# Prediction: Multiple Linear Regression
# Part 2: Variable Selection in Linear Regression

Aram Balagyozyan

Department of Operations and Information Management
Kania School of Management
The
University of Scranton

January 6, 2020

# 1. Variable Selection in Linear Regression

# Reducing the Number of Predictors

- In deciding on what predictors to include in a regression equation, one may be tempted to use the kitchen-sink approach. After all, not including predictors that are actually correlated with the outcome variable can increase the **bias** of prediction.
- However, there are several compelling arguments against doing that.
  1. It can be shown that using predictors that are uncorrelated with the outcome variable increases the **variance** of predictions.
  2. When two or more predictors share the same linear relationship with the outcome variable (that is when they are correlated with each other) then the regression coefficient estimates of those correlated predictors turn out to be unstable and imprecise. This problem is referred to as *multicolinearity*.
  3. The more predictors, the higher the chance of missing values in the data.
  4. In some settings (e.g. surveys), we may be able to measure fewer predictors more accurately.
  5. It may be expensive or not feasible to collect a full set of predictors for future predictions.

# How to Reduce the Number of Predictors

- The first step in trying to reduce the number of predictors should always be to use domain knowledge.
- It is important to understand what the various predictors are measuring and why they are relevant for predicting the outcome variable.
- Some practical reasons for predictor elimination are
  1. the expense of collecting this information in the future
  2. measurement inaccuracy
  3. high correlation with another predictor
  4. many missing values
  5. irrelevance
- It is often helpful to examine potential predictors using summary statistics and graphs, such as frequency and correlation tables, predictor-specific summary statistics and plots, and missing value counts.

# Two General Approaches to Reducing the Number of Predictors

- In general, there are two types of methods for reducing the number of predictors in a model.
  1. Exhaustive search for the "best" subset of predictors. This is usually accomplished by fitting regression models with all the possible combinations predictors. This approach is often time consuming, tedious, and unstable.
  2. Partial, stepwise search. This approach relies on a partial, iterative search through the space of all possible regression models.
- We consider each of these approaches next.

# Exhaustive Search

- Exhaustive search evaluates all possible subsets of predictors. With 3 predictors, this approach involves running 1 model with all 3 predictors, 3 models with 2 predictors, and 3 models with 1 predictor. Thus, with 3 predictors we need to estimate 7 models. With 4 predictors, we need to estimate 20 regressions. So the number of estimated models becomes exceedingly large even with a moderate number of predictors.
- After creating all possible models, we need a criterion for evaluating and comparing the models. Several criteria for evaluating and comparing models are based on metrics computed from the training data. Some of the most popular criteria are:
  1. Adjusted $R^2$
  2. Akaike Information Criterion (AIC)
  3. Schwartz's Bayesian Information Criterion (BIC)
  4. Mallow's $C_p$

# Adjusted $R^2$

- The formula of adjusted $R^2$ is the following:

$$R^2_{adj} = 1 - \frac{n-1}{n-p-1}(1 - R^2)$$

  where $R^2$ is the proportion of explained variability in the model, `p` is the number of predictors, and `n` is the number of observation in the dataset.
- Like $R^2$, higher values of ${R^2}_{adj}$ indicate better fit. Unlike $R^2$, which does not account for the number of predictors used, ${R^2}_{adj}$ uses a penalty on the number of predictors. This avoids the artificial increase in $R^2$ that can result from simply increasing the number of predictors but not the amount of information.

# AIC and BIC

- AIC and BIC measure the goodness of fit of a model, but also include a penalty that is a function of the number of parameters in the model.
- AIC and BIC are estimates of prediction error based in information theory.
- Both compare various models for the same data set.
- Models with smaller AIC and BIC values are considered better.
- BIC penalizes model complexity more heavily than AIC. Thus in most cases, BIC will choose a simpler model that what AIC will choose.

