Final Capstone Final Report

Zhikang Jiang (URMC-oncology)

Introduction

Polypharmacy refers to the concurrent use of multiple medications, which is extremely common in older adults, and it is associated with medications that are "potentially inappropriate." In other words, such polypharmacy problem increases the death rate of patients, particularly with geriatric oncology. In this program, my teammates and I are mainly focusing on conducting the research and analysis of polypharmacy problem of patients with geriatric oncology.

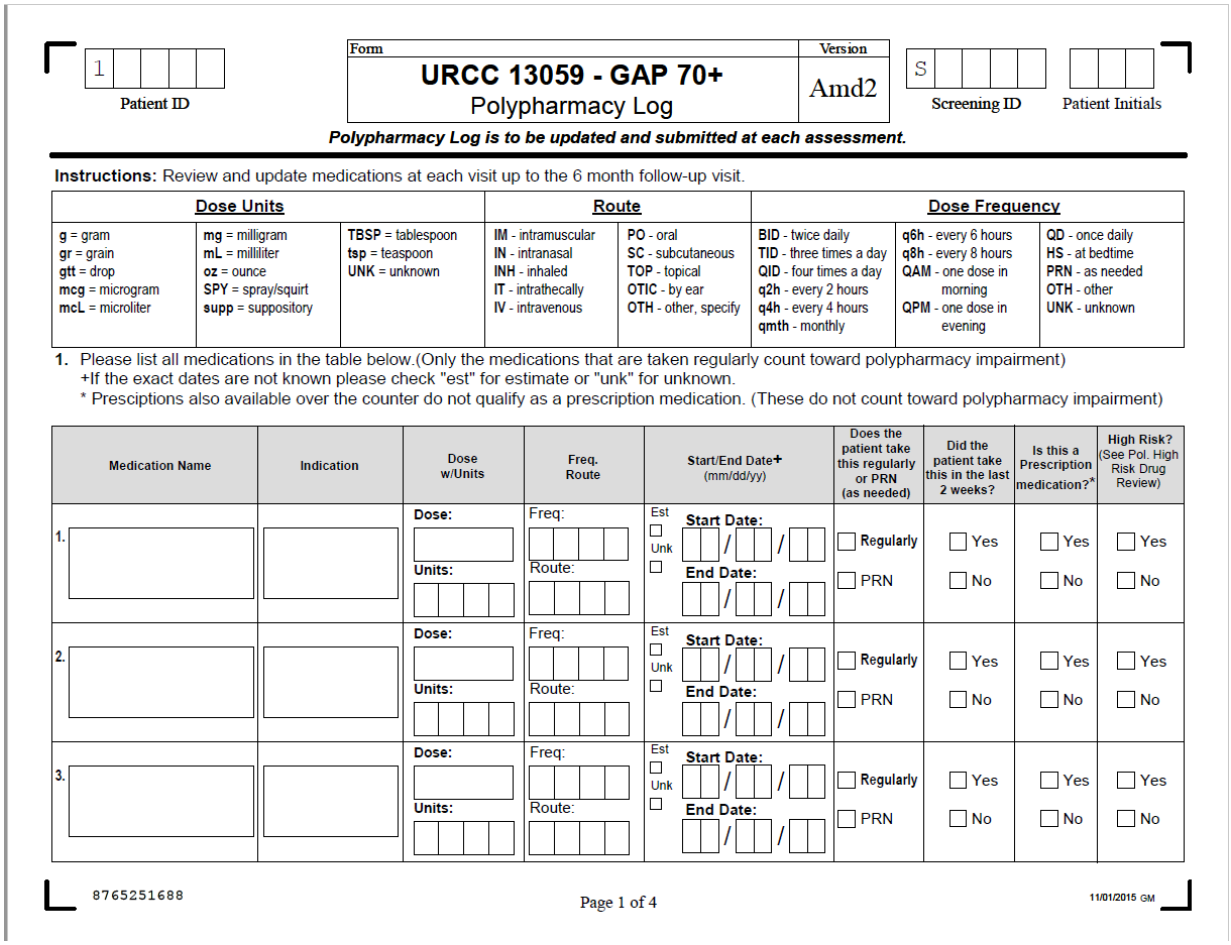
After the very beginning and productive meeting with the URMC team, we have decided to accomplish two objectives in this project. Objective One is to conduct data cleaning on medication lists. Due to the fact that there are many incorrect spelling and wrong trade names, we have to design an algorithm, which converts trade names to generic names. And we need to correct the misspelled trade names in the medication list based on the medication dictionary provided by the URMC team. After that, we have to split and categorize the combination medications in the dataset. In the end, all the data should be grouped by the patients' ID for further research.

