Using Environmental Factors During Pregnancy to Predict Asthma in Children

Judy Fordjuoh

April 14, 2022

# INTRODUCTION

## Background

There have been several single-exposure studies that have documented possible effects of environmental factors on lung function; however there have not been any studies that rely on an exposome approach. In “Early-life exposome and lung function in children in Europe: an analysis of data from the longitudinal, population-based HELIX cohort” by Ageir et. al, the authors aimed to evaluate the association between a broad range of prenatal and postnatal lifestyle and environmental exposures and lung function in children by using data from HELIX harmonized birth cohorts in Europe. After the research group working on HELIX released four datasets that represents their real HELIX data, various researchers have shared their their analytic methods for high-dimensional data. In this research project, I will use the various datasets to evaluate the association between a broad range of prenatal lifestyle and environmental exposures and lung function, specifically the development of asthma, in children.

## Research Question

Research Question: Does specific factors of the built environment during a mother’s pregnancy predict whether a child develops asthma?

Why is predicting this important? Predicting asthma based on the mother’s environmental conditions while pregnant is important because once researchers understand what features contribute the most to asthma development, tracking this feature and limiting the exposure to the specific feature during pregnancy can reduce the chances of asthma development in the child. Also, if the mother’s medical team is able to keep track of the asthma-influencing features, then if the mother has a high exposure during pregnancy, it can alert the child’s medical provider of their higher risk of asthma development. This is essential as an early asthma diagnosis is important for the proper treatment of young children with respiratory symptoms. Equally important, identifying children at high risk for asthma is important because it could lead to personalized and improved disease management and can potentially reduce healthcare costs. Also by having an earlier diagnosis, parents and clinicians can slowly monitor the asthma symptoms and can have an accurate prognosis of whether the asthma will develop into chronic asthma.

## Data Cleaning, Prepping, and Exploration

I utilized the exposome dataset, which contains all of the environmental features measured on children, and the phenotype dataset, which contains health outcomes measured during the study. To narrow down the exposome dataset, I specifically included variables that are associated with the built environment and were measured during pregnancy. The features I selected was: Walking and/or cycling activity during pregnancy (h\_pamod\_t3\_None), Is there a green space in a distance of 300m? (h\_greenyn300\_preg\_None), pm10 value (h\_pm10\_ratio\_preg\_None), meters of public transport mode lines (only buses) inside each 300 m buffer (h\_accesslines300\_preg\_dic0), number of bus public transport mode stops inside each 300 m buffer(h\_accesspoints300\_preg\_Log), building density (m2 built/km2) within a buffers of 300 m buffer during pregnancy period (h\_builtdens300\_preg\_Sqrt), connectivity density within a buffers of 300 m buffer during pregnancy period(number of intersections / km2)(h\_connind300\_preg\_Sqrt), and walkability index during pregnancy period (h\_walkability\_mean\_preg\_None). The selected features are related to the mother’s air quality/air pollution levels during pregnancy. Factors such as access to green space and the walkability index are factors related to the built environment that may influence the amount of daily physical activity a mother may have. Lack of physical activity during pregnancy can lead to various health issues for pregnant individuals, which can affect children in the future.

There were 321 participants who did not have green space in a distance of 300m during pregnancy and 980 participants who did. The average number of bus public transport mode stops inside each 300 m buffer was 2.67 (range = 3.26). The meters of public transport mode lines (only buses) inside each 300 m buffer was 0.199m. The average pm10 value during pregnancy was 23.504 (range = 39.632). The average building density during pregnancy was 417.06 m2 built/km2 (range = 796.55). The average connectivity density during pregnancy was 12.737 (range = 25.389). The average walkability index during pregnancy was 0.2674 (range = 0.525).

The outcome (asthma) is unbalanced, with 1159 children without asthma and 142 with asthma. The datasets were merged by ID.

# Algorithms

In this project, I will be doing a mix of algorithms to discover which of the features I’ve selected in my model are the most important in predicting asthma. Variables with high importance will be drivers of the outcome (asthma) and their values have a significant impact on one’s asthma status. I will also be assessing the model’s accuracy. I will do bagging, random forest, LASSO, and a support vector machine model to see which algorithms will provide me with the best accuracy.

