Kaggle Ames Housing Project Reports

Emily Goren, Andrew Sage, Haozhe Zhang

Report (4) (on April 28, 2017)

Method	CV RMSE	Kaggle Score
PLS ("Non-honest" CV)	0.1248234	0.12290
PLS ("honest" CV)	0.1638773	0.12689
Elastic Net ("Non-honest" CV)	0.1338417	0.13309
Elastic Net ("Honest" CV)	0.174089	0.17501
Random Forest ("Non-honest" CV)	0.1434234	0.14174
Random Forest ("Honest" CV)	0.1310726	0.14389

- This week we spent most of time in developing R function for "honest" cross-validation in "caret", which does imputation for each fold in cross validation rather than the whole dataset.
- We did the "honest" cross-validation for PLS, Elastic Net and Random Forests. For RF, the Kaggle score came out a little worse than the CV score. For PLS, the Kaggle score came out considerably better.
- We updated the feature matrix by incorporating the ordering of categorical variabls in the preprocessing steps.
- We tried some work on 2-step random forest predictions where an initial RF is used to narrow down the choice of predictors and a second RF is then used to make predictions.

Report (3) (on April 21, 2017)

Current Progress:

- Kaggle Scoreboard: 0.11944 (#514) by stacking xgboost and Elastic Net predictions. The current best cross validation error is 0.12037.
- Inspired by Friday's on-class discussion, we re-generated the feature matrix. The major modification includes: 1. imputes most missing values by exploiting correlation other predictors, e.g., if garage year built is missing impute it by house year built; 2. transform almost all the categorical variables into numeric variables as Tanner Carbonati's posting suggests; 3. detect some outliers of SalePrice.
- We changed the loss function into logarithm scale. However, fitting an elastic net model and performing CV on the log outcome scale did not change Kaggle's RMS[log]E (using the old feature matrix).

Future work:

- We will try to perform dimension deduction on the feature matrix by doing PCA. We hope to see improvement on the cross validation error.
- For the new feature matrix, We will tune and cross-validate again the Elastic Net, Lasso, random forest, and xgboost predictions.
- Adding some method's prediction into the feature matrix as a new feature may improve the cross validation error. We will try this approach.
- Assume we have tuned all the methods that we want to use and obtained the training predictions of these methods, denoted as $\hat{\mathbf{y}}_1, \ldots, \hat{\mathbf{y}}_m$. Now, we want to stack these methods linearly. One possible way is to find the optimal coefficients by cross validation. We are wondering whether we could determine

these coefficients by fitting Lasso regression, i.e.,

$$\min\{||\mathbf{y} - \alpha_1\hat{\mathbf{y}}_1 - \ldots, \alpha_m\hat{\mathbf{y}}_m||^2 + \lambda\sum_{i=1}^{m}|\alpha_i|\}$$

.

Report (2) (on April 14, 2017)

Current Progress

Method	Cross-validation RMSE
Elastic Net (tuned to Ridge Regression)	27267.2
Random GLM (order 2)	609239.4
Random Forest	27928.9
Conditional Random Forest	30489.9
PLS	32350.0

Kaggle Scoreboard: 0.12746 (#987) using average of random forest and PLS predictions. We carefully tuned and cross-validated the Elastic Net, PLS, random forest, and conditional random forest predictions individually. We will start to build essemble predictions based on current models.

Report (1) (on April 7, 2017)

Our team "CycloneSTAT" made a primitive attempt of submission to Kaggle this week. The current public score is 0.13037 with a ranking of #1149 (out of 2244 teams). This week we focused on data cleaning, data exploration, handling missing values and feature selection. We will start to do repeated cross validation to select models.

Data Cleaning

Including but not limited to:

- Change MSSubClass to a factor;
- Combine condition1 and condition2 variables into an indicator for each level.
- For numeric variables, replace `NA`'s with the median value;
- For factor variables, make NA's into their own factor level. Most of these are "not applicable"" so they'll likely be correlated with similar variables, e.g. garage area and garage quality;
- Change the two condition variables into binary indicators for each condition, etc.

Feature Selection

We used a feature selection R package called "Boruta" (see Kursa & Rudnicki, J Stat Software, 2010) to identify a total of 56 "important" features for prediction. The importance ranking of these features were also investigated in a linear model, boosted tree model, and random forest (all untuned). Most of the features we considered were the raw variables provided in the training set.It seemed that the models without the unimportant variables did better, but we didn't test this very extensively.

Predictive Methods

We used "caret" package to select tuning parameters for PLS, PCR, RF and tree methods, and used these models to make predictions on the test data. At this point, we just took a straight average of the estimates produced by the 4 techniques. We also fitted an elastic net model using the Boruta selected features and their first order interactions (tuning to include ridge and lasso penalties).