In the Objective Two, we have formed a hypothesis on falling problem among aging cancer patients and after that we successfully built a predictive model, which helps clinicians to predict how likely a patient is going to fall.

Data Collection

The URMC team has collected data by encouraging patients with geriatric oncology in this program to fill several specific types of forms for future study. Then all the form

written by handwriting would be scanned into the dataset. Here comes the problem in the Data Collection part, due to the fact that sometimes handwritings are unrecognizable in computer, there are some misspelled names in the dataset. The figure below is the example of form.



1 Patient ID

Form: **URCC 13059 - GAP 70+** Polypharmacy Log — Version: Amd2

S Screening ID — Patient Initials

Polypharmacy Log is to be updated and submitted at each assessment.

Instructions: Review and update medications at each visit up to the 6 month follow-up visit.

Dose Units			Route		Dose Frequency		
g = gram **gr** = grain **gtt** = drop **mcg** = microgram **mcL** = microliter	**mg** = milligram **mL** = milliliter **oz** = ounce **SPY** = spray/squirt **supp** = suppository	**TBSP** = tablespoon **tsp** = teaspoon **UNK** = unknown	**IM** - intramuscular **IN** - intranasal **INH** - inhaled **IT** - intrathecally **IV** - intravenous	**PO** - oral **SC** - subcutaneous **TOP** - topical **OTIC** - by ear **OTH** - other, specify	**BID** - twice daily **TID** - three times a day **QID** - four times a day **q2h** - every 2 hours **q4h** - every 4 hours **qmth** - monthly	**q6h** - every 6 hours **q8h** - every 8 hours **QAM** - one dose in morning **QPM** - one dose in evening	**QD** - once daily **HS** - at bedtime **PRN** - as needed **OTH** - other **UNK** - unknown

1. Please list all medications in the table below.(Only the medications that are taken regularly count toward polypharmacy impairment)
   +If the exact dates are not known please check "est" for estimate or "unk" for unknown.
   * Presciptions also available over the counter do not qualify as a prescription medication. (These do not count toward polypharmacy impairment)

Medication Name	Indication	Dose w/Units	Freq. Route	Start/End Date+ (mm/dd/yy)	Does the patient take this regularly or PRN (as needed)	Did the patient take this in the last 2 weeks?	Is this a Prescription medication?*	High Risk? (See Pol. High Risk Drug Review)
1.		Dose: Units:	Freq: Route:	Est Unk Start Date: End Date:	Regularly PRN	Yes No	Yes No	Yes No
2.		Dose: Units:	Freq: Route:	Est Unk Start Date: End Date:	Regularly PRN	Yes No	Yes No	Yes No
3.		Dose: Units:	Freq: Route:	Est Unk Start Date: End Date:	Regularly PRN	Yes No	Yes No	Yes No

8765251688 — Page 1 of 4 — 11/01/2015 GM

Figure 1. Form for collecting polypharmacy log of patients[2]

Furthermore, the URMC team has already collected data of 691 patients and 7426 medication names and form the cleaned dataset for 60 patients. All the drug names are formulated into 3 columns Excel spreadsheet with trade name, generic name and second drug.

Data visualization

Until now, the URMC team has provided us with a dataset of 60 patients with geriatric oncology, which has already been cleaned as test dataset for our designed algorithm.

There are 28 columns and 480 rows in the cleaned dataset. Also, they have also sent us three datasets worked as dictionaries for identifying the misspelled names and for correspondence between trade names and generic names. In addition, we are provided with a larger dataset with only two attributes: patients' ID and trade names of medications.

