**DSC 423: Data Analysis and Regression** Assignment 5:

Name: Adarsh Shankar

Student ID: 2117611

Honor Statement: “I have completed this work independently. The solutions given are entirely my own work.”

Question 1: Short Essay. The purpose of k-fold cross validation is often misunderstood. a. (10 points) How do you use cross validation to select a final (or production) model? Note: it is not the “best” of the k models you have built using cross validation.

Answer: K-fold cross validation is a method for evaluating how well a model can be trained on a set of data and then used to predict new data. We can perform k-fold cross validation using the 80/20 split by training the model k times on 80% of the data and testing on the remaining 20%. The 20% test set contained Each piece of information only appears once. Instead of creating them from scratch, the technique known as cross-validation is utilised to evaluate them. Cross-validation is used to determine which model performs better before we educate it. We don't employ the model instances we trained during cross-validation in our final prediction model.

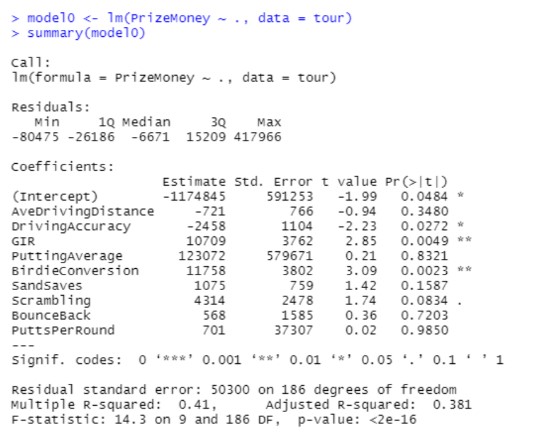
Question 2: PGA. The pgatour2006.csv dataset contains data for 196 players. The variables in the dataset are:

* Player’s name
* PrizeMoney = average prize money per tournament
* DrivingAccuracy = percent of times a player is able to hit the fairway with his tee shot
* GIR = percent of time a player was able to hit the green within two or less than par (Greens in Regulation)
* BirdieConversion = percentage of times a player makes a birdie or better after hitting the green in regulation
* PuttingAverage = putting performance on those holes where the green was hit in regulation.
* PuttsPerRound= average number of putts per round (shots played on the green)
* Etc.

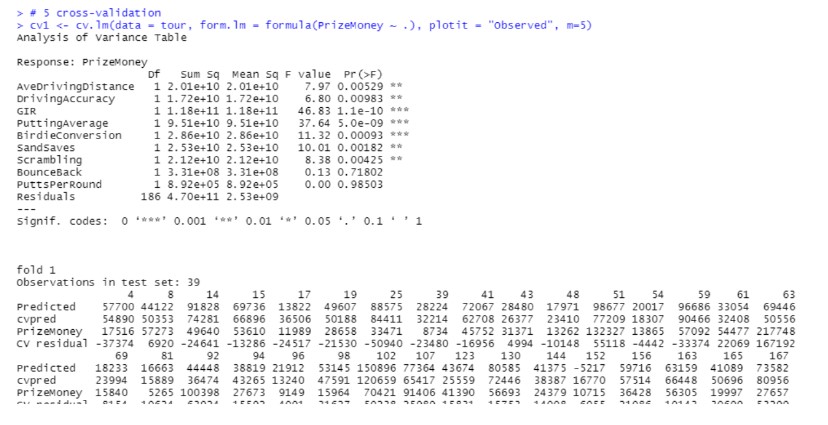
a. (10 points) Build a complete first-order model. Evaluate the model using 5-fold cross validation. If necessary, remove a non-significant variable and repeat until you have your final first-order model. Present the model.

