

Introduction
Variables
Missing Data
Modeling
Feature
Importance
Take-Away
Future Work

Travelers Data Science Competition 2019: Customer Retention

May 9, 2019

Aaron Palmer

Introduction: Challenge

Introduction

Variables

Missing Data

Modeling

Feature
Importance

Take-Away

Future Work

Using historical policy data, create a retention model to predict those policies that are most likely to cancel as well as understand what variables are most influential in causing a policy cancellation.

- The data set is based on 4 years of property insurance policies from 2013 - 2017 (in reality the data was simulated)
- There are approximately 250,000 policies that were canceled during the effective term.
- Goal: Build on train, predict on test
- Objective measure: AUC

Variables: Definition and Properties

Variable	Variable Type
id	unique policy identifier
claim	categorical: nominal (2 levels)
gender	categorical: nominal (2 levels)
marital status	categorical: nominal (2 levels)
coverage type	categorical: nominal (3 levels)
sales channel	categorical: nominal (3 levels)
dwelling type	categorical: nominal (4 levels)
house color	categorical: nominal (4 levels)
state*	categorical: nominal (6 levels)
credit	categorical: ordinal (3 levels)
year	positive integer
tenure	positive integer
length at residence	positive integer
number of adults	positive integer
number of children	positive integer
age of policyholder	positive integer
premium	positive real

* After converting zipcode to state

Introduction

Variables

Missing Data

Modeling

Feature
Importance

Take-Away

Future Work

Variables: Initial Concerns

Introduction

Variables

Missing Data

Modeling

Feature
Importance

Take-Away

Future Work

- dwelling type contains a category “Landlord” not present in training data
- training data contains years 2013 - 2016, testing is only 2017
- Errors in target of training data (value is -1 instead of 0 or 1)
- Extreme values e.g. age
- Missing values (98.63% of the data is complete)
- Unbalanced classes (0: 253,097 vs 1: 792,026 in training)
- Distribution similarity between train and test (besides year and dwelling type)

Missing Data: Initial Look

id	0
cancel	0
year	0
zipcode	951
housecolor	945
age	1002
lenatres	967
credit	907
covtype	980
dwelltype	993
premium	957
channel	1025
gender	960
married	994
nadults	931
nkids	938
tenure	980
claim	986

Figure 1: Training Set

id	0
year	0
zipcode	370
housecolor	406
age	391
lenatres	380
credit	380
covtype	405
dwelltype	422
premium	387
channel	394
gender	372
married	431
nadults	405
nkids	398
tenure	401
claim	427

Figure 2: Testing Set

Possibilities

Introduction

Variables

Missing Data

Modeling

Feature
Importance

Take-Away

Future Work

Many machine learning and statistical models require complete data, how do we address missingness?

- Impute on combined test and train
- Impute on on train and test separately
- Delete all cases with missing in training, impute on testing

Preparing for Multiple Imputation

Introduction

Variables

Missing Data

Modeling

Feature
Importance

Take-Away

Future Work

Multiple imputation has certain requirements. As the data is simulated, we cannot be sure about missing mechanisms. However, missingness is present even in the test set. Perhaps missing at random (MAR) is a reasonable assumption.

- Marginal sample data distributions from test set seem similar to train set
- For age ≥ 110 can either impute or hard code. We chose to impute. Need to be cognizant of censoring issues.
- Any missing data must be set to "NA" (this was not the case originally)
- For dwelling type = "Landlord" either set as "NA" and impute, or choose closest other category i.e. "House"
- Concatenate the training and testing set together, dropping the cancel and id variables

Multiple Imputation

Several packages are available for imputation, but we stick with the MICE package in R. After setting each variable's data type, MICE sets up the appropriate imputation model*:

- Continuous variables: predictive mean matching
- Binary variables: logistic regression
- Nominal categorical variables: Bayesian polytomous regression (multinomial regression)
- Ordinal categorical variables: proportional odds model

* All variables were allowed to be predictors for each model (response, id absent) **Challenge:** Due to cluster time and resource limitations and data size, max iterations was estimated and set to 5 and sets imputed set to 3. Sets were imputed in parallel. (Max iterations restricted to 6 before timeout)

Introduction

Variables

Missing Data

Modeling

Feature
Importance

Take-Away

Future Work

Modeling: Multiple Data sets

Introduction
Variables
Missing Data
Modeling
Feature
Importance
Take-Away
Future Work

With $m = 3$ imputed training and testing data sets, we are left with a big question on how to proceed.

Different Choices

- Model Averaging: multiples models on *single* data sets and averaged
- Multiple Model Averaging: multiple models on *multiple* data sets and averaged
- Model Stacking & Multiple Model Stacking: predict on predictions – very important to make sure out-of-sample data is preserved

Models:Neural Networks

Introduction
Variables
Missing Data
Modeling
Feature Importance
Take-Away
Future Work

- Black-box powerful function approximators capable of capturing complex relationships
- Tuning these models can be tricky
- Prone to over-fitting
- Capable of modeling large datasets by using mini-batches for stochastic gradient descent

Best architecture contained 2 hidden layers, the first with 30 nodes, and the second with 5. Regularization was used, but didn't affect the model much. Score improvements were very slow.