Here we group the 60-patient cleaned dataset by patients' ID and the graph below illustrates the dataset.

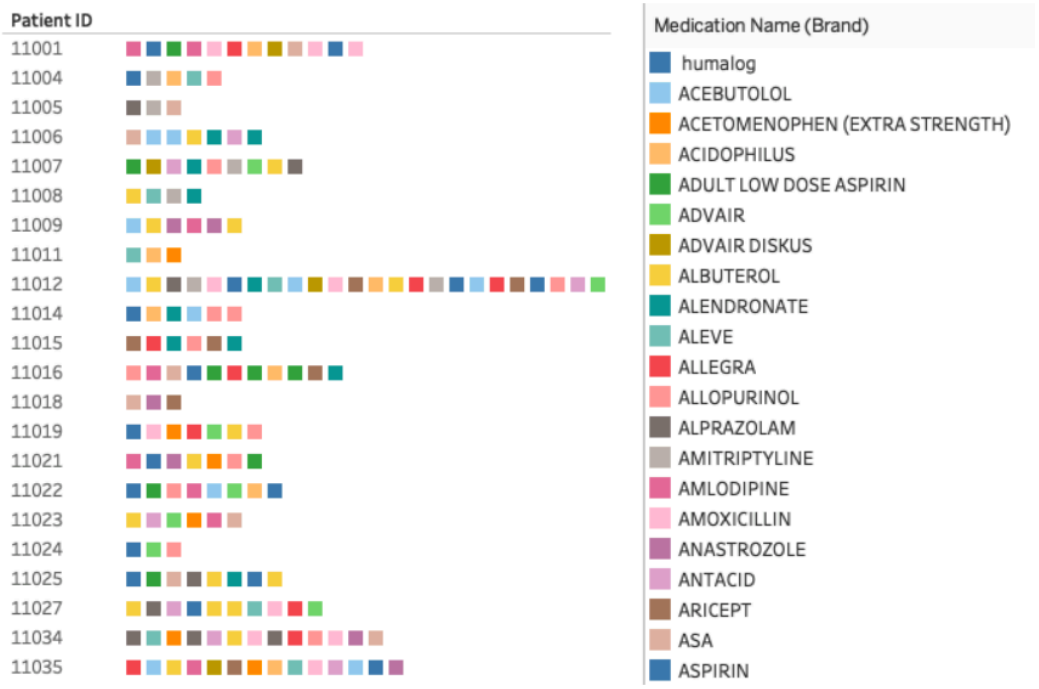


Figure 2. Part of illustration of 60-patients dataset grouped by patient ID

The graph below is the dictionary (master list), which shows the correspondence between drug name and generic name. There would me multiple generic name match-ups for one trade name. Also, generic names may contain multiple words, which need to be split in the data cleaning process.

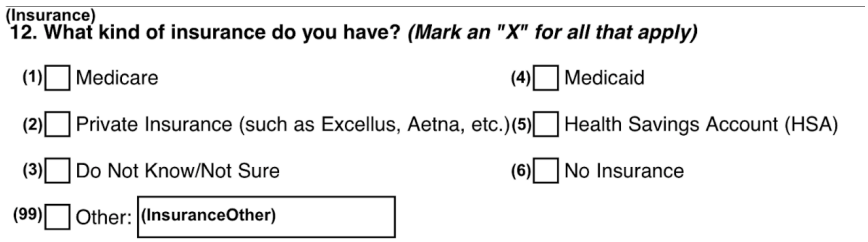
DRUG_NAME	GENERIC_NAME
1ST TIER UNIFINE PENTIPS	NEEDLES, INSULIN DISPO..
8-MOP	METHOXSALEN
A-HYDROCORT	HYDROCORTISONE SOD S..
A-METHAPRED	METHYLPREDNISOLONE S..
ABACAVIR	ABACAVIR SULFATE
ABACAVIR-LAMIVUDINE-Z..	ABACAVIR/LAMIVUDINE/..
ABELCET	AMPHOTERICIN B LIPID C..
ABILIFY	ARIPIPRAZOLE

Figure 3. Partof corresponding relationdictionary

Also, there are 77 features in the dataset. We have selected 35 features, which has less than 15 missing values to further perform. Among these 35 features, only 5 features are non-numeric features.