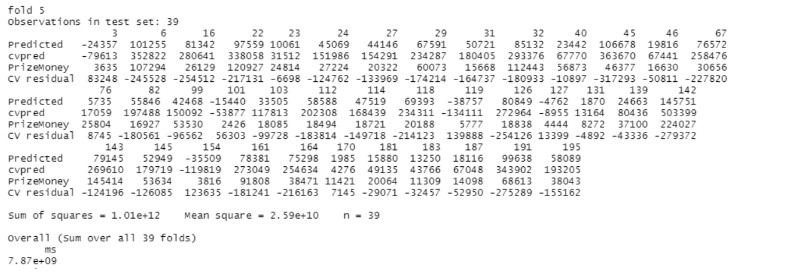
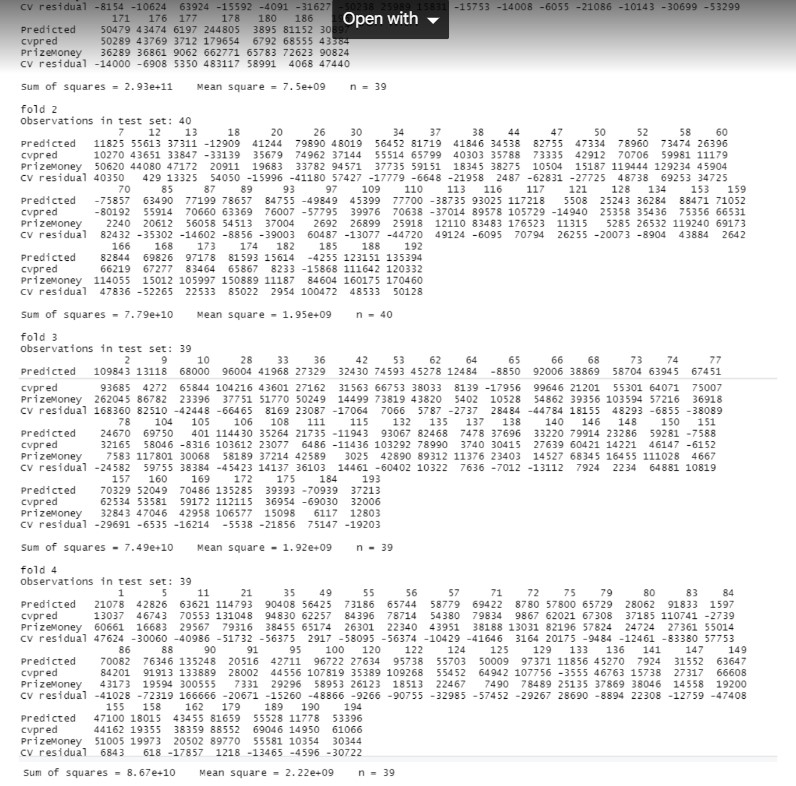
Solution:

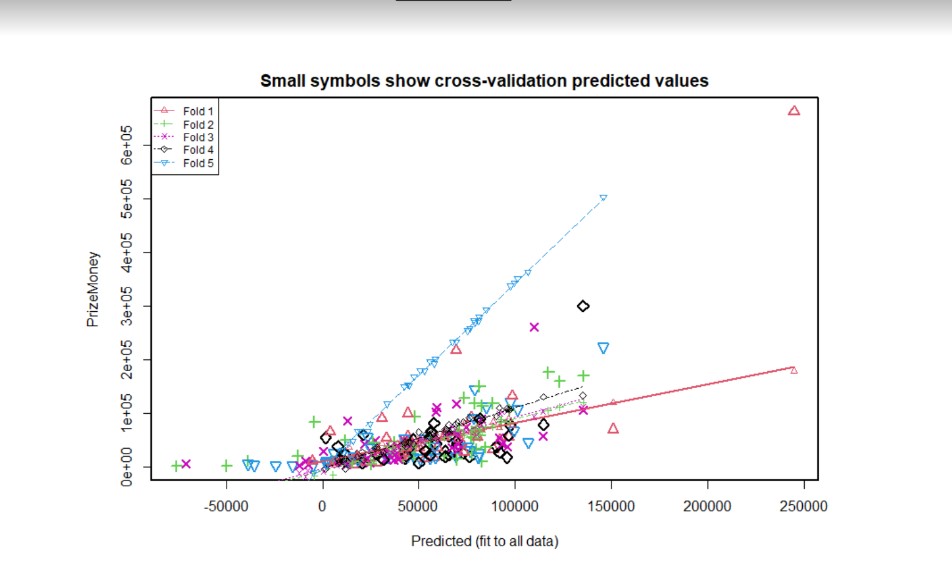
Initialize first model with all the variables