# Mallow's $C_p$

- This criterion assumes that the full model (with all predictors) is unbiased, although it may have predictors that if dropped would reduce prediction variability.
- With this assumption, we can show that if a subset model is unbiased, the average $C_p$ value equals `p + 1` (= number of predictors + 1), the size of the subset.
- A reasonable approach to identifying subset models with small bias is to examine those with values of $C_p$ that are near `p + 1`.
- Good models are those that have values of $C_p$ near `p + 1` and that have small `p`.
- $C_p$ is computed from the formula

$$C_p = \frac{SSE}{\hat{\sigma}^2_{full}} + 2(p+1) - n$$

where `SSE` is the sum of squared errors from the subset model, `p` is the number of predictors, `n` is the number of observations, and $\hat{\sigma}^2_{full}$ is the variance of errors obtained from the model with all predictors.

# Exhaustive Predictor Search in Action

- In order to perform an exhaustive variable search we will rely on the `regsubsets()` function of the **leaps** package.

```r
library(leaps)
search <- regsubsets(Price ~ ., data = train.df,
            nbest = 1, nvmax = dim(train.df)[2],
            method = "exhaustive")
sum<-summary(search)
```

- Note that the textbook (Data Mining for Business Analytics by Shmueli et al.), goes through an extra step of converting the `Fuel_Type` categorical variable into a a set of 3 dummy variables. With the new version of the **leaps** package, this is no longer required.

# Examining the Results of `regsubsets()`

- To see the results of the `regsubsets()` algorithm, execute the following code:

```
sum$which
```

| | (Intercept) | Age_08_04 | KM | Fuel_TypeDiesel | Fuel_TypePetrol | HP | Met_Color | Automatic | CC | Doors | Quarterly_Tax | Weight |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 3 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 4 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| 5 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| 6 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| 7 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| 8 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| 9 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 |
| 10 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| 11 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

- If you want your model to include only one predictor (other than the intercept), then that should be `Age_08_04`. If you wanted to include 2 predictors then those should be `Age_08_04` and `Weight`. If you think that including 3 predictors is more appropriate, then those should be `Age_08_04`, `Weight`, and `KM`. And so on.
- But how many predictors to include?

# How Many Variables to Include?

- In order to determine the appropriate number of variables, consider examining the ${R^2}_{adj}$ and Mallow's $C_p$.

```
sum$adjr2
```

```
## [1] 0.753 0.794 0.843 0.862 0.865 0.868 0.869
0.868 0.868 0.868 0.868
```

```
sum$cp
```

```
## [1] 522.34 334.81 115.90 29.80 19.32 5.17 4.96
6.26 8.08 10.01
## [11] 12.00
```

- ${R^2}_{adj}$ picks the 7th model with 7 predictors.
- As mentioned before, good models are those that have values of MAllow's $C_p$ near the `number of predictors` + 1 and that have small `p`. Thus, $C_p$ would choose model 1 (with 1 predictor) if its value was near 2, model 2 (with 2 predictors) if its value was near 3, and so on. It seems there is a significant drop in $C_p$ at the model with 6 predictors.

# How Many Variables to Include?

- The above 2 slides imply that if our objective was to choose a model with an optimal number of predictors using an exhaustive search, then we would go with either $\mathrm{C_p}$'s recommendation to include 6 predictors:

$$\begin{aligned} Price = \beta_0 + \beta_1 Age\_08\_04 + \beta_2 KM + \beta_3 HP + \beta_4 Quarterly\_Tax \\ + \beta_5 Weight + \beta_6 Fuel\_TypePetrol \end{aligned}$$

or with $\mathrm{R^2}_{\mathrm{adj}}$'s recommendation to include 7 predictors

$$\begin{aligned} Price = \beta_0 + \beta_1 Age\_08\_04 + \beta_2 KM + \beta_3 HP + \beta_4 Quarterly\_Tax \\ + \beta_5 Weight + \beta_6 Fuel\_TypeDiesel + \beta_7 Fuel\_TypePetrol \end{aligned}$$

# Popular Stepwise Selection Algorithms

- The exhaustive search mechanism is preferable when there is a moderate number of predictors. However, as the number of predictors grows, the computation costs may grow prohibitively large. In those cases, you may opt for partial, stepwise search through the space of all possible regression models.
- The stepwise partial search methods are computationally cheaper but have the potential of missing "good" combinations of methods.
- Three stepwise search algorithms are
  1. Forward selection
  2. Backward elimination
  3. Bidirectional selection
- Let's quickly go over each.