Data Cleaning

Data cleaning is essential for our dataset because there are so many missing values in the dataset. For feature in numeric, we filled up all the missing values by groups' median. For categorical features, we filled up all the missing values by groups' mode and we transferred all non-numeric data to categorical data to further clean. Take one of the non-numeric features- Insurance as an example; just as graph below shows, we firstly distribute the data by frequency. Then we choose the five most frequent choices and assign them with numbers from one to five so that we transferred all those non-numeric data to categorical data.


(Insurance)
12. What kind of insurance do you have? *(Mark an "X" for all that apply)*

(1) ☐ Medicare
(4) ☐ Medicaid
(2) ☐ Private Insurance (such as Excellus, Aetna, etc.)
(5) ☐ Health Savings Account (HSA)
(3) ☐ Do Not Know/Not Sure
(6) ☐ No Insurance
(99) ☐ Other: (InsuranceOther)

Figure 4. Illustration of Insurance feature

Data Preparation

After meeting with URMC team, we began to write the algorithm to reform our dataset. The goal is to correct the misspelled trade names based on the given dictionaries and changes the brand names of the medication into generic names. And if one patient is taking multiple drugs, we have to separate those drugs into multiple rows for future study.

Furthermore, we used a Python package – difflib to coreect the misspelling words. This package is used to match the misspelled names with dictionaries based on difference level (cutoff). In this program, the cutoff is set to 0.9, which means the misspelled name should have at least 90% similarity with its match word. The graphs below show the result. Unknown values have to be further manipulated in the future.

	Algorithm correction	Directed matching	Unknow value
0	390	3597	3680

Figure 5. Generic Name Matching Result

Meanwhile, we match the trade names with the correct trade names and correct generic names in the dictionaries because in the data collection process, there are some errors that some generic names are in the drug name column.

	Total Correction Number	Directed matching	Algorithm correction	Unknow value	Warnings
Times	5980	1785	208	1687	10

Figure 6. Trade Name Matching Result

In the Figure4, warning assigns the misspelling words, which has the first letter different from the matched word, because we find that it has a high possibility of mismatching and manually reviewing would be recommended.

In the end, we change trade names to one or more generic names and export the data to csv. File.

Data Preprocessing – Feature Selection

Due to the fact there are so many features in the dataset, we have to select those features, which contribute the most to our prediction. Otherwise having irrelevant features in our model can decrease the accuracy of prediction. We used two feature selection techniques in our modeling: RFECV and Tree based feature selection.

RFECV stands for recursively feature selection using cross validation and it is a technique which not only ranks feature importance, but also provides how many features we need for best accuracy or best AUC area. The graph below shows the result.

```
Optimal number of features : 16
Best features : Index(['FH2', 'Age', 'Gender', 'KPS', 'ImpairedMiniCog', 'ImpairedWeight',
       'ImpairedBMI', 'ImpairedMNA', 'ImpairedTUG', 'ImpairedSPPB',
       'ImpairedADL', 'ImpairedCom', 'ImpairedGDS', 'cancertype',
       'treatment_type', 'FH3'],
      dtype='object')
```

Figure 7. Result of RFECV

Meanwhile, we perform the Tree-based Feature Selection, which is a feature elimination algorithm that uses random classifier to compute the feature importance.