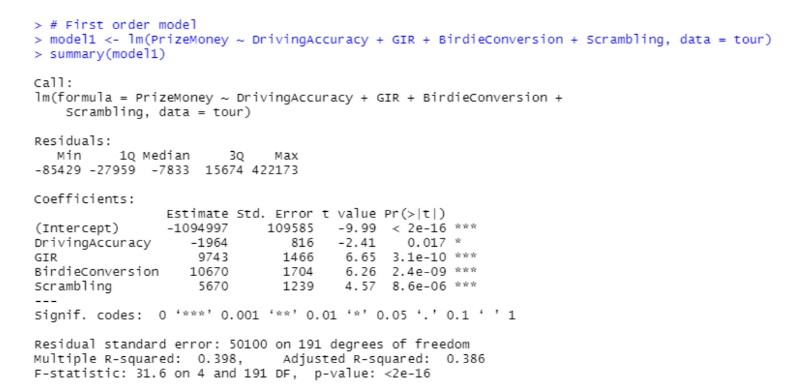
5 cross-validation



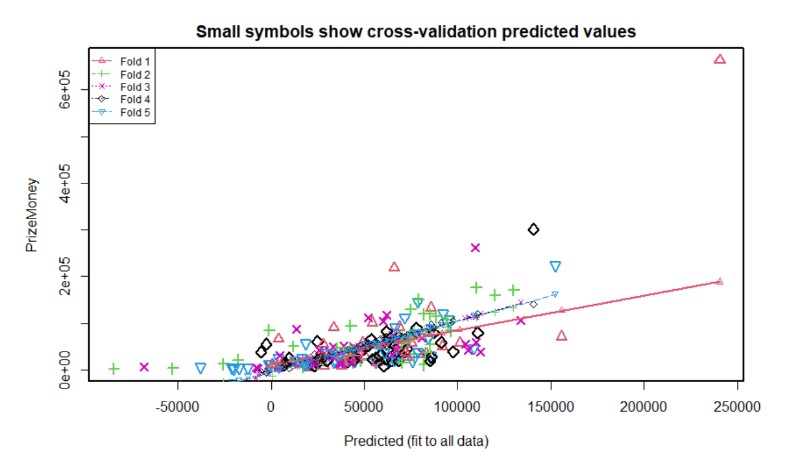
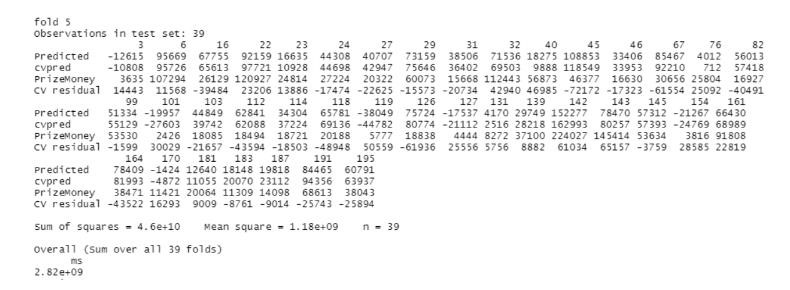
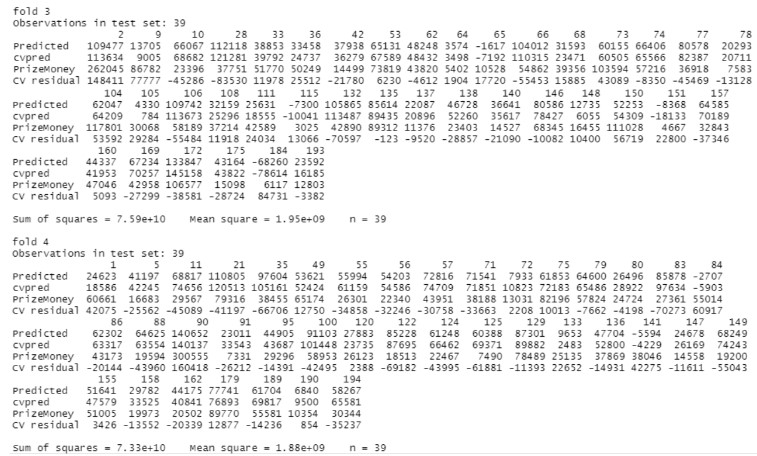
