Exercise 3

Radiomics

October 13, 2025

Info

Overview
In this exercise, you will work with a real radiomics dataset of non-small cell lung cancer (NSCLC) [1].
To extract radiomic features, images were resampled in 3D to an isotropic voxel size of 0.98 mm^3 using the B-spline interpolation method for images and B-spline interpolation with a threshold of 0.5 for masks. Radiomic features were extracted using a 3D averaging aggregation method with a fixed number of bins (16 bins).

Learning objectives
By the end of this exercise, you should be able to:

- identify and remove highly correlated features,
- apply different dimensionality reduction methods,
- perform univariate statistical analysis,
- conduct multivariate analysis and evaluate model performance.

1 Data

You are provided with two CSV files: one file contains the clinical information, the other file contains radiomic data.

2 Endpoint

The goal of the exercise is to build a binary classification model relying on radiomic data. Choose one of the following endpoints:

- Survival/death at 18 months
- Prediction of one of the histology types: squamous cell carcinoma, large cell or adenocarcinoma.

If you choose prediction of the histology type, you can drop the entries without specified histology type or histology type described as NOS.

3 Exercises

3.1 Dimensionality Reduction

The data set might contain highly correlated features. Reduce the feature correlation in the data set. Explain which correlation reduction method you selected and why. Also, describe why highly correlated features are not desirable in the final feature set.

3.2 Univariate Analysis

Before performing multivariate modeling, it is useful to assess the contribution of each feature to predictive performance individually.

Use the Mann–Whitney U test to evaluate feature significance (significance level $p < 0.05$) and calculate the area under the ROC curve (ROC-AUC). The ROC-AUC can be calculated directly from the U statistic using the following formula

$$AUC = \frac{U}{n_1 n_2}$$

where n_1 and n_2 are the simple sizes of the positive and the negative classes, respectively.

What are the highest AUC values of individual features for your chosen endpoint? Plot the ROC curve for the feature with the highest AUC. What does it mean when $p < 0.05$?

3.3 Multivariate Analysis

The next step is to build a multivariate predictive model that incorporates multiple radiomic features. For the endpoint, pick the same target as the one used for the previous exercise.

Try to implement various steps discussed during the lecture, e.g.:

- data cleaning
- imbalance reduction (if present)
- feature selection
- different classifiers

Useful packages:

- scikit-learn
- imbalanced-learn
- xgboost

IMPORTANT: If you decide to use scikit-learn with data cleaning/resampling methods from imbalanced-learn package, make sure to use pipeline class from imbalanced-learn, because pipeline from scikit-learn will not work.

```
from imblearn.pipeline import Pipeline
```

After training the final model, vvaluate the performance of the model using ROC curves and PR curves. Try to interpret the most important features that contribute to the performance of the model.

Here's a basic template that you can (but don't have to) use.

```
from sklearn.feature_selection import SelectKBest, f_classif
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import RepeatedStratifiedKFold,
    RandomizedSearchCV
from sklearn.model_selection import cross_validate
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler

# X = radiomics data
# y = endpoint encoded as a binary vector

RANDOM_STATE = 42

pipe = Pipeline([
    ("scale", StandardScaler()),
    ("clf", LogisticRegression(
        penalty="l2",
```

```
        solver="liblinear",
        max_iter=5000,
        random_state=RANDOM_STATE
    )),
])

param_dist = {
    "clf__C": [0.01, 0.03, 0.1, 0.3, 1.0, 3.0, 10.0],
}

inner_cv = RepeatedStratifiedKFold(n_splits=5, n_repeats=2,
    random_state=RANDOM_STATE)  # feel free to change n_splits and
    n_repeats
rs = RandomizedSearchCV(
    estimator=pipe,
    param_distributions=param_dist,
    n_iter=30,  # increase to 200 for larger param_dist spaces
    scoring="roc_auc",
    cv=inner_cv,
    n_jobs=-1,
    random_state=RANDOM_STATE,
    refit=True
)

outer_cv = RepeatedStratifiedKFold(n_splits=5, n_repeats=2,
    random_state=RANDOM_STATE)  # feel free to change n_splits and
    n_repeats
scores = cross_validate(rs, X, y, scoring='roc_auc', cv=outer_cv,
    return_train_score=True)
```

Hint

It's often a good idea to start from simple pipelines and increase complexity monitoring the changes in model performance.

Appendix A — References

1. https://www.cancerimagingarchive.net/collection/nsclc-radiomics/
2. https://scikit-learn.org/stable/
3. https://imbalanced-learn.org/stable/index.html
4. https://xgboost.readthedocs.io/en/stable/