Exploratory Analysis

In advance, accuracy test is conducted on the 60-patients dataset using the same algorithm and cutoff rate. The testing set has 496 rows in total and 272 rows were matched (Gen_Found) in our algorithm.

	testing_misspelled_names	Correction	new_generic_list	ans	new_mark
0	AMLODIPINE	AMLODIPINE	[AMLODIPINE]	[AMLODIPINE]	Gen_Found
1	ASPIRIN	ASPIRIN	[ASPIRIN]	[ASPIRIN]	Gen_Found
2	FINASTERIDE	FINASTERIDE	[FINASTERIDE]	[FINASTERIDE]	Gen_Found
3	GABAPENTIN	GABAPENTIN	[GABAPENTIN]	[GABAPENTIN]	Gen_Found
4	GLIMEPIRIDE	GLIMEPIRIDE	[GLIMEPIRIDE]	[GLIMEPIRIDE]	Gen_Found
5	LISINOPRIL	LISINOPRIL	[LISINOPRIL]	[LISINOPRIL]	Gen_Found
6	METAPROLOL	METOPROLOL	[METOPROLOL]	[METOPROLOL]	Gen_Changed

Figure 8. Test Result

However we found that many of the generic names are full name instead of the partial name. Thus, the real accuracy rate should be little higher because such matching results should be covered.

Also, some of the unknown matching was reviewed manually and we found trade names were directed input to answer column as the generic name in the answer list.

49	ACEBUTOLOL	ACEBUTOLOL	[]	[ACEBUTOLOL]	Unknown
54	DULOXETINE	DULOXETINE	[]	[DULOXETINE]	Unknown
57	HYDRALAZINE	HYDRALAZINE	[]	[HYDRALAZINE]	Unknown

Figure 9. Unknown Generic name review

After all the medicating our algorithm, we calculated the accuracy rate when we apply it on 60-patient dataset. The final accuracy rate is 88.71%.

Initial Modeling

For Objective two, we have discussed with URMC team and came up with a hypothesis on falling problem among patients with geriatric oncology. Can we use cognitive performance, cancer type, biomarker, social support and demographic information to predict how likely a patient is going to fall?

Based on that, we have done a productive literature reviews on the falling problem.

● *Falls are one of the leading causes of injuries in the aging people; 30% aging people over 65 are victim of falls (Todd and Skelton, 2004); 10% of falls lead to serious damage.*

(Goldacre et al., 2002)

● *Well studied predictors of falls include: age, dementia, physical performance, frailty, visual and musculoskeletal disorder, and medications such as psychotropic drugs. (Lastrucci et al., 2017)*

● *Measurements include: Frontal Assessment Battery (Kataoka, 2015), Short Physical Performance Battery (Pua, 2018), Fried Frailty Index Components (Sharma et al., 2019), etc.*

However, to the best of our knowledge, no classification model of falling has been published up to the data using multiple predictors.

Hopefully we could find out the answer of how likely a patient is going to fall by the end of this project.

First Model

For our first attempt, we decided to perform random forest classifier on our dataset and the prediction result below seems like successful when you noticed the accuracy is pretty high.

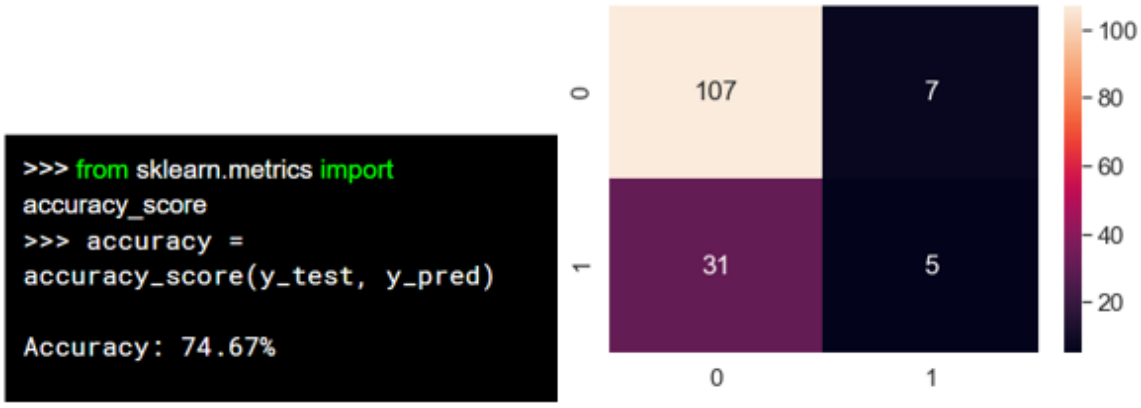


Figure 10. Accuracy and confusion matrix of random forest on raw data

However, the confusion matrix brought us to the reality that almost every data point in the test dataset has been classified to class 0, which predicts not falling. And the true positives were extremely poor. Therefore we decided to have a review on our datasets.

