Colorectal Cancer Dataset

FINAL PROJECT PRESENTATION

Seminar on Artificial Intelligence 2 NAIL052

Morelli Davide - Rishikesh Kumar

A little refresh on what was done so far...

EXPLORATORY DATA ANALYSIS

- Dataset has been cleaned
- Features have been correctly encoded

CLASSIFICATION OF
CANCER STAGE

What we tried to do next

We tried to work in two parallel directions:

Trying to improve the classification of CANCER STAGE

Trying new tasks:
classification of mortality/survival prediction and regression on healthcare cost/ mortality rate

Trying to improve the classifier

How **bad** the model really was?

RANDOM FOREST
Accuracy = 0.39

Classification Report:

	precision	recall	f1-score	support
0	0.40	0.51	0.45	16745
1	0.20	0.00	0.01	8430
2	0.39	0.48	0.43	16700
accuracy			0.40	41875
macro avg	0.33	0.33	0.30	41875
weighted avg	0.36	0.40	0.35	41875

Only slightly better than random choice

Trying to improve the classifier

How **bad** the model really was?

RANDOM FOREST
Accuracy = 0.39

```
Classification Report:
              precision    recall  f1-score   support

           0       0.40      0.51      0.45     16745
           1       0.20      0.00      0.01      8430
           2       0.39      0.48      0.43     16700

    accuracy                           0.40     41875
   macro avg       0.33      0.33      0.30     41875
weighted avg       0.36      0.40      0.35     41875
```

Only slightly better than random choice

What could the **issue** be?

1

2

3

Trying to improve the classifier

How **bad** the model really was?

RANDOM FOREST
Accuracy = 0.39

```
Classification Report:
              precision    recall  f1-score   support

           0       0.40      0.51      0.45     16745
           1       0.20      0.00      0.01      8430
           2       0.39      0.48      0.43     16700

    accuracy                           0.40     41875
   macro avg       0.33      0.33      0.30     41875
weighted avg       0.36      0.40      0.35     41875
```

Only slightly better than random choice

What could the **issue** be?

1 **unbalanced classes**

2

3

Trying to improve the classifier

How **bad** the model really was?

RANDOM FOREST
Accuracy = 0.39

Classification Report:

	precision	recall	f1-score	support
0	0.40	0.51	0.45	16745
1	0.20	0.00	0.01	8430
2	0.39	0.48	0.43	16700
accuracy			0.40	41875
macro avg	0.33	0.33	0.30	41875
weighted avg	0.36	0.40	0.35	41875

Only slightly better than random choice

What could the **issue** be?

1 **unbalanced classes**

wrong algorithm

2

3

Trying to improve the classifier

How **bad** the model really was?

RANDOM FOREST
Accuracy = 0.39

```
Classification Report:
              precision    recall  f1-score   support

           0       0.40      0.51      0.45     16745
           1       0.20      0.00      0.01      8430
           2       0.39      0.48      0.43     16700

    accuracy                           0.40     41875
   macro avg       0.33      0.33      0.30     41875
weighted avg       0.36      0.40      0.35     41875
```

Only slightly better than random choice

What could the **issue** be?