Bagging is an ensemble algorithm that fits multiple models on different subsets of a training dataset, then combines the predictions from all models. Specifically, bagging averages results across bootstrapped samples of training data. Random forest is an extension of bagging that also randomly selects subsets of features used in each bootstrap. With random forest, we are not so concerned with overfitting because variance is reduced by the decorrelation of trees. By conducting both, I can see which one would provide a better accuracy in for the model. Bagging and random forest are a tree models which typical rank features by how well they improve the purity of the node. Variable selection is an important part of prediction model development and bagging and random forest will give estimates of what features are important in the classification of asthma.

LASSO is the only regularized regression model included in this project. The LASSO model has the ability to shrink data values towards a central point such as 0. Due to LASSO’s shrinkage ability, it is well-suited for feature selection/elimination, which is what I am concerned about in this project.

Support Vector Machine (SVM) is a robust supervised machine learning algorithm which can be used for classification or regression problems. SVM will use the optimal hyperplane to correctly classify between the data points of the two asthma levels. Typically, SVM is better than most of the other algorithms used as it has a better accuracy in results, and I believe it will be interesting to compare the accuracies fo SVM to the other algorithms in this study. Feature importance for asthma can also be determined using SVM by by comparing the size of the feature coefficients to each other. This will show us which variables are not important and holds less variance.

### BAGGING

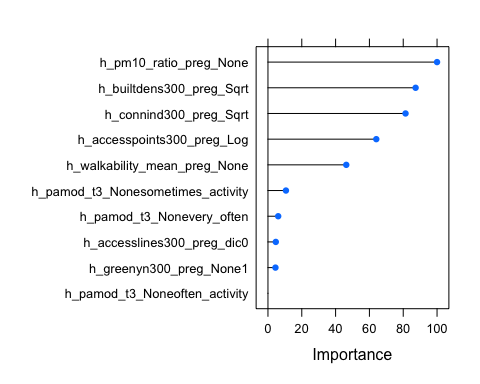
set.seed(100)   
  
#Using the treebagg method, which provides greater control of the hyperparameters  
controlsettings<-trainControl(method="cv", number=1)  
bagging\_asthma<-train(asthma ~., data=train\_asthma, method="treebag", trcontrol=controlsettings, nbagg=10, control=rpart.control(minsplit=20, cp=0))  
  
bagging\_asthma$results

## parameter Accuracy Kappa AccuracySD KappaSD  
## 1 none 0.8809206 0.003192283 0.0138357 0.03619081

varImp(bagging\_asthma)

## treebag variable importance  
##   
## Overall  
## h\_pm10\_ratio\_preg\_None 100.000  
## h\_builtdens300\_preg\_Sqrt 87.320  
## h\_connind300\_preg\_Sqrt 81.362  
## h\_accesspoints300\_preg\_Log 64.063  
## h\_walkability\_mean\_preg\_None 46.311  
## h\_pamod\_t3\_Nonesometimes\_activity 10.631  
## h\_pamod\_t3\_Nonevery\_often 6.053  
## h\_accesslines300\_preg\_dic0 4.632  
## h\_greenyn300\_preg\_None1 4.427  
## h\_pamod\_t3\_Noneoften\_activity 0.000

plot(varImp(bagging\_asthma))



confusionMatrix(bagging\_asthma) #0.8809

## Bootstrapped (25 reps) Confusion Matrix   
##   
## (entries are percentual average cell counts across resamples)  
##   
## Reference  
## Prediction 0 1  
## 0 87.9 10.6  
## 1 1.3 0.2  
##   
## Accuracy (average) : 0.8809

With the bagging method, the average accuracy was 0.8809. In this model, the most important variable in predicting asthma was the pm10 value during pregnancy (100.00), the building density within a buffers of 300 m buffer during pregnancy period (87.320), and the connectivity density within a buffers of 300 m buffer during pregnancy period(number of intersections / km2) (81.362). Interestingly the level “often\_activity” from the walking and/or cycling activity during pregnancy feature had 0.000, which means it very little predictive power for asthma however, “sometimes\_activity” was 10.631 and “very\_often” was 6.053.