The graph below shows the class distribution of our datasets and it is the first time that we realized the class imbalance played an important role on our data analysis.

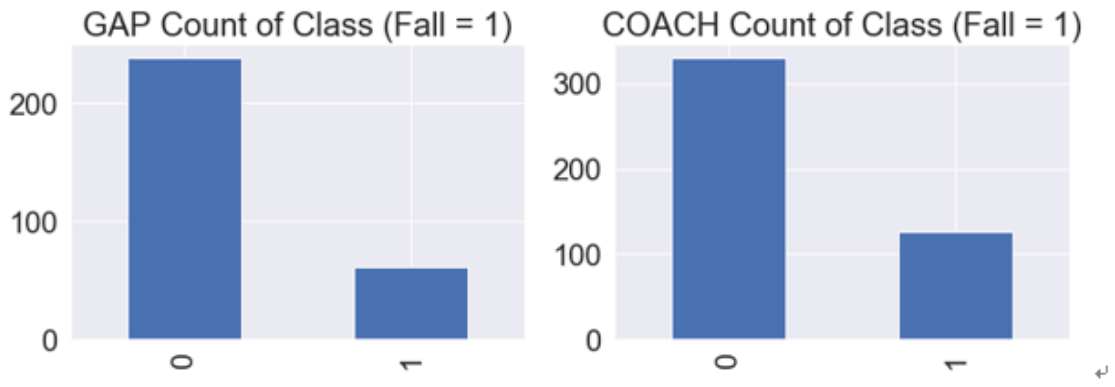


Figure 11. Class distribution of datasets

Second Model

As we found out the class imbalance in our datasets, we decided to resample our data by performing SMOTE to over-sample the minority class and then performing TomekLink to under-sample the majority class for both training and testing datasets. After resampling, we performed the random forest classifier again and gained the prediction result below.

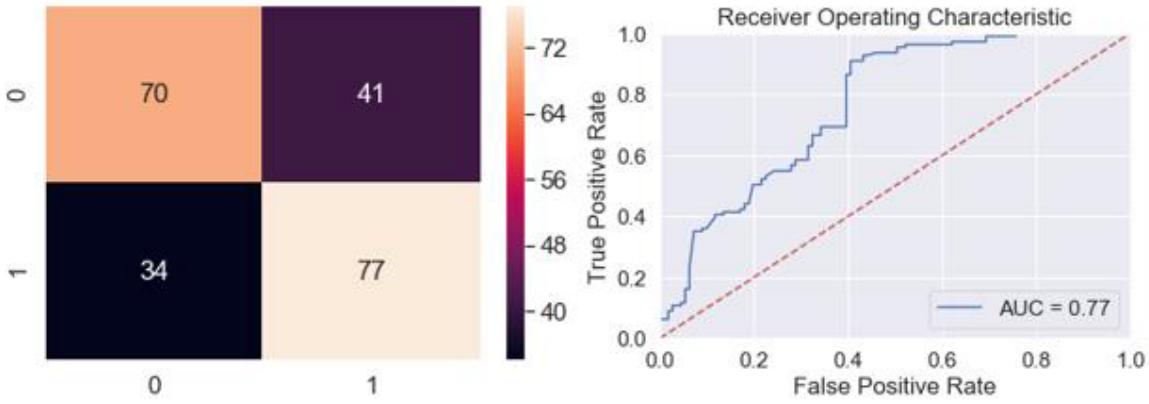


Figure 12. Confusion matrix and ROC curve of random forest classifier

It seems like we have achieved the acceptable results now. But after discussing with Prof. Ajay, we realized that we should only perform resampling techniques on our training dataset but not testing set. However, if we only apply resampling techniques on training

set, the prediction result as shown below is extremely poor and impossible for clinical use.