# Forward Selection

- This algorithm starts with no predictors and then adds them one by one.
- Each added predictor is the one that has the largest contribution to $R^2$ on top of the predictors that are already in it. An alternative approach to using $R^2$ as a metric, the process may add a predictor if it is statistically significant.
- The algorithm stops when the contribution of additional predictors doesn't increase the $R^2$ (or if there are no more statistically significant predictors).
- Shortcoming: will miss some pairs of variables that perform well together but perform poorly as single predictors.

# Backward Elimination

- This algorithm starts with all predictors and then at each step eliminates the least useful (least statistically significant) predictor.
- The algorithm stops when all all the remaining predictors have statistically significant contributions.
- Shortcoming: Computing the initial model with all the predictors can be time consuming and unstable.

# Bidirectional Search

- This algorithm is a combination of the forward and backward algorithms.
- At each step, it looks for variables to be included or excluded from the model
- Shortcoming: This algorithm can be as time- and resource-consuming as the exhaustive search algorithm.

# Performing Forward, Backward, or Bidirectional Search in R.

- R has several libraries with stepwise functions.
- `regsubsets()` in the **leaps** package implements (in addition to exhaustive search) forward, backward, and bidirectional selection. You can change the desired method by specifying the `method` option of the function. There is nothing in the documentation of `regsubsets()` about the criterion ($R^2$, ${R^2}_{adj}$, or $C_p$) the search algorithm uses. I can only guess, it uses a combination of algorithms.
- Function `step()` of the base R package chooses the model for which AIC is equal to Mallows' $C_p$ (another model optimality condition).
- An example of code using the `step()` function with backward, forward, and bidirectional search is given on the following slide.

## `step()` with backward Search

```r
car.lm.step.bw <- step(car.lm, direction = "backward")
summary(car.lm.step.bw)$coefficients# disp. coef.
summary(car.lm.step.bw)$adj.r.squared
```

| | Estimate | Std. Error | t value | Pr(>\|t\|) |
|---|---|---|---|---|
| (Intercept) | -4622.47 | 1634.08 | -2.829 | 0.005 |
| Age_08_04 | -133.132 | 4.859 | -27.398 | 0 |
| KM | -0.021 | 0.002 | -9.267 | 0 |
| Fuel_TypeDiesel | 888.55 | 596.236 | 1.49 | 0.137 |
| Fuel_TypePetrol | 2138.334 | 571.475 | 3.742 | 0 |
| HP | 37.609 | 5.155 | 7.295 | 0 |
| Quarterly_Tax | 12.979 | 2.599 | 4.994 | 0 |
| Weight | 15.962 | 1.454 | 10.98 | 0 |
| Adj. R-square | 0.869 | | | |

- As you can see, the `backward` search algorithm returned the same seven predictors and same $\mathrm{R}^2{}_{\mathrm{adj}}$ as the exhaustive search algorithm that relied on $\mathrm{R}^2{}_{\mathrm{adj}}$ for the optimality criteria.