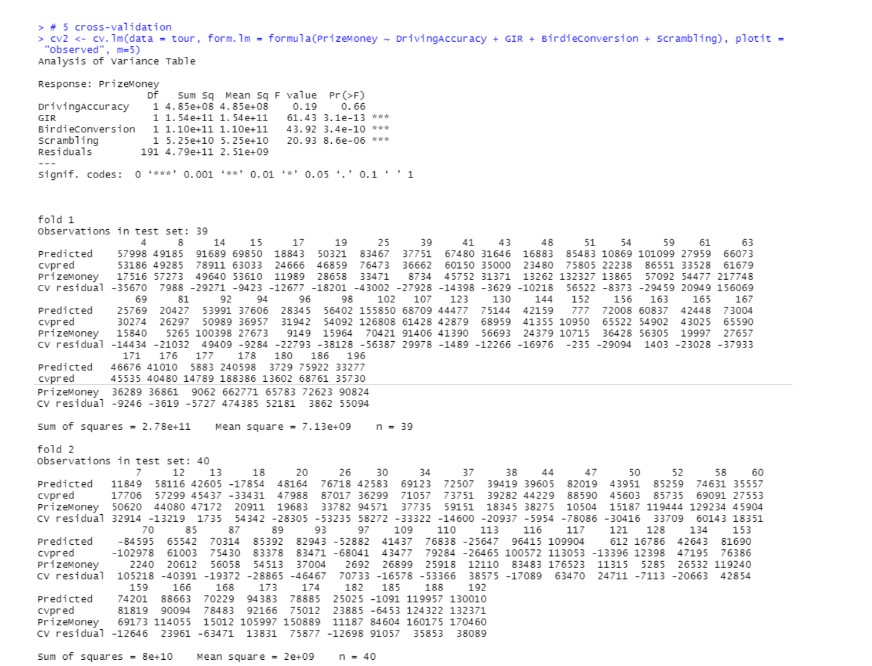




First order model after removing variables AveDrivingDistance, PuttingAverage, SandSaves, BounceBack, PuttsPerRound

