Hold Out Method vs. Cross Vaildation

Tadros Salama

2/17/2021

This analysis uses 3 linear models performances on predicting sales price of a home to compare different testing and validating processes. The first method will be the hold out method approach for `lm1, lm2, lm3`, and the second, a 10-fold cross validation using `train()` from the R package (caret)[https://cran.r-project.org/web/packages/caret/caret.pdf] on `cv_lm1, cv_lm2, cv_lm3`.

Data from R package (AmesHousing)[https://github.com/topepo/AmesHousing]

Spliting data into 70% train and 30% test

```
set.seed(123)
train_index <- sample(1:nrow(ames), round(nrow(ames) * 0.7))
ames_train <- ames[train_index, ]
ames_test <- ames[-train_index, ]
```

Model 1, single predictor variable - total rooms in a house

Model 2, two predictor variables - total rooms & year the house was built

Model 3, three perdictor variables- total rooms above ground, year built, and overall condition of house

```
lm1 <- lm(Sale_Price ~ Gr_Liv_Area + Year_Built, ames_train)
lm2 <- lm(Sale_Price ~ Gr_Liv_Area + Year_Built + TotRms_AbvGrd, ames_train)
lm3 <- lm(Sale_Price ~ Gr_Liv_Area + Year_Built + TotRms_AbvGrd + Overall_Cond, ames_train)
```

```
pred_lm1 <- predict(lm1, newdata = ames_test)
pred_lm2 <- predict(lm2, newdata = ames_test)
pred_lm3 <- predict(lm3, newdata = ames_test)

RMSE <- function(y, y_hat) {
  sqrt(mean((y - y_hat)^2))
}

RMSE(ames_test$Sale_Price, pred_lm1)
```

```
## [1] 45444.77
```

```
RMSE(ames_test$Sale_Price, pred_lm2)
```

```
## [1] 44950.55
```

```
RMSE_lm3 <- RMSE(ames_test$Sale_Price, pred_lm3)
```

```r
MAE <- function(y, y_hat) {
  mean(abs(y - y_hat))
}

MAE(ames_test$Sale_Price, pred_lm1)
```

```
## [1] 31916.6
```

```r
MAE(ames_test$Sale_Price, pred_lm2)
```

