

Final Presentation - The Gould-en Rule

Stats 101C Lecture 3

Andy Shen, Ethan Allavarpu

Fall 2020

Section 1

Introduction

Introduction

- With the rise in popularity of YouTube, many people are now making a living off creating YouTube videos
- The more views gained by the video, the more likely it is for that channel to profit
- We are interested in predicting the growth rate in video views between the **second** and **sixth** hour that a YouTube video is published

Section 2

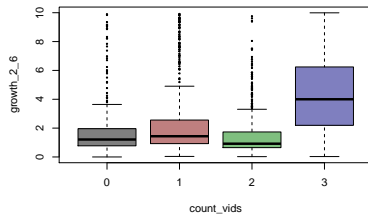
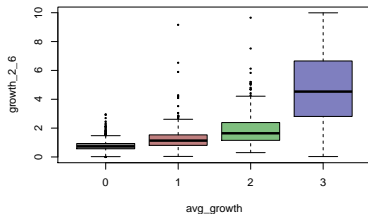
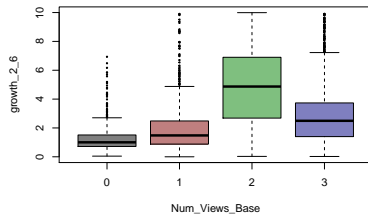
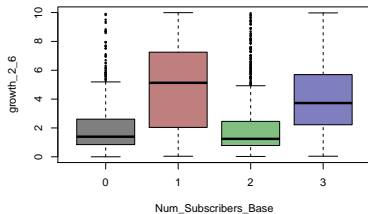
Methodology

Subsection 1

Preprocessing

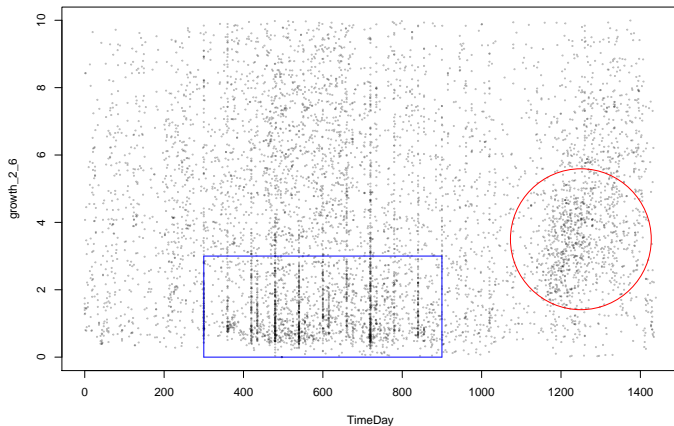
Feature Transformation

- Combined binary variables into a single factor with four levels



Feature Expansion

- TimeDay: What time in the day was a video published? (made continuous)



- Small upward cluster toward 1200 (lower clusters earlier in the day)

- Examined a univariate plot to look for stray points and removed them systematically
 - ▶ Based off personal judgment and inference on the effect of the stray points
- We also remove highly correlated variables as indicated by a heat map
 - ▶ To avoid overfitting based on having too many predictors

Predictor Selection

- We use LASSO to select significant predictors
 - ▶ LASSO pushes the coefficients of non-significant predictors to zero
 - ▶ Keeps the most significant ones
- First, fit a LASSO model for a sequence of candidate λ values
- Then select our optimal value of λ as the one that resulted in the lowest test MSE
- Extract the predictors with nonzero coefficients in the LASSO model as our predictors for the candidate model

Subsection 2

Statistical Model

- Most of our models were fit using bagging or random forest
 - ▶ Only adjusted certain parameters at a time (number of trees, depth, and λ)
- Preliminary least-squares model
 - ▶ Kaggle score of ~ 1.65