Models: XGBoost

Introduction

Variables

Missing Data

Modeling

Feature
Importance

Take-Away

Future Work

- XGBoost is a highly scalable ensemble learning method
- Handles large data well
- Often the 'go to' for many data science competitions
- Feature importance available
- Randomized parameter search even using 2 fold cross validation was very expensive
- Models were trained on each imputed data set

Models: Aggregation

Introduction

Variables

Missing Data

Modeling

Feature
Importance

Take-Away

Future Work

- By using multiple models it becomes harder to track feature importance. Since XGBoost was the main model used, each data set offered its own features.
- After each model was trained, predicting on its own test set, the predicted probabilities were averaged, i.e.

$$\hat{y}_i = \frac{1}{3} \sum_{j=1}^3 \hat{y}_{j,i} \text{ where } \hat{y}_{j,i} \text{ is the } j^{th} \text{ model for subject } i.$$

Feature Importance: XGBoost

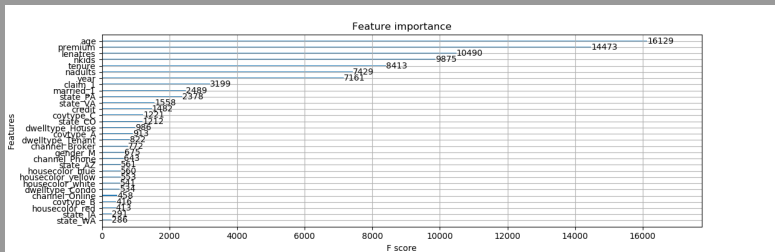
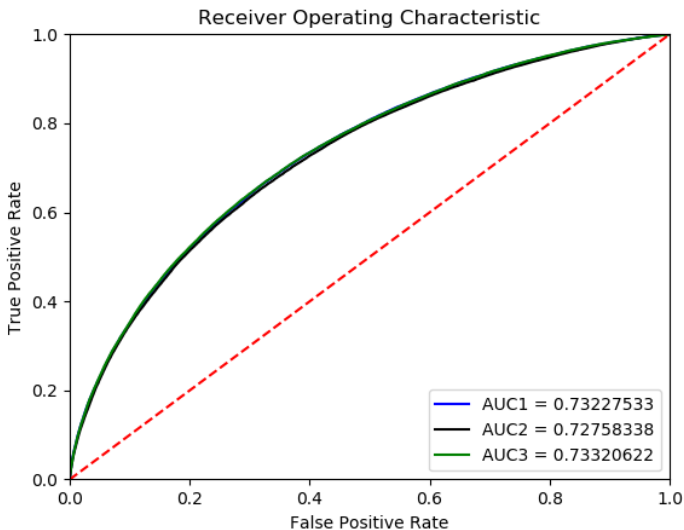


Figure 3: Representative feature importance for each imputed data set

Question:

How do the results inform model criticism? With variables 'nkids' and 'nadults' high up, perhaps feature engineering may help creating a new variable $kar = \frac{nkids}{nadults}$. This didn't help much.

Receiver Operating Characteristic curve



Current Leaderboard























#	Team Name	Kernel	Team Members	Score 📊	Entries	Last
1	West's Checkmate		   	0.72941	7	4d
📍	benchmark.csv			0.72932		
2	Rule of Three		  	0.72906	17	1h
3	Never overfit		  	0.72826	15	5d
4	Xplorers			0.72825	26	4d
5	Yo			0.72779	3	4d
6	TheDataWhisperers			0.72757	1	20h
7	Prof. Chaos			0.72752	3	4d
8	KKKartMan			0.72734	4	10h
9	HHL		  	0.72552	15	4d
10	Omni		 	0.55749	2	6h
11	Normal Curves, Fat Tails		 	0.50011	3	1h

Figure 5: Public leader board as of 12pm, competition closes tonight

Take-Away: The Problem

Introduction

Variables

Missing Data

Modeling

Feature
Importance

Take-Away

Future Work

- Recall the task was measured with AUC.
- Low training error did not necessarily correspond to low high due to class imbalance.
- Modifying the models to take this into account increased error.
- This task was largely a tuning problem, but brought to light important issues when attempting to merge statistical methodologies with machine learning models.

Future work

Introduction

Variables

Missing Data

Modeling

Feature
Importance

Take-Away

Future Work

- Allow multiple imputation to run longer
- Can we handle some of the data differently, i.e. how does encoding 'Landlord' to 'House' affect the model.
- Can temporal information be better utilized?
- Find better ways to combine models

Reference

Introduction

Variables

Missing Data

Modeling

Feature
Importance

Take-Away

Future Work

- ① Buuren, S. van, and Karin Groothuis-Oudshoorn. "mice: Multivariate imputation by chained equations in R." Journal of statistical software (2010): 1-68.
- ② Chen, Tianqi, and Carlos Guestrin. "Xgboost: A scalable tree boosting system." Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. ACM, 2016.
- ③ Murphy, Kevin P. Machine learning: a probabilistic perspective. MIT press, 2012.