# step() with forward Search

```r
car.lm.step.fw <- step(car.lm, direction = "forward")
summary(car.lm.step.fw)$coefficients# disp. coef.
summary(car.lm.step.fw)$adj.r.squared
```

| | Estimate | Std. Error | t value | Pr(>\|t\|) |
|---|---|---|---|---|
| (Intercept) | -4754.38 | 1661.72 | -2.861 | 0.004 |
| Age_08_04 | -133.272 | 4.902 | -27.187 | 0 |
| KM | -0.021 | 0.002 | -9.111 | 0 |
| Fuel_TypeDiesel | 896.206 | 603.164 | 1.486 | 0.138 |
| Fuel_TypePetrol | 2191.368 | 575.629 | 3.807 | 0 |
| HP | 37.258 | 5.233 | 7.119 | 0 |
| Met_Color | 51.315 | 123.395 | 0.416 | 0.678 |
| Automatic | 63.568 | 262.282 | 0.242 | 0.809 |
| CC | 0.011 | 0.098 | 0.11 | 0.912 |
| Doors | -55.7 | 63.966 | -0.871 | 0.384 |
| Quarterly_Tax | 13.08 | 2.608 | 5.015 | 0 |
| Weight | 16.22 | 1.527 | 10.622 | 0 |
| Adj. R-square | 0.868 | | | |

- The forward search algorithm selected all 11 predictors and produced an $R^2_{adj}$ of 0.868 that is lower than the $R^2_{adj}$ of the exhaustive and backward searches.

# step() with Bidirectional Search

```r
car.lm.step.bt <- step(car.lm, direction = "both")
summary(car.lm.step.bt)$coefficients# disp. coef.
summary(car.lm.step.bt)$adj.r.squared
```

| | Estimate | Std. Error | t value | Pr(>\|t\|) |
|---|---|---|---|---|
| (Intercept) | -4622.47 | 1634.08 | -2.829 | 0.005 |
| Age_08_04 | -133.132 | 4.859 | -27.398 | 0 |
| KM | -0.021 | 0.002 | -9.267 | 0 |
| Fuel_TypeDiesel | 888.55 | 596.236 | 1.49 | 0.137 |
| Fuel_TypePetrol | 2138.334 | 571.475 | 3.742 | 0 |
| HP | 37.609 | 5.155 | 7.295 | 0 |
| Quarterly_Tax | 12.979 | 2.599 | 4.994 | 0 |
| Weight | 15.962 | 1.454 | 10.98 | 0 |
| Adj. R-square | 0.869 | | | |

- Bidirectional search returned the same 7 predictors and $\text{R}^2{}_{\text{adj}}$ as the exhaustive and backward search algorithms.

# Evaluating the Out-of-sample Forecast Performance of Different Models

- Given the known issues with the forward search algorithms and given the fact that the exhaustive, backwards, and bidirectional algorithms all returned the same model with the same seven predictors (or six, when $C_p$ was used as an optimality condition), one would naturally lean toward selecting the model with seven predictors.
- However, we still don't know how the models will compare in terms of their out-of-sample forecast accuracy.
- The next slide we will demonstrate how using the validation set, we can check that as well.

# Evaluating the Out-of-sample Forecast Performance of Different Models

- First obtain the accuracy of the out-of-sample forecast performance of the model with all 11 predictors:

```
library(forecast)
car.lm.step.fw.pred<-
          predict(car.lm.step.fw,valid.df)
accuracy(car.lm.step.fw.pred, valid.df$Price)
```

```
##              ME RMSE  MAE   MPE MAPE
## Test set 19.6 1325 1049 -0.75 9.35
```

- Now do the same for the model with 7 predictors:

```
library(forecast)
car.lm.step.bw.pred<-
          predict(car.lm.step.bw,valid.df)
accuracy(car.lm.step.bw.pred, valid.df$Price)
```

```
##              ME RMSE  MAE    MPE MAPE
## Test set 20.4 1328 1055 -0.736 9.41
```

# Evaluating the Out-of-sample Forecast Performance of Different Models

- It seems that although the 7-predictor model provides us with a tighter fit within the training set, in terms of out-of-sample forecasting performance, it doesn't do as well as the 11-predictor model.
- Thus, if forecasting was our goal, we would still go with the 11-predictor model.
- On the other hand, if our goal was to better understand the training set then we would use the 7-predictor model.