1 **unbalanced classes**

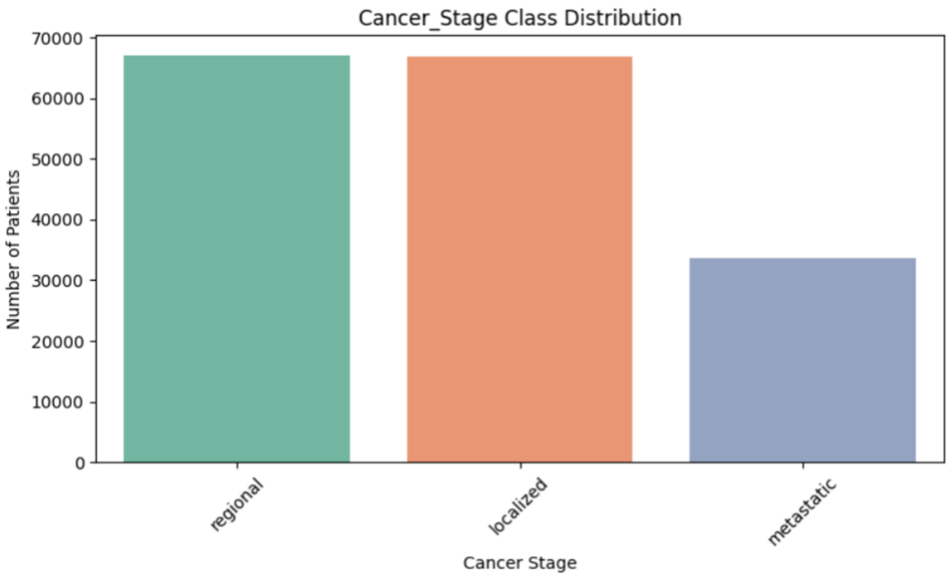
wrong algorithm

2 **feature engineering needed**

3

Trying to improve the classifier

How are the classes unbalanced?

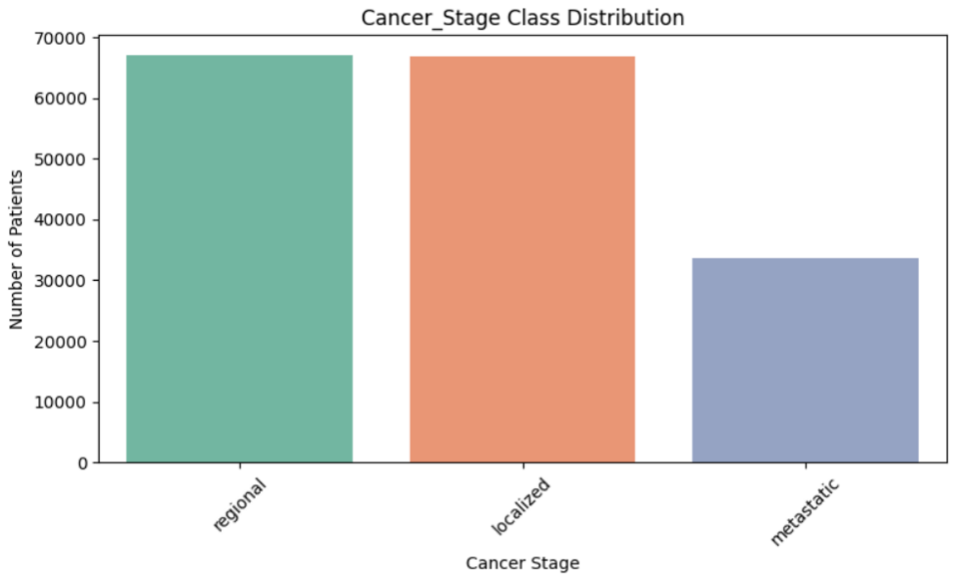


Percentage distributions:

REG: 39.99% LOC: 39.88% MET: 20.13%

Trying to improve the classifier

How are the classes unbalanced?



Percentage distributions:

REG: 39.99% LOC: 39.88% MET: 20.13%

How can we solve this?

SMOTE
(Synthetic Minority Over-sampling Technique)

PERFORMANCE DIDN'T IMPROVE

Accuracy = 0.381

Trying to improve the classifier

Trying new, different models:

Naive-Bayes

Classification Report:

	precision	recall	f1-score	support
localized	0.40	0.41	0.40	13360
metastatic	0.21	0.26	0.23	6744
regional	0.40	0.35	0.37	13396
accuracy			0.35	33500
macro avg	0.34	0.34	0.34	33500
weighted avg	0.36	0.35	0.36	33500

Accuracy: 0.353

Trying to improve the classifier

Trying new, different models:

Logistic regression

```
Classification Report:
              precision    recall  f1-score   support

   localized       0.40      0.42      0.41     13360
  metastatic       0.22      0.11      0.14      6744
    regional       0.40      0.48      0.44     13396

    accuracy                           0.38     33500
   macro avg       0.34      0.34      0.33     33500
weighted avg       0.37      0.38      0.37     33500
```