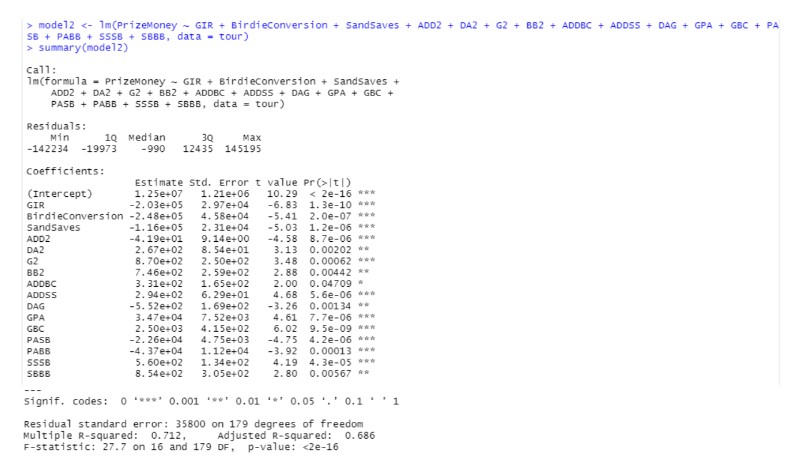


Doing 5 cross-validation



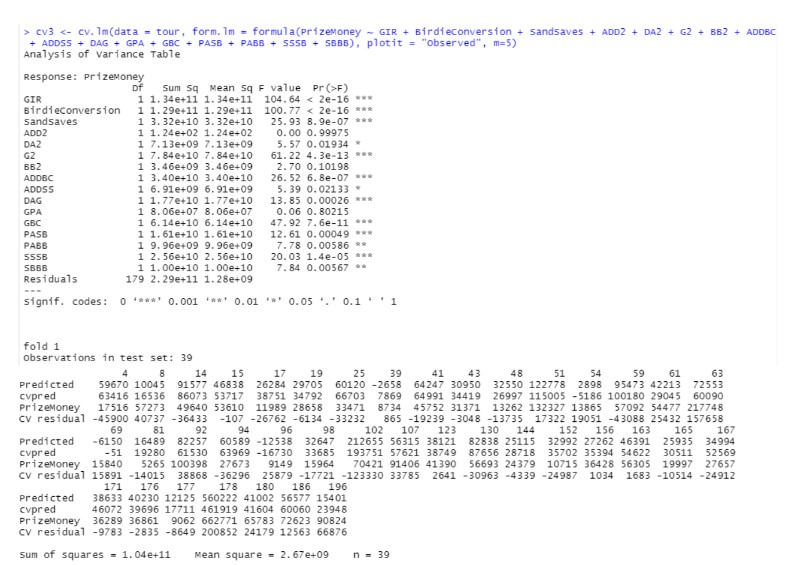
b. (10 points) Evaluate scatterplots to determine which second-order terms should be tested. Test them using 5-fold cross validation and add them one-by-one until you arrive at a model you feel is appropriate. Present the model.

