**Data Exploration:**

* The original data set contains 1,997,490 rows and 13 variables.
* Looking into the distribution of target variable, the observations/records are extremely unbalanced. Only 2420 records (0.12%) have conversion (target values 1) out of 1.9 million+ records. So if it is assumed “NO CONVERSION AT ALL” based prediction, the overall accuracy would be 99.88%. So overall accuracy is misleading. We need to increase the prediction accuracy for the POSITIVE CONVERSATION RATE (target value with 1).
* Variable 1 (year) has zero variance. So it has been removed. Even though variable 2 (month) has near zero variance, it has been kept as it has been found that month has increased predictability of the model.

nzv <- nearZeroVar(data)

datawo\_nzv <- data[, -nzv]

**Feature Engineering:**

* Derived a new variable weekday from the atime variable
* If given more time, I would have categorized the site names. For that I need to call API to categorize the sites (as for example: simillarweb.com allows calling API to categorize the sites)

**Training and Evaluation Methodology:**

* Since the data set is extremely unbalanced from the distribution of target variables, training set has been sampled with approx. 1:1 ratio of target variables having 0 and 1. That means I’ve used under-sampling (deleted instances from the over-represented class) in the training set
* I’ve split the dataset into 3 sets – ensembleData, blenderData and testingData
* blenderData and testingData set have been used to apply ensemble approach to improve prediction accuracy by combining the prediction from 3 different models. However, no improvement observed.
* I’ve used confusion matrix to evaluate the overall accuracy as well as Precision (the number of True Positives divided by the number of True Positives and False Positives)

**Algorithm:**

* **Naïve Bayes –** simplest Bayesian approach – just for benchmark

**Overall accuracy –** 0.9989502

Precision - 0.4066986 ( very poor!!)

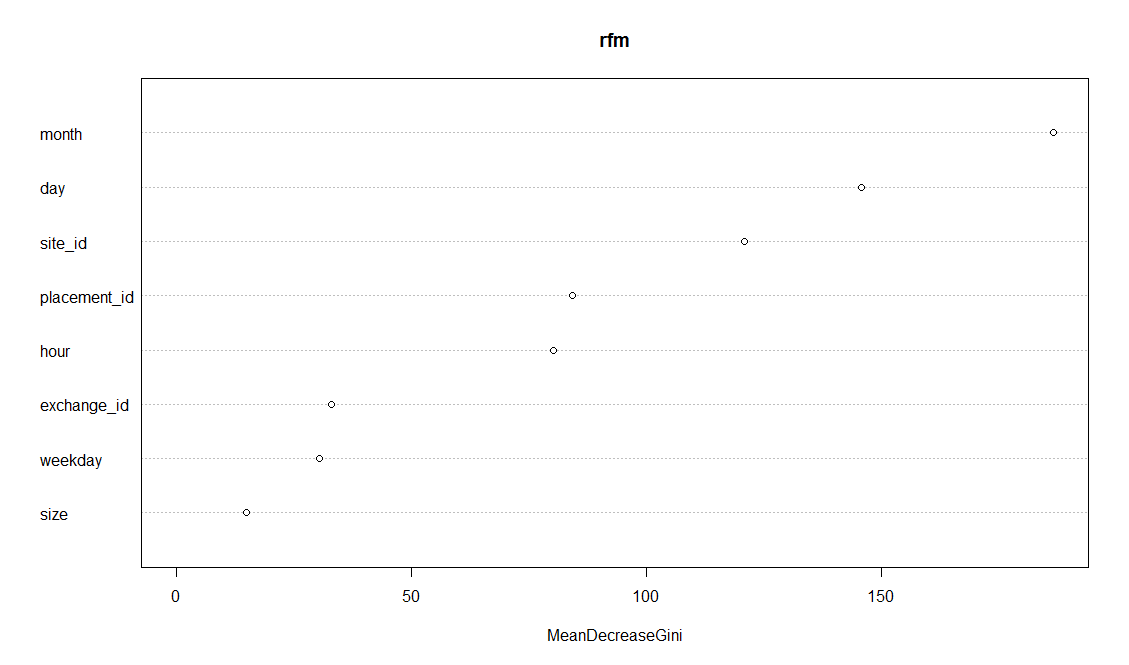
* **RandomForest**

1. Since decision trees based algorithms often perform well on imbalanced datasets. The splitting rules that look at the class variable used in the creation of the trees, can force both classes to be addressed.
2. It also does bootstrapping internally via randomizing sample with replacement to avoid overfitting
3. It has an excellent ensemble approach via majority voting at multiple decisions trees leaf nodes.

**Overall accuracy –** 0.8997332 (decreased compared to naïve Bayes. However, most important metric is precision)

Precision - 0.8115824 (increased a lot compared to Naïve Bayes!)

varImpPlot(rfm)



**GBM:** gradient boosted method as it tries to improve prediction after each iteration.

**Overall accuracy –** 0.910729

Precision - 0.8091354 (decreased compared to random Forest!)

* I also tried GBM, RANDOMFOREST and CART to apply ensemble based prediction. However, ensemble didn’t show any prediction improvement due to collinearity among the predicted values from the 3 models individually.

**Runtime**: To run the entire R code, it took less than 5 minutes using a single machine with 6GB memory and 4 CPU.