Accuracy: 0.381

Trying to improve the classifier

Trying new, different models:

k-NN (k = 5)

Classification Report:

	precision	recall	f1-score	support
localized	0.40	0.53	0.45	13360
metastatic	0.20	0.17	0.18	6744
regional	0.40	0.30	0.35	13396
accuracy			0.37	33500
macro avg	0.33	0.33	0.33	33500
weighted avg	0.36	0.37	0.36	33500

Accuracy: 0.366

Trying to improve the classifier

New feature engineering ideas

- (CLASS SUGGESTION) Ignoring features that are outcome-dependant
- Continuous feature quantization using bins
- Ordinal encoding of Obesity_BMI

	Obesity_BMI	Obesity_BMI_encoded	Age	Age_Band	Tumor_Size_mm	Tumor_Size_Category
0	Overweight	2	77	60-79	69	Large
1	Overweight	2	59	40-59	33	Medium
2	Normal	1	66	60-79	17	Small
3	Obese	3	83	80+	14	Small
4	Normal	1	66	60-79	34	Medium

Trying to improve the classifier

How did the models improve?
ACCURACY

- Naive bayes: **0.353**

0.396

- Random forest: **0.39**

0.397

XGBoost: **0.401**

- k-NN: **0.366**

Decision tree: **0.359**

Other tasks:

Other tasks:

Survival prediction

classification

```
LogisticRegression: ACC=0.600, ROC_AUC=0.498
RandomForest: ACC=0.584, ROC_AUC=0.501
GradientBoosting: ACC=0.600, ROC_AUC=0.498
MLPClassifier: ACC=0.564, ROC_AUC=0.501
```

Other tasks:

Survival prediction
classification

```
LogisticRegression: ACC=0.600, ROC_AUC=0.498
RandomForest: ACC=0.584, ROC_AUC=0.501
GradientBoosting: ACC=0.600, ROC_AUC=0.498
MLPClassifier: ACC=0.564, ROC_AUC=0.501
```

Mortality prediction
classification

```
LogisticRegression: ACC=0.599, ROC_AUC=0.498
RandomForest: ACC=0.586, ROC_AUC=0.501
GradientBoosting: ACC=0.599, ROC_AUC=0.502
MLPClassifier: ACC=0.557, ROC_AUC=0.504
```

Other tasks:

Survival prediction
classification

```
LogisticRegression: ACC=0.600, ROC_AUC=0.498
RandomForest: ACC=0.584, ROC_AUC=0.501
GradientBoosting: ACC=0.600, ROC_AUC=0.498
MLPClassifier: ACC=0.564, ROC_AUC=0.501
```

Mortality prediction
classification

```
LogisticRegression: ACC=0.599, ROC_AUC=0.498
RandomForest: ACC=0.586, ROC_AUC=0.501
GradientBoosting: ACC=0.599, ROC_AUC=0.502
MLPClassifier: ACC=0.557, ROC_AUC=0.504
```

Healthcare costs
regression

```
LinearRegression: MSE=747066258.45, R2=-0.001
RandomForestReg: MSE=760362250.74, R2=-0.018
GradientBoostingReg: MSE=747256776.59, R2=-0.001
MLPRegressor: MSE=748217485.37, R2=-0.002
```

Other tasks:

Survival prediction
classification

```
LogisticRegression: ACC=0.600, ROC_AUC=0.498
RandomForest: ACC=0.584, ROC_AUC=0.501
GradientBoosting: ACC=0.600, ROC_AUC=0.498
MLPClassifier: ACC=0.564, ROC_AUC=0.501
```

Mortality prediction
classification

```
LogisticRegression: ACC=0.599, ROC_AUC=0.498
RandomForest: ACC=0.586, ROC_AUC=0.501
GradientBoosting: ACC=0.599, ROC_AUC=0.502
MLPClassifier: ACC=0.557, ROC_AUC=0.504
```

