Name: Shashishekhar Godbole

Location: Mumbai, India

Competition: Practice Fusion Diabetes Classification Competition

1. Summary

SyncPatient table was used as the base dataset and other datasets were transformed to one-row-per-patient level and then merged with the base dataset. Features were generated from fields in tables SyncPatient, SyncTranscript, SyncDiagnosis, SyncLabResult, SyncMedication, SyncLabPanel and SyncLabObservation. No special technique was used for feature selection. Training techniques used for training individual models were: gradient boosting (package GBM in R), random forests (package randomForest in R) and neural networks (package nnet in R). Individual models were ensemble using a neural network (package nnet in R). Wikipedia (http://en.wikipedia.org/wiki/List_of_ICD-9_codes) was used to group similar diagnoses together. No other external source of data / information was used.

2. Features selection / extraction

Features were generated to capture as much meaningful health related information as possible from various tables. The embedded excel spreadsheet gives a list of all features generated from the data and their usage in each individual model:

3. Modeling techniques and training

The final model was an ensemble of 10 individual models, 8 of which were random forests and the remaining 2 were GBM and neural network. The table below shows the details of each of the models

Model name	Specifications	CV	CV error
GBM	N = 2500, S = 0.01, I = 5, BF = 0.75, TF = 0.95, OBS = 50	Folds = 10, Repeats = 2	0.3514
NN	S = 3, R = 0.0, D = 0.1, N = 500	Folds = 10, Repeats = 20	0.3562
RF	N = 2500, S = 20	Folds = 25, Repeats = 2	0.3665
RF2	N = 2500, S = 20	Folds = 25, Repeats = 2	0.3600
RF3	N = 2000, S = 100	Folds = 25, Repeats = 2	0.4147
RF4	N = 2500, S = 20, M = 20	Folds = 25, Repeats = 2	0.3614
RF5	N = 4000, S = 20, M = 10	Folds = 25, Repeats = 4	0.3596
RF6	N = 3000, S = 20, M = 6	Folds = 25, Repeats = 2	0.3594
RF7	N = 2500, S = 60	Folds = 25, Repeats = 2	0.3880
RF8	N = 3000, S = 20, M = 25	Folds = 25, Repeats = 2	0.3628
RF9	N = 4000, S = 20, M = 10	Folds = 25, Repeats = 2	0.3595
RF10	N = 500, S = 20, M = 10	Folds = 25, Repeats = 4	0.3543
Final Ensemble (NN)	**S = 7, R = 0.0, D = 0.1, N = 500**	**Folds = 25, Repeats = 1**	**0.3301**

For GBM, N = number of trees, S = learning rate, I = interaction depth, BF = bag fraction, TF = train fraction, OBS = minimum observations in a node

For RF, N = number of trees, S = minimum observations in a node, M = mtry

For NN, S = size, R = rang, D = decay, N = number of iterations

4. Code description

The structure of the code for each of the 12 individual models is very similar. The code for each model consists of the following R scripts:

a. Master_Code_XX.R (where XX = GBM or NN or RF): This is the main code that should be run in order to execute each model. It has the other three codes embedded in it. This code imports the data, creates features, creates CV samples, runs the models and scores the test dataset.

b. Data_import.R: This code imports all the required data files from trainingSet and testSet folders. For convenience, the prefixes “training_” and “test_” were removed from the raw data files to be imported.

c. Data_process.R: This code creates most of the features required by the model. It takes SyncPatient as the base dataset and merges new features with it.

d. XX_model.R (where XX = GBM or NN or RF): This code contains the actual command used to build the model and score it on validation.

The code for ensembling merges cross-validation and test datasets for each of the 12 models into one datasets. Then it builds a neural network model on the cross-validation data and scores it on the test data.

5. How to generate the solution

The code Master_Code_XX.R needs to be run first for each of the 12 models. Then, the code “Ens_2012_09_10_V1.R” from the folder “2012-09-10-V1-Ens” needs to be run to generate the solution. The name of the final solution file is “FinalSubmit.csv”

6. Additional comments and observations

Ensembling of various random forest models was found to improve the overall performance. However, ensembling of various GBM and NN models did not improve the overall performance.

CV score was found to be the better measure of private leaderboard score rather than the public leaderboard score.

7. Figures

No charts / figures were created during the analysis.

8. References

No specific external references. The solution was designed based on the knowledge gained from previous kaggle competitions and ideas shared by other kagglers.