Baseline Modeling

Tyler Marshall

10/12/2022

```r
data <- read.csv("priv_mcare_f_pay.csv")
hospital_data <- read.csv("Hospital_Master_Sheet.csv")

hospitals_msa <- hospital_data %>%
  group_by(MSA_CD) %>%
  summarise(Hospitals = n(),
            PctTeaching = sum(teaching == "YES")/n(),
            PctLargeHospital = sum(beds_grp == "500+")/n(),
            Urban = ifelse(sum(urban_rural == "URBAN")/n() == 1, "Urban","Rural"),
            PctPrivate = sum(ownership == "PRIVATE (NOT FOR PROFIT)" | ownership == "PRIVATE (FOR PROFIT
  rename(msa = MSA_CD)

new_data <- data %>%
  filter(priv_pay_median >= 0 | is.na(priv_pay_median)) %>%
  filter(priv_count != 0)

new_data_with_hospital <- left_join(new_data, hospitals_msa, by = "msa")


model_data <- new_data_with_hospital %>%
  filter(priv_count >= 50) %>%
  filter(!is.na(priv_pay_median)) %>%
  select(-priv_pay_mean, -priv_pay_iqr)

predict_data <- new_data_with_hospital %>%
  filter(priv_count <= 10) %>%
  filter(priv_pay_median > 0)

set.seed(123) #Set seed for reproducible analysis
dt = sort(sample(nrow(model_data), nrow(model_data)*.8)) #Split data
train <-model_data[dt,] #80% training data
test <-model_data[-dt,] #20% test data



# Random Forest model
set.seed(123) #Set seed for reproducibility
# Fit Random Forest Model on training data
Random_Forest <- randomForest(
  formula = priv_pay_median ~ .,
  data    = train,
  num.trees = 500,
```

```
  mtry = 7,
  nodesize = 20,
  na.action = na.omit
)

train_predict <- train %>%
  mutate(pred_priv_pay_median = predict(Random_Forest, train)) %>%
  filter(!is.na(pred_priv_pay_median))
MAPE(train_predict$pred_priv_pay_median, train_predict$priv_pay_median)
```
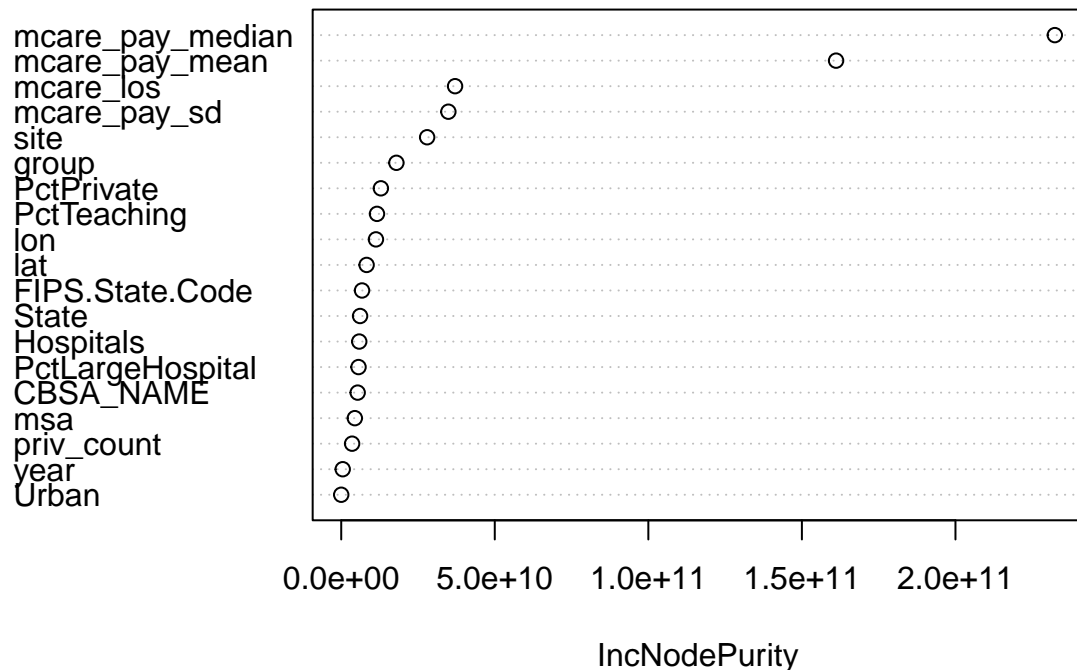
[1] 0.1575888

```
mean(abs((train_predict$priv_pay_median - train_predict$pred_priv_pay_median)/train_predict$priv_pay_med
```

[1] 15.75888

```
varImpPlot(Random_Forest)
```

Random_Forest

IncNodePurity

```
test_predict <- test %>%
  mutate(pred_priv_pay_median = predict(Random_Forest, test)) %>%
  filter(!is.na(pred_priv_pay_median))
mean(abs((test_predict$priv_pay_median - test_predict$pred_priv_pay_median)/test_predict$priv_pay_median
```

[1] 21.78875

```
MAPE(test_predict$pred_priv_pay_median, test_predict$priv_pay_median)
```

[1] 0.2178875