### RANDOM FOREST

set.seed(100)  
  
#Trying three different values of mtry (square root, half)  
# since we are not specifying our cross validation, the default is a bootstrap. R is bootstrapping 25 times.  
  
mtry.vals <- c(ncol(train\_asthma) - 1, sqrt(ncol(train\_asthma) - 1), 0.5\*ncol(train\_asthma) - 1)  
  
mtry.grid <- expand.grid(.mtry = mtry.vals)  
rf\_asthma <- train(asthma ~., data = train\_asthma, method = "rf", metric = "Accuracy", tuneGrid = mtry.grid, ntree = 100)  
  
confusionMatrix(rf\_asthma) #Accuracy (average) : 0.8868

## Bootstrapped (25 reps) Confusion Matrix   
##   
## (entries are percentual average cell counts across resamples)  
##   
## Reference  
## Prediction 0 1  
## 0 88.5 10.6  
## 1 0.7 0.2  
##   
## Accuracy (average) : 0.8868

rf\_asthma$results

## mtry Accuracy Kappa AccuracySD KappaSD  
## 1 2.828427 0.8865768 0.005352124 0.01540246 0.03557021  
## 2 3.500000 0.8866975 0.016551433 0.01572857 0.04255275  
## 3 8.000000 0.8788789 0.014351772 0.01668224 0.04911192

rf\_asthma$bestTune

## mtry  
## 2 3.5

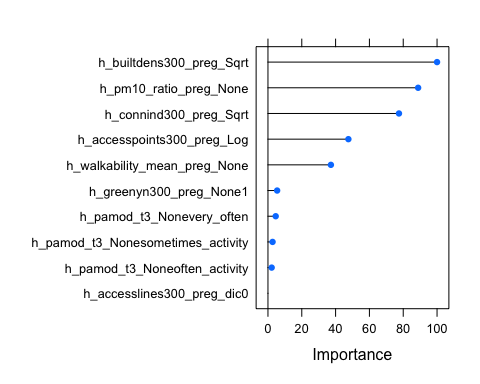
rf\_asthma$finalModel

##   
## Call:  
## randomForest(x = x, y = y, ntree = 100, mtry = min(param$mtry, ncol(x)))   
## Type of random forest: classification  
## Number of trees: 100  
## No. of variables tried at each split: 4  
##   
## OOB estimate of error rate: 11.51%  
## Confusion matrix:  
## 0 1 class.error  
## 0 806 6 0.007389163  
## 1 99 1 0.990000000

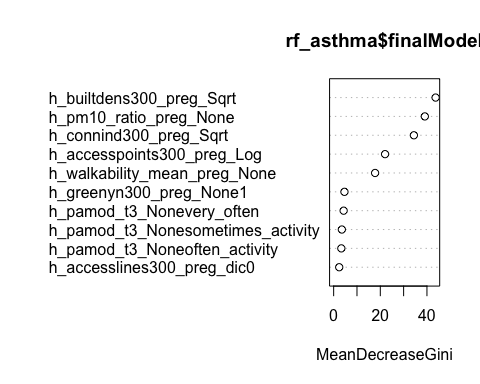
varImp(rf\_asthma)

## rf variable importance  
##   
## Overall  
## h\_builtdens300\_preg\_Sqrt 100.000  
## h\_pm10\_ratio\_preg\_None 88.815  
## h\_connind300\_preg\_Sqrt 77.481  
## h\_accesspoints300\_preg\_Log 47.566  
## h\_walkability\_mean\_preg\_None 37.225  
## h\_greenyn300\_preg\_None1 5.430  
## h\_pamod\_t3\_Nonevery\_often 4.578  
## h\_pamod\_t3\_Nonesometimes\_activity 2.724  
## h\_pamod\_t3\_Noneoften\_activity 2.194  
## h\_accesslines300\_preg\_dic0 0.000

plot(varImp(rf\_asthma))



varImpPlot(rf\_asthma$finalModel)

 With the random forest method, the average accuracy was 0.8868. In this model, the most important variable in predicting asthma was the building density within a buffers of 300 m buffer during pregnancy period (100), the pm10 value during pregnancy (88.815), and the connectivity density within a buffers of 300 m buffer during pregnancy period(number of intersections / km2) (77.481). The top variables for the random forest method were the same as bagging. The lowest feature was the meters of public transport mode lines (only buses) inside each 300 m buffer (0.00).