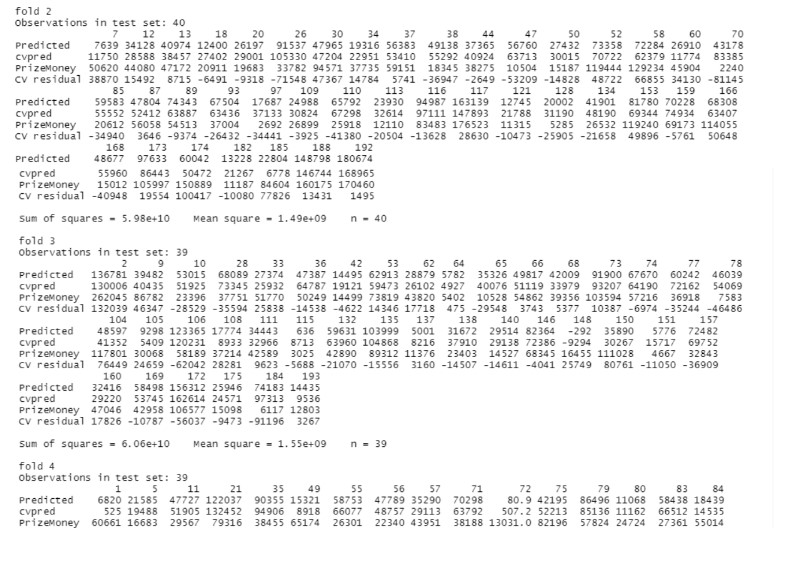
Solution: Final second-order model

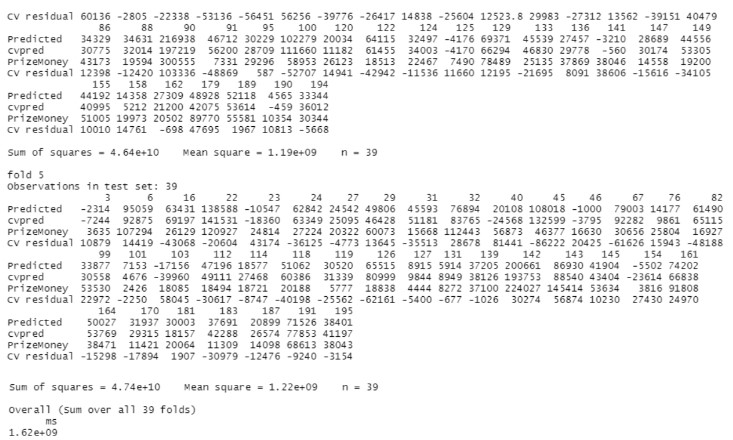


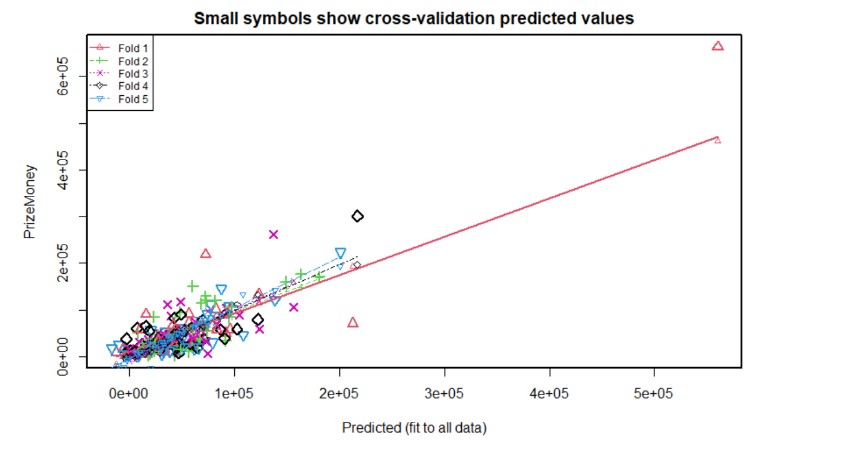
tour$ADD2 <- tour$AveDrivingDistance^2 tour$DA2 <- tour$DrivingAccuracy^2 tour$G2 <- tour$GIR^2 tour$BB2 <- tour$BounceBack^2 tour$ADDBC <- tour$AveDrivingDistance \* tour$BirdieConversion tour$ADDSS <- tour$AveDrivingDistance \* tour$SandSaves tour$DAG <- tour$DrivingAccuracy \* tour$GIR tour$GPA <- tour$GIR \* tour$PuttingAverage tour$GBC <- tour$GIR \* tour$BirdieConversion tour$PASB <- tour$PuttingAverage \* tour$Scrambling tour$PABB <- tour$PuttingAverage \* tour$BounceBack tour$SSSB <- tour$SandSaves \* tour$Scrambling tour$SBBB <- tour$Scrambling \* tour$BounceBack