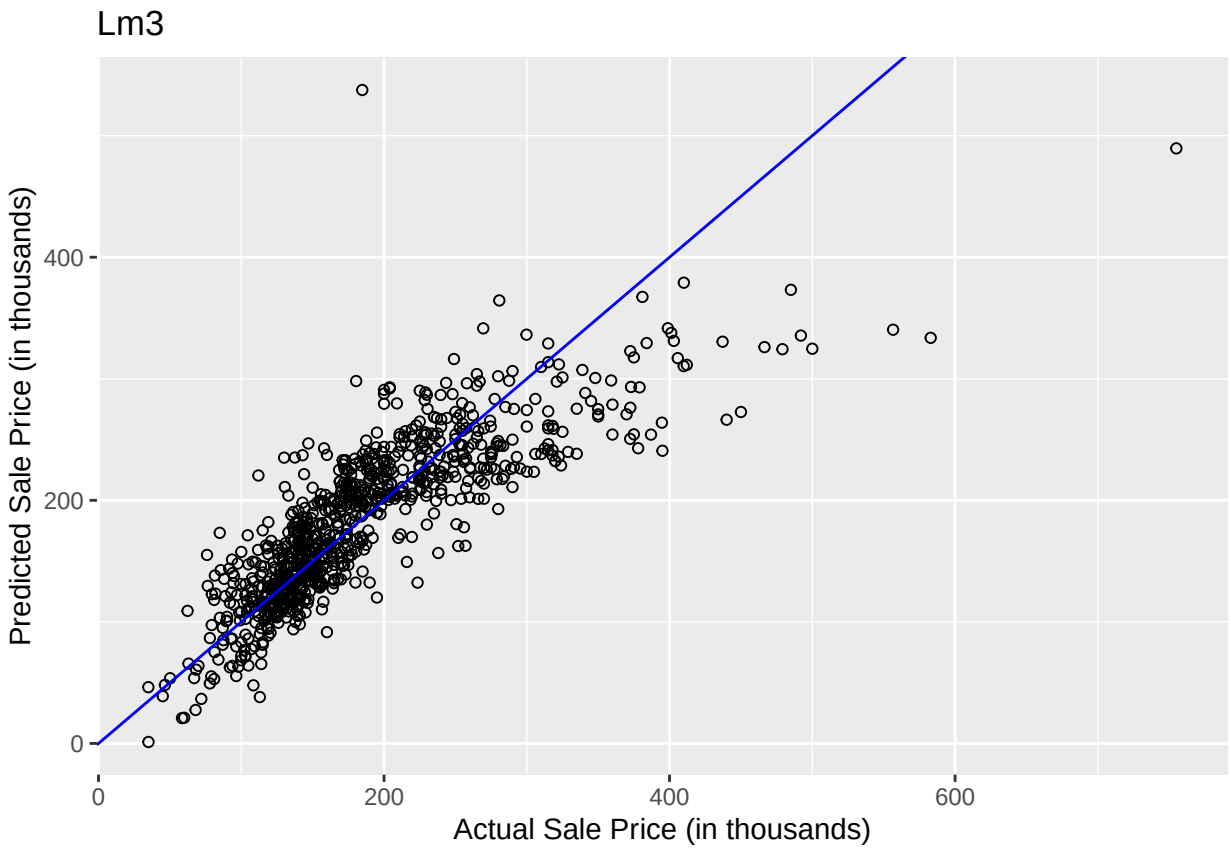
```
## [1] 31654.69
```

```r
MAE(ames_test$Sale_Price, pred_lm3)
```

```
## [1] 30368.43
```

```r
df_pred_lm3 <- data.frame(
  actual = ames_test$Sale_Price / 1000,
  predLm3 = pred_lm3 / 1000
  )

ggplot(df_pred_lm3, aes(actual, predLm3)) +
  geom_point(shape =1) +
  geom_abline(intercept = 0, slope = 1, color = 'blue') +
  xlab("Actual Sale Price (in thousands)") +
  ylab("Predicted Sale Price (in thousands)") +
  ggtitle("Lm3")
```



Using Cross-Validiation to compare model perfromance

Model 4, single predictor variable - total rooms in a house

Model 5, two predictor variables - total rooms & year the house was built

Model 6, three perdictor variables- total rooms above ground, year built, and overall condition of house

```
set.seed(123)
cv_lm1 <- train(
  Sale_Price ~ TotRms_AbvGrd,
  data = ames,
  method = 'lm',
  trControl = trainControl(method = 'cv', number = 10)
)
cv_lm1
```

```
## Linear Regression
##
## 2930 samples
##    1 predictor
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 2637, 2637, 2637, 2636, 2637, 2638, ...
## Resampling results:
##
##   RMSE      Rsquared   MAE
##   69347.21  0.2481994  49475.09
##
## Tuning parameter 'intercept' was held constant at a value of TRUE
```

```
set.seed(123)
cv_lm2 <- train(
  Sale_Price ~ TotRms_AbvGrd + Year_Built,
  data = ames,
  method = 'lm',
  trControl = trainControl(method = 'cv', number = 10)
)
cv_lm2
```

```
## Linear Regression
##
## 2930 samples
##    2 predictor
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 2637, 2637, 2637, 2636, 2637, 2638, ...
## Resampling results:
##
##   RMSE      Rsquared   MAE
##   56332.97  0.5049611  38953.76
##
## Tuning parameter 'intercept' was held constant at a value of TRUE
```

```r
set.seed(123)
cv_lm3 <- train(
  Sale_Price ~ TotRms_AbvGrd + Year_Built + Overall_Cond,
  data = ames,
  method = 'lm',
  trControl = trainControl(method = 'cv', number = 10)
)
```

```
## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading
```

```r
cv_lm3
```

```
## Linear Regression
##
## 2930 samples
##    3 predictor
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 2637, 2637, 2637, 2636, 2637, 2638, ...
## Resampling results:
##
##   RMSE      Rsquared   MAE
##   55047.42  0.5275418  38017.46
##
## Tuning parameter 'intercept' was held constant at a value of TRUE
```