### LASSO

#LASSO  
#NTS: first create a grid to search lambda  
lambda <- 10^seq(-5,5, length = 100)  
  
set.seed(100)  
  
lasso\_m <- train(  
 asthma ~., data = train\_asthma, method = "glmnet", trControl = trainControl("cv", number = 10, sampling = "down"), preProc = c("center", "scale"), tuneGrid = expand.grid(alpha = 1, lambda = lambda)  
)  
  
#Print the values of alpha and lambda that gave best prediction  
lasso\_m$bestTune %>% knitr::kable() # 1(alpha)|0.004(lambda)| (Accuracy):0.5417

|  | alpha | lambda |
| --- | --- | --- |
| 27 | 1 | 0.0042292 |

# Model coefficients  
coef(lasso\_m$finalModel, lasso\_m$bestTune$lambda)

## 11 x 1 sparse Matrix of class "dgCMatrix"  
## s1  
## (Intercept) 8.990652e-06  
## h\_pamod\_t3\_Nonesometimes\_activity .   
## h\_pamod\_t3\_Noneoften\_activity 1.310652e-01  
## h\_pamod\_t3\_Nonevery\_often 6.660038e-02  
## h\_greenyn300\_preg\_None1 6.897059e-03  
## h\_pm10\_ratio\_preg\_None .   
## h\_accesslines300\_preg\_dic0 -2.644020e-01  
## h\_accesspoints300\_preg\_Log 4.794318e-01  
## h\_builtdens300\_preg\_Sqrt -2.956366e-01  
## h\_connind300\_preg\_Sqrt 5.866106e-02  
## h\_walkability\_mean\_preg\_None 1.141978e-01

#h\_pamod\_t3\_NoneOften and h\_pm10\_ratio\_preg\_None went to 0. The largest beta is h\_accesspoints300\_preg\_Log = 4.794318e-01  
  
#Confusion Matrix  
confusionMatrix(lasso\_m) #Accuracy (average) : 0.5417

## Cross-Validated (10 fold) Confusion Matrix   
##   
## (entries are percentual average cell counts across resamples)  
##   
## Reference  
## Prediction 0 1  
## 0 48.1 4.9  
## 1 40.9 6.0  
##   
## Accuracy (average) : 0.5417

With the LASSO, the average accuracy was 0.5417. The value of lambda that provided the best prediction was 0.0042. The two features that went to 0 in this model were the “sometimes\_activity” level of the walking and/or cycling activity during pregnancy feature and the pm10 value when pregnant. The largest beta was for the number of bus public transport mode stops inside each 300 m buffer which was 4.794318e-01. The smallest beta was for building density which was -2.956366e-01.

### Support Vector Machine

#we are doing an SVM because SVM offers very high accuracy compared to other classifiers such as logistic regression, and decision trees. It is known for its kernel trick to handle nonlinear input spaces.  
  
#add levels to the asthma variable because I'll need it for SVM.  
studydata\_svm = studydata   
  
summary(studydata\_svm) #data is unbalanced. No asthma is 1159 and asthma is 142

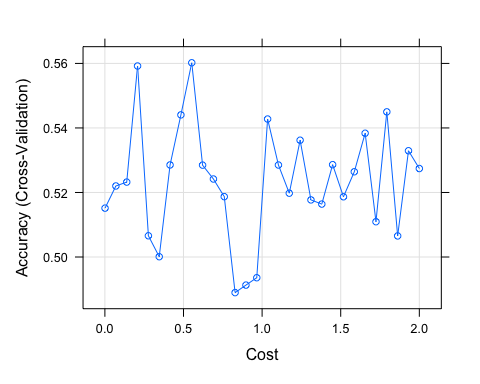
## h\_pamod\_t3\_None h\_greenyn300\_preg\_None h\_pm10\_ratio\_preg\_None  
## no\_activity : 42 0:321 Min. : 8.066   
## sometimes\_activity:474 1:980 1st Qu.:17.535   
## often\_activity :191 Median :23.018   
## very\_often :594 Mean :23.504   
## 3rd Qu.:27.677   
## Max. :47.698   
## h\_accesslines300\_preg\_dic0 h\_accesspoints300\_preg\_Log h\_builtdens300\_preg\_Sqrt  
## Min. :0.0000 Min. :1.270 Min. : 11.02   
## 1st Qu.:0.0000 1st Qu.:1.963 1st Qu.:340.04   
## Median :0.0000 Median :2.879 Median :401.49   
## Mean :0.1991 Mean :2.670 Mean :417.06   
## 3rd Qu.:0.0000 3rd Qu.:3.349 3rd Qu.:502.97   
## Max. :1.0000 Max. :4.528 Max. :807.57   
## h\_connind300\_preg\_Sqrt h\_walkability\_mean\_preg\_None asthma   
## Min. : 1.887 Min. :0.1000 0:1159   
## 1st Qu.: 9.983 1st Qu.:0.2000 1: 142   
## Median :12.935 Median :0.2500   
## Mean :12.737 Mean :0.2674   
## 3rd Qu.:15.898 3rd Qu.:0.3250   
## Max. :27.276 Max. :0.6250