5 cross-validation









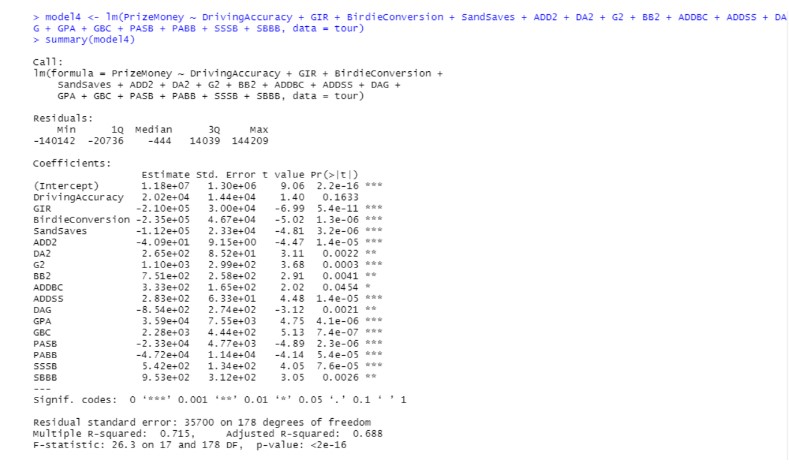
c. (10 points) Beginning from scratch, engineer all possible second-order terms and add them to your dataset. From this dataset, produce a model using backward selection.

Evaluate this model using 5-fold cross validation. Do you arrive at the same model as above? Explain.

Solution:

The model produced using backward selection has one extra variable ‘DrivingAccuracy’.

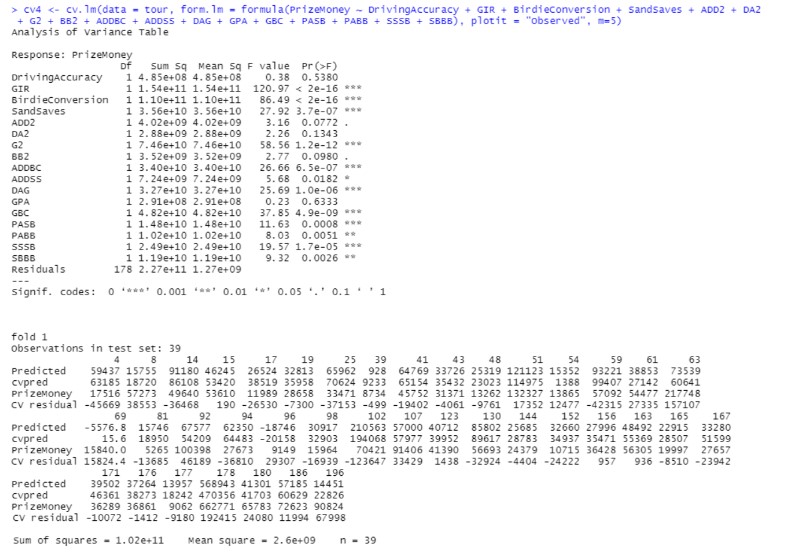
Second-order model by backward selection

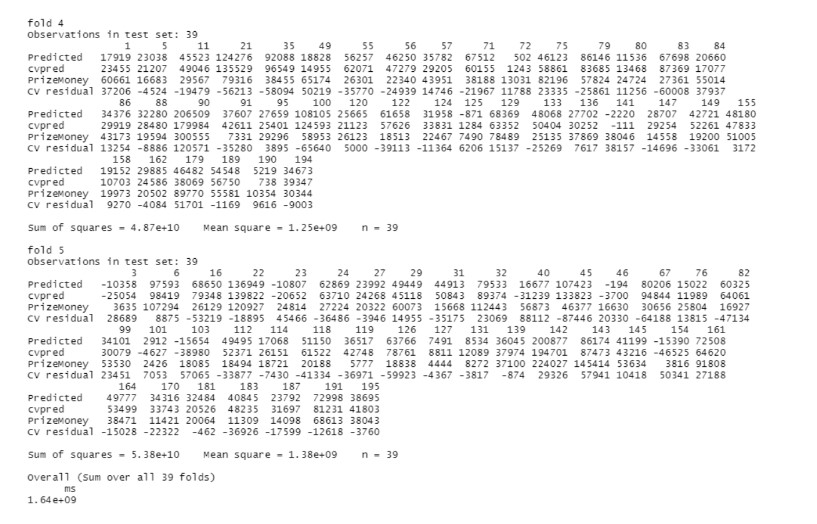
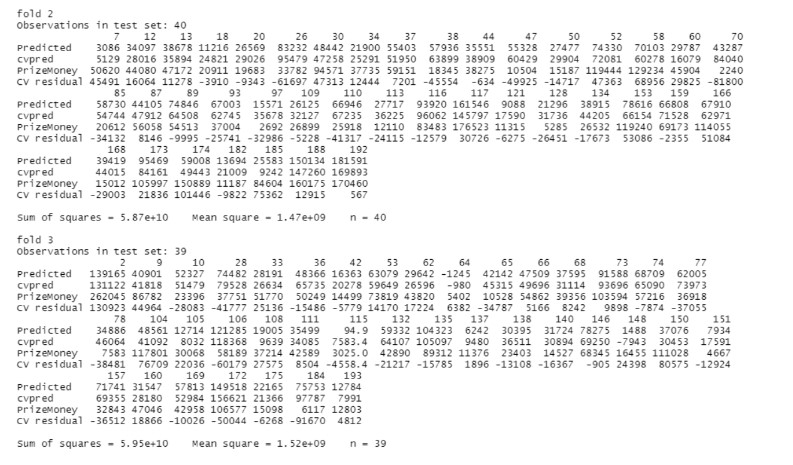


