrix(Test_sample[ , 2:K ]), s = net_lambda)
summary(elastic_fit)
plot(elastic_fit)
```

Appendix 4

```r
# Regression Trees
tree_preschool <- rpart(Attendance_ratio ~ ., data = Data, method = "anova", )
summary(tree_preschool )
plotcp(tree_preschool)
printcp(tree_preschool)
rpart.plot(tree_preschool)
```

Appendix 5

```r
# Random Forests
rf_preschool <- randomForest(Attendance_ratio ~., data = Data)
# Predict the test sample
rf_pred <- predict(rf_preschool, newdata = Test_sample)
```

Appendix 6

```r
# KNN
# Loop over different values of neighbours
neighbours <- c(1, 5, 10, 15, 20, 25, 35, 50, 100)
mat <- matrix(0, nrow = length(neighbours), ncol = 1)
for (i in 1:length(neighbours))
{
  knn_gna <- knn.reg(train = Training_sample,
                     test = Test_sample,
                     y = Data$Attendance_ratio,
                     k = neighbours[i])$pred
  test <- MSE(y = Data$Attendance_ratio, fhat = knn_gna)
  mat[i, ] <- test
}
mat <- data.frame(neighbours = neighbours, Test_MSE = mat)
# Select the optimal amount of neighbours
best_neighbours = neighbours[which.min(mat$Test_MSE)]
# Predictions using the optimal neighbours
knn_fit <- knn.reg(train = Test_sample, y = Data$Attendance_ratio,
                   k = best_neighbours)$pred
```

Appendix 7

```r
# Kernel Estimators
# Local linear estimator
Training_sample_df <- as.data.frame(Training_sample)
Test_sample_df <- as.data.frame(Test_sample)
bw_ll <- npregbw(ydat = Training_sample_df[, 1],
                 xdat = Training_sample[, -1], regtype = "ll",
ckertype = "epanechnikov")
model_ll <- npreg(bws = bw_ll, newdata = Training_sample)
# Using the optimal bandwidth on the third data set
model_ll <- npreg(bws = bw_ll$bw, ydat = Test_sample_df[, 1],
                  xdat = Test_sample[, -1],
regtype = "ll", ckertype = "epanechnikov")
# Get the predictions
ll_fit <- model_ll$mean
```

Appendix 8

```r
nn_preschool <- neuralnet(Attendance_ratio ~ income + nadults +
                            ndays_absentsick + ndays_absentother +
                            work_predictable + rank_aca_skills +
```

```
                              rank_soc_change3 + rank_soc_change5 +
                              timepref3_2 + timepref3_3 + parent_drops +
                              parent_takescare,
                            data = Data, hidden = c(3, 3, 3),
                            stepmax = 1000000, lifesign = "full")
nn_pred <- predict(nn_preschool, newdata = Test_sample, all.units = FALSE)
```

Appendix 9

```
PCAData <- Data %>%
  select(2:17,28,45:50)

results <- prcomp(PCAData, scale = TRUE, center = TRUE)

summary(results)
results$rotation
library(factoextra)
fviz_eig(results,
         addlabels = TRUE,
         ylim = c(0, 70))

fviz_pca_biplot(results,
                label="var", select.var = list(contrib=10), repel = T)

fviz_pca_var(results, col.var="contrib",
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
             select.var = list(contrib=10), repel = T)

# contribute for each axes
fviz_contrib(results, choice = "var", axes = 1, top = 5)
fviz_contrib(results, choice = "var", axes = 2, top = 5)
```

Appendix 10

```
pcr_fit <- pcr(Training_sample$Attendance_ratio ~
                 data.matrix(Training_sample[ , 2:(K - 1)]), scale = TRUE,
validation = "CV")
summary(pcr_fit)

pcr_pred <- predict(pcr_fit, data.matrix(Test_sample[ , 2:(K - 1)]), ncomp = 7)
```

Appendix 11