levels(studydata\_svm$asthma) <- c("no\_asthma", "yes\_asthma")  
  
#Set No Asthma as Reference Level  
studydata\_svm$asthma <- relevel(studydata\_svm$asthma, ref = "no\_asthma")  
  
set.seed(100)  
trainsvm.indices <- createDataPartition(y = studydata\_svm$asthma,p = 0.7,list = FALSE)  
trainsvm\_asthma <- studydata\_svm[trainsvm.indices, ]  
testsvm\_asthma <- studydata\_svm[-trainsvm.indices, ]

set.seed(100)  
  
#Set 10-fold cross-validation. Note if you want predicted probabilities, you need to set class Probs=True  
traincontrol\_svm <- trainControl(method = "cv", number = 10, sampling = "down", classProbs = T)  
  
  
#Train model. Note we are scaling data  
svm\_asthma <- train(asthma ~ ., data = trainsvm\_asthma, method = "svmLinear", trControl = traincontrol\_svm, preProcess = c("center", "scale"))  
  
svm\_asthma #Accuracy:0.4879

## Support Vector Machines with Linear Kernel   
##   
## 912 samples  
## 8 predictor  
## 2 classes: 'no\_asthma', 'yes\_asthma'   
##   
## Pre-processing: centered (10), scaled (10)   
## Resampling: Cross-Validated (10 fold)   
## Summary of sample sizes: 821, 821, 820, 821, 821, 821, ...   
## Addtional sampling using down-sampling prior to pre-processing  
##   
## Resampling results:  
##   
## Accuracy Kappa   
## 0.4879001 -0.002886459  
##   
## Tuning parameter 'C' was held constant at a value of 1

#Incorporate different values for cost (C)  
svm\_asthma2 <- train(asthma ~ ., data = trainsvm\_asthma, method = "svmLinear", trControl = traincontrol\_svm, preProcess = c("center", "scale"), tuneGrid = expand.grid(C = seq(0.001,2, length = 30)))  
  
#Visualize accuracy versus values of C  
plot(svm\_asthma2)



#Obtain metrics of accuracy from training  
confusionMatrix(svm\_asthma2) #Accuracy: 0.5603

## Cross-Validated (10 fold) Confusion Matrix   
##   
## (entries are percentual average cell counts across resamples)  
##   
## Reference  
## Prediction no\_asthma yes\_asthma  
## no\_asthma 50.9 5.8  
## yes\_asthma 38.2 5.2  
##   
## Accuracy (average) : 0.5603

#See information about final model  
svm\_asthma2$finalModel

## Support Vector Machine object of class "ksvm"   
##   
## SV type: C-svc (classification)   
## parameter : cost C = 0.552448275862069   
##   
## Linear (vanilla) kernel function.   
##   
## Number of Support Vectors : 179   
##   
## Objective Function Value : -95.3575   
## Training error : 0.415   
## Probability model included.