Healthcare costs
regression

```
LinearRegression: MSE=747066258.45, R2=-0.001
RandomForestReg: MSE=760362250.74, R2=-0.018
GradientBoostingReg: MSE=747256776.59, R2=-0.001
MLPRegressor: MSE=748217485.37, R2=-0.002
```

Mortality rate
regression

```
LinearRegression: MSE=52.26, R2=-0.001
RandomForestReg: MSE=53.15, R2=-0.018
GradientBoostingReg: MSE=52.27, R2=-0.001
MLPRegressor: MSE=55.46, R2=-0.062
```

Deep learning

We tried to use Deep Learning to discover the best parameters for our new tasks

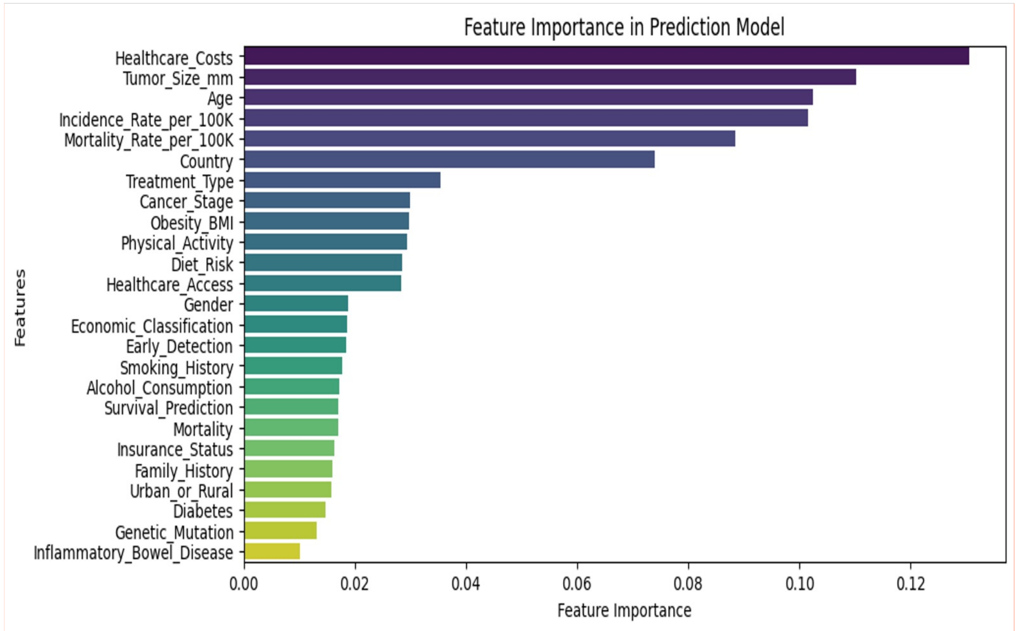
```
-- DL Survival Model --
Epoch 1/30
4188/4188 - 5s - 1ms/step - accuracy: 0.5980 - auc: 0.5014 - loss: 0.6753 - val_accuracy: 0.5996 - val_auc: 0.4971 - val_loss: 0.6734
Epoch 2/30
4188/4188 - 4s - 855us/step - accuracy: 0.5996 - auc: 0.5002 - loss: 0.6736 - val_accuracy: 0.5996 - val_auc: 0.5026 - val_loss: 0.6732
Epoch 3/30
4188/4188 - 4s - 877us/step - accuracy: 0.5996 - auc: 0.5032 - loss: 0.6733 - val_accuracy: 0.5996 - val_auc: 0.4996 - val_loss: 0.6732
Epoch 4/30
4188/4188 - 4s - 856us/step - accuracy: 0.5996 - auc: 0.5062 - loss: 0.6732 - val_accuracy: 0.5996 - val_auc: 0.5035 - val_loss: 0.6733
Epoch 5/30
4188/4188 - 4s - 895us/step - accuracy: 0.5996 - auc: 0.5060 - loss: 0.6732 - val_accuracy: 0.5996 - val_auc: 0.4964 - val_loss: 0.6732
Epoch 6/30
4188/4188 - 4s - 854us/step - accuracy: 0.5996 - auc: 0.5065 - loss: 0.6731 - val_accuracy: 0.5996 - val_auc: 0.5020 - val_loss: 0.6735
Epoch 7/30
4188/4188 - 3s - 834us/step - accuracy: 0.5996 - auc: 0.5119 - loss: 0.6730 - val_accuracy: 0.5996 - val_auc: 0.4977 - val_loss: 0.6734
DL Model: ACC=0.600, AUC=0.503
```