tour$ADD2 <- tour$AveDrivingDistance^2

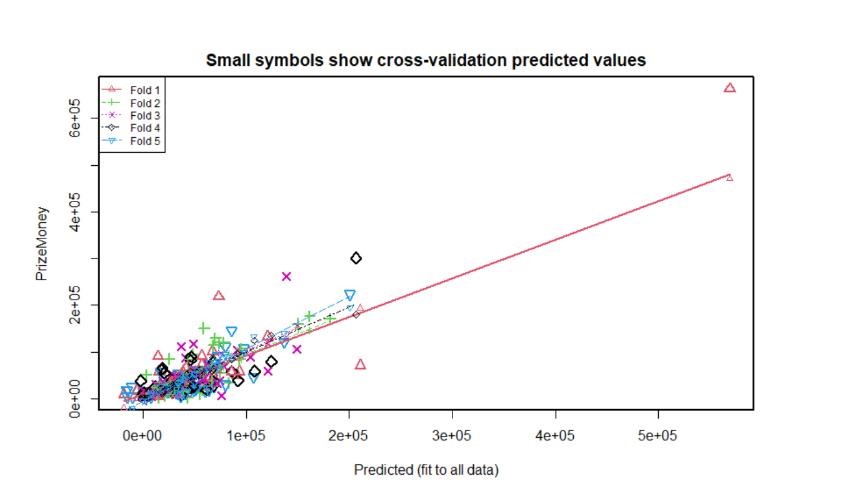
tour$DA2 <- tour$DrivingAccuracy^2 tour$G2 <- tour$GIR^2 tour$BB2 <- tour$BounceBack^2 tour$ADDBC <- tour$AveDrivingDistance \* tour$BirdieConversion tour$ADDSS <- tour$AveDrivingDistance \* tour$SandSaves tour$DAG <- tour$DrivingAccuracy \* tour$GIR tour$GPA <- tour$GIR \* tour$PuttingAverage tour$GBC <- tour$GIR \* tour$BirdieConversion tour$PASB <- tour$PuttingAverage \* tour$Scrambling tour$PABB <- tour$PuttingAverage \* tour$BounceBack tour$SSSB <- tour$SandSaves \* tour$Scrambling tour$SBBB <- tour$Scrambling \* tour$BounceBack

5cross validation





\



d. (10 points) You have used two procedures to build a second-order model. Compare these two procedures. Which do you think is “best”? Explain.

Solution: Although the backward selection technique is quicker than the first, it adds a driving accuracy variable with a p value greater than 0.05 that doesn't improve the model. In my opinion, the backward selection model is inferior to the first model without the additional variable. However, it might lead to overfitting. Backward selection can characterise the variation of the dependent variable.