#Make predictions in testset  
svm\_asthma2\_predtest <- predict(svm\_asthma2, testsvm\_asthma)  
  
#Get evaluation metrics from test set  
confusionMatrix(svm\_asthma2\_predtest, testsvm\_asthma$asthma, positive = "yes\_asthma")

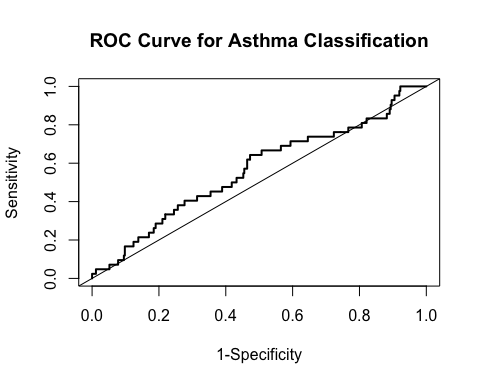
## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction no\_asthma yes\_asthma  
## no\_asthma 124 19  
## yes\_asthma 223 23  
##   
## Accuracy : 0.3779   
## 95% CI : (0.3295, 0.4281)  
## No Information Rate : 0.892   
## P-Value [Acc > NIR] : 1   
##   
## Kappa : -0.0303   
##   
## Mcnemar's Test P-Value : <2e-16   
##   
## Sensitivity : 0.54762   
## Specificity : 0.35735   
## Pos Pred Value : 0.09350   
## Neg Pred Value : 0.86713   
## Prevalence : 0.10797   
## Detection Rate : 0.05913   
## Detection Prevalence : 0.63239   
## Balanced Accuracy : 0.45248   
##   
## 'Positive' Class : yes\_asthma   
##

#Accuracy:0.3779 with a 95% CI of (0.3296, 0.4281) with a sensitivity of 0.54762 and a specificity of 0.35735  
  
#Create ROC Curve for Analysis  
pred.prob <- predict(svm\_asthma2, testsvm\_asthma, type = "prob")  
  
#Another potential evaluation: Area under the Reciver Operating Curve (AUROC)  
analysis <- roc(response = testsvm\_asthma$asthma, predictor = pred.prob[,2])

## Setting levels: control = no\_asthma, case = yes\_asthma

## Setting direction: controls > cases

plot(1 - analysis$specificities,analysis$sensitivities,type = "l",  
ylab = "Sensitivity",xlab = "1-Specificity",col = "black",lwd = 2,  
main = "ROC Curve for Asthma Classification")  
abline(a = 0,b = 1)



#Variable Importance  
varImp(svm\_asthma2)

## ROC curve variable importance  
##   
## Importance  
## h\_builtdens300\_preg\_Sqrt 100.000  
## h\_pm10\_ratio\_preg\_None 92.350  
## h\_accesspoints300\_preg\_Log 88.992  
## h\_accesslines300\_preg\_dic0 48.933  
## h\_walkability\_mean\_preg\_None 32.822  
## h\_connind300\_preg\_Sqrt 14.180  
## h\_pamod\_t3\_None 4.527  
## h\_greenyn300\_preg\_None 0.000

#h\_builtdens300\_preg\_Sqrt : 100.000, h\_pm10\_ratio\_preg\_None: 92.350, h\_accesspoints300\_preg\_Log = 88.992

When we modeled SVM with the training data, the accuracy was 0.4879 However when modeling SVM with a different cost value, the accuracy went up to 0.5603 When I made predictions on the test data using the second SVM model (with the different cost), the accuracy went down to 0.3779 (95% CI: 0.3296, 0.4281) with a sensitivity of 0.54762 and a specificity of 0.35735. The top important features in this model was building density (100), pm10 value (92.350), and the number of bus public transport mode stops inside each 300 m buffer (88.992). The least important features were physical activity (4.527) and access to green space in a distance of 300m (0.00). When looking at the ROC curve, this model is a poor classifier of asthma, meaning that the model will not able to distinguish the two classes well and has caused the ROC curve is be closer to the diagonal random line. A ROC curve that is a better/good classifier of asthma would have been closer to the top left of the graph, demonstrating a high true positive rate at a low false positive rate.