This brought no improvement compared to the previous results

Feature importance

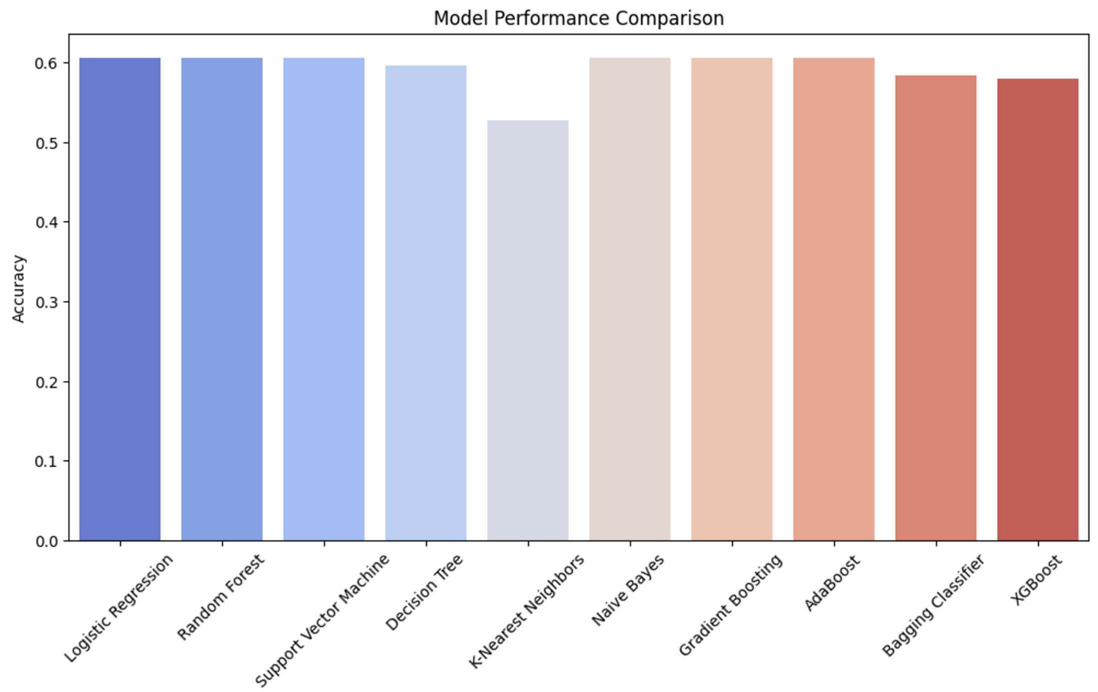
Which are the best features to focus on?

FEATURE SELECTION
using
GridSearchCV



Try again with important features

We tried the same trainings as before but with only the **most important features** and this time we **increased the amount of models** used



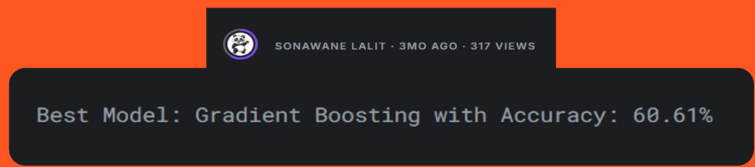
Best model is Gradient boosting

Worst model is k-NN

Comparing with others on Kaggle

For Cancer_Stage prediction we didn't find any other notebook so we have only our result to go with

We found another notebook analysing Survival_Prediction



So our result is in line with theirs, also the best model is for both **Gradient Boosting**

Thank you

for the attention