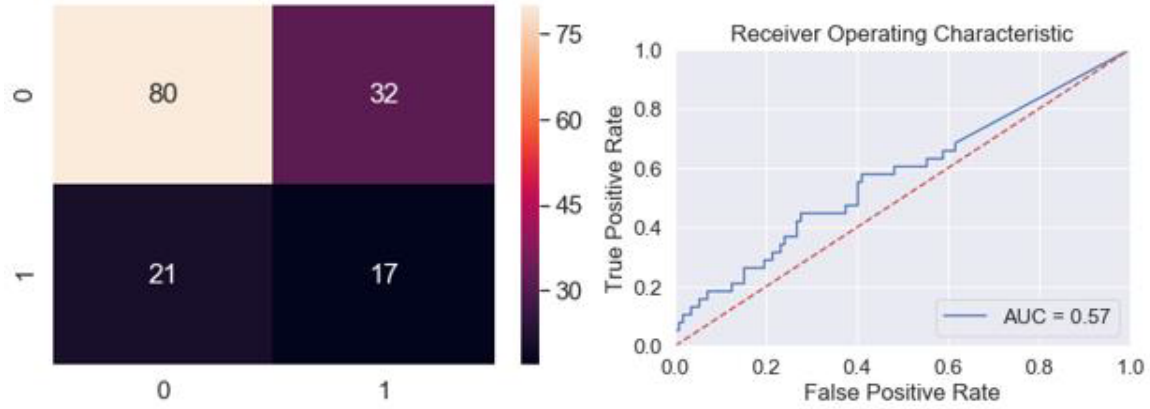


Figure 13.Confusion matrixandROC curve of resampling on train set

Third Model

Though we have failed our first two attempts, we then discussed with Prof. Ajay and moved our attention from accuracy to recall rate of our model. So why recall rate is that important? From clinical perspectives, people value true-positives and false-negatives rather than false-positives and true-negatives. In other words, it is more important for our model to predict a patient falling and the patient do fall in the real case. Because doctors value the high precision of predicting the patients who are going to fall rather than the patients who are not going to fall. Also, we have performed instance harness threshold to under-sample the training dataset and then conducted Logistic Regression using cross-validation to run multiple times for reducing the randomness.

	precision	recall	f1-score	support
0	0.87	0.43	0.57	112
1	0.33	0.82	0.47	39
micro avg	0.53	0.53	0.53	151
macro avg	0.60	0.62	0.52	151
weighted avg	0.73	0.53	0.55	151

Figure 14. Result of Logistic Regression using cross validation

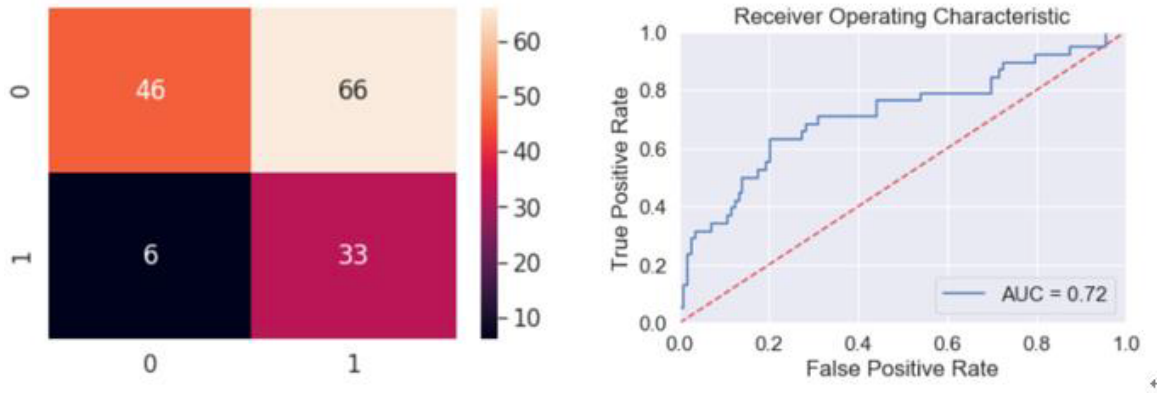


Figure 15.Confusion matrix and ROC curve of Logistic Regression using cross validation