Candidate Model: Random Forest

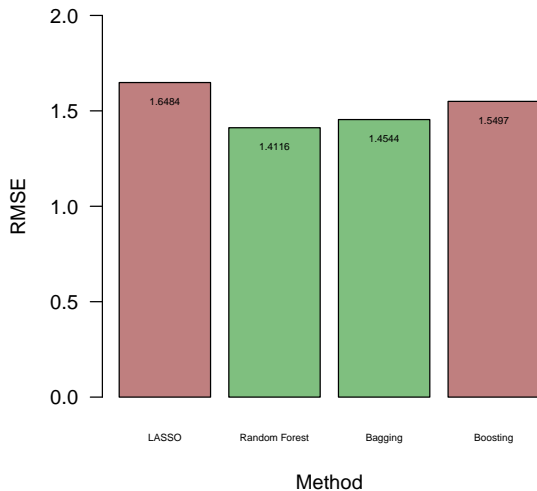
- After running LASSO to refine predictors, we use this smaller subset to find m to use for random forest
 - ▶ m : Number of variables randomly considered in each node of each decision tree
 - ▶ Find optimal m with 5-fold cross-validation: select m corresponding to the lowest *median* RMSE of the 5 folds
 - ★ Median is more preferable than mean due to the mean's sensitivity to extreme points
- Once optimal m is selected, fit random forest model to 80% of the preprocessed training data
 - ▶ Extra model and validation RMSE ensures consistent performance

Candidate Model: Bagging

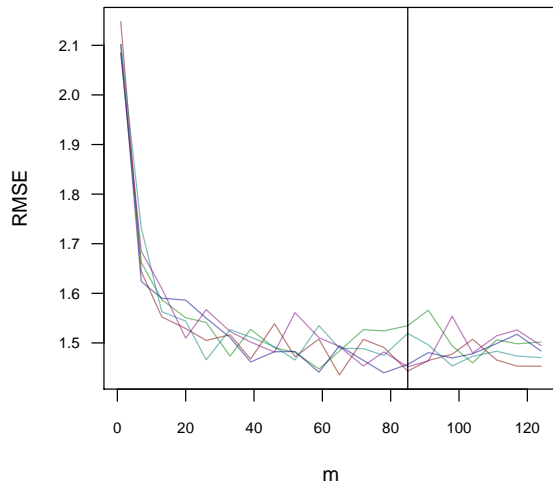
- Considered a bagging approach ($m = p$) as a secondary model to random forest ($m = p$)

RMSEs

RMSEs for Various Approches



Best m for Random Forest



Section 3

Results

Results

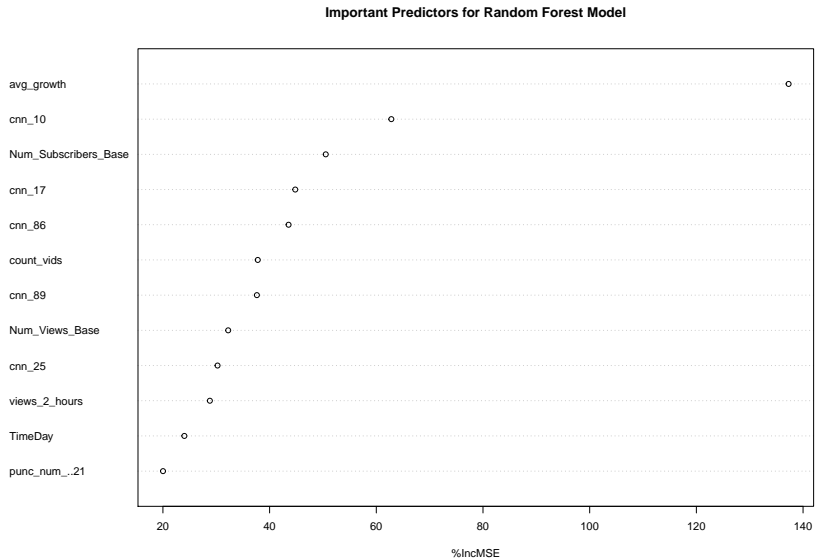
Kaggle scores:

Ⓐ 1.39753

Ⓑ 1.40285

Model (B) (Model 15b) only differs from (A) (Model 15d) in that $m = p$ as opposed to m equaling the value with the lowest median RMSE. Here p is the number of predictors.

Variable Importance



Section 4

Conclusion

Conclusion

- We believed our model performed well due to the fact that it works as an ensemble method
 - ▶ Combines multiple individual models to get more accurate responses
- By using cross-validation for our selection of m , we limit the potential effect of a random seed showing us an inaccurately good or bad RMSE
- TimeDay custom variable is important (11th out of 120+) - creating our own variables helped