# CONCLUSION

Out of the 4 different prediction models I used, random forest had the highest accuracy. Equally important, specific features in random forest like building density and pm10 value were also important variables in other models like SVM and bagging. I will conduct a random forest model to the testing data to see what the difference is between the accuracies.

set.seed(100)  
  
mtryvals <- c(ncol(train\_asthma) - 1, sqrt(ncol(train\_asthma) - 1), 0.5\*ncol(train\_asthma) - 1)  
  
mtrygrid <- expand.grid(.mtry = mtryvals)  
  
rf\_asthma\_test <- train(asthma ~., data = test\_asthma, method = "rf", metric = "Accuracy", tuneGrid = mtrygrid, ntree = 100)  
  
confusionMatrix(rf\_asthma\_test) #Accuracy (average) : 0.8871

## Bootstrapped (25 reps) Confusion Matrix   
##   
## (entries are percentual average cell counts across resamples)  
##   
## Reference  
## Prediction 0 1  
## 0 88.6 10.0  
## 1 1.3 0.1  
##   
## Accuracy (average) : 0.8871

rf\_asthma\_test$results

## mtry Accuracy Kappa AccuracySD KappaSD  
## 1 2.828427 0.8869099 -0.004630898 0.02614626 0.04135895  
## 2 3.500000 0.8819989 -0.010263127 0.02179440 0.04724797  
## 3 8.000000 0.8737077 -0.018167549 0.02253817 0.05442626

rf\_asthma\_test$bestTune

## mtry  
## 1 2.828427

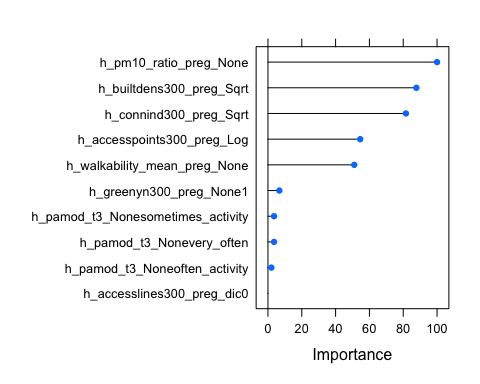
rf\_asthma\_test$finalModel

##   
## Call:  
## randomForest(x = x, y = y, ntree = 100, mtry = min(param$mtry, ncol(x)))   
## Type of random forest: classification  
## Number of trees: 100  
## No. of variables tried at each split: 3  
##   
## OOB estimate of error rate: 11.83%  
## Confusion matrix:  
## 0 1 class.error  
## 0 343 4 0.01152738  
## 1 42 0 1.00000000

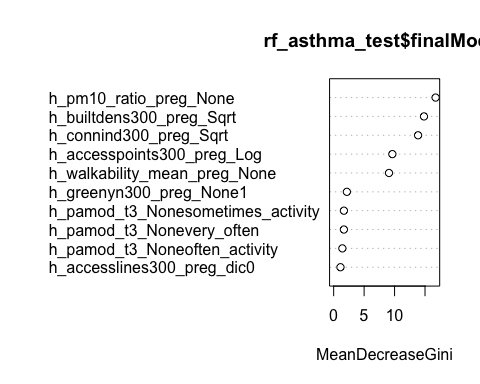
varImp(rf\_asthma\_test)

## rf variable importance  
##   
## Overall  
## h\_pm10\_ratio\_preg\_None 100.000  
## h\_builtdens300\_preg\_Sqrt 87.785  
## h\_connind300\_preg\_Sqrt 81.585  
## h\_accesspoints300\_preg\_Log 54.543  
## h\_walkability\_mean\_preg\_None 51.026  
## h\_greenyn300\_preg\_None1 6.743  
## h\_pamod\_t3\_Nonesometimes\_activity 3.607  
## h\_pamod\_t3\_Nonevery\_often 3.591  
## h\_pamod\_t3\_Noneoften\_activity 2.003  
## h\_accesslines300\_preg\_dic0 0.000

plot(varImp(rf\_asthma\_test))



varImpPlot(rf\_asthma\_test$finalModel)

 With the random forest method using the testing data, the average accuracy was 0.8871 with the best mtry value being 2.828427. In this model, the most important variable in predicting asthma was the pm10 value during pregnancy (100), the building density during the pregnancy period (87.785), and the connectivity density during pregnancy period (81.585). The least important feature was the meters of public transport mode lines (only buses) inside each 300 m buffer (0.00).

By using various algorithms to discover important features when predicting asthma and identifying which model will have the highest accuracy, we are able to use this information to improve health outcomes and treatment plans for children whose mothers were exposed to a poorer built environment during pregnancy. Through this model, we learn that environmental variables such as pm10 value, building density, and connectivity density are connected to asthma in children.