The result above shows an acceptable result based on decent recall rate. Then, Sixu has come up with a great idea after the negotiation with Erika that we might make the prediction more readable for physicians and clinicians. We came up with the idea of returning the likelihood of each patient falling, rather than directly predicting their classes. Then we used predict_proba(X) parameter from logistic regression package to return the probability of falling. The graph below shows the function.

cancertype	Health	Grade	Age	FH2	Live	KPS	Living	Driving	Feel	Marital	ImpairedMNA	ImpairedMiniCog	ImpairedIADL	ImpairedGDS	ImpairedCom	Probability
2	2.0	7.0	1	1.0	1.0	3	1.0	1.0	2	4.0	0	0	0	0	1	0.242482
2	5.0	1.0	0	3.0	2.0	2	2.0	0.0	1	5.0	1	0	1	1	1	0.932660
2	3.0	7.0	0	2.0	1.0	3	2.0	1.0	2	2.0	0	0	1	0	0	0.559271
6	4.0	2.0	0	1.0	1.0	2	2.0	1.0	2	6.0	1	0	1	0	0	0.448995
1	5.0	5.0	1	2.0	1.0	3	1.0	1.0	2	2.0	1	1	1	1	0	0.823329

Figure 15. predict_proba(X) results

Conclusion

In the end, with the 0.82 recall rate and 0.72 AUC, we believe this capstone project is successful and the model we build can be further developed and used by clinicians to predict how likely an aging cancer patient is going to fall. I really appreciate the time I spent with my team and we four worked as a group along the project. I have contributed a lotto Objective one, which is to build an algorithm to transform trade names to generic names and prepared for the final presentation. Thanks to the entire URMC team and Professor Ajay for the great support they granted. Also thanks to all of my teammates for helping me through the project. They are all the best.

Reference:

1. URMC-oncology presentation slides on Blackboard. URL: https://learn.rochester.edu/webapps/blackboard/execute/content/file?cmd=view&content_id=_1768437_1&course_id=_38661_1&frameSetWrapped=true (Accessed 3/8/2019)

2. URCC Revised Final Appendices. Accessed through Box Drive of University of Rochester (Accessed 3/8/2019)

3. Python Package Difflib. URL: https://docs.python.org/2/library/difflib.html

4. Skelton, D. A., et al. "Prevention of Falls Network Europe: a Thematic Network Aimed at Introducing Good Practice in Effective Falls Prevention across Europe." *European Journal of Ageing*, vol. 1, no. 1, 2004, pp. 89–94.

5. Tinetti, Mary E. "Factors Associated with Serious Injury During Falls by Ambulatory Nursing Home Residents." Journal of the American Geriatrics Society, vol. 35, no. 7, 1987, pp. 644–648.

6. Lastrucci, Vieri, et al. "Identification of Fall Predictors in the Active Elderly Population from the Routine Medical Records of General Practitioners." Primary Health Care Research & Development, vol. 19, no. 02, 2017, pp. 131–139.

7. Kataoka, Hiroshi, and Satoshi Ueno. "Low FAB Score as a Predictor of Future Falling in Patients with Parkinson's Disease: a 2.5-Year Prospective Study." *Journal of Neurology*, vol. 262, no. 9, 2015, pp. 2049–2055.

8. Sharma, Anjali, et al. "Frailty as a Predictor of Falls in HIV-Infected and Uninfected Women." *Antiviral Therapy*, 2019.

9. imblearn Package SMOTE. URL: https://imbalanced-learn.readthedocs.io/en/stable/generated/imblearn.over_sampling.SMOTE.html

10. imblearn Package Tomek Link. URL: https://imbalanced-learn.readthedocs.io/en/stable/generated/imblearn.under_sampling.TomekLinks.html

11. imblearn Package Instance Hardness Threshold. URL: https://imbalanced-learn.readthedocs.io/en/stable/generated/imblearn.under_sampling.InstanceHardnessThreshold.html

12. Sklearn Package RFECV. URL: https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.RFECV.html

13. Sklearn Package Logistic Regression CV. URL: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegressionCV.html