Objectif of the project :

Breast calcifications are calcium deposits that form in the breast tissue. They are not related to the amount of calcium taken in the diet or obtained through supplements.  
Breast calcifications are quite common and most are not associated with cancer. To be sure, the radiologist looks at their size, shape and arrangement on a mammogram, where they often appear as small white spots. Some of their characteristics, such as an irregular shape or certain groupings, may be suspicious.  
There are two types of calcifications: macrocalcifications and microcalcifications.  
Macrocalcifications are large deposits of calcium in the breast. They are more common in women over 50 years old. They are often associated with benign changes in the breast, such as aging of the breast arteries, old lesions, inflammation or masses such as fibroadenoma. For this reason, when these macrocalcifications are found, the radiologist does not routinely recommend a biopsy.  
Microcalcifications are tiny calcium deposits in the breast. Their presence sometimes means that the activity of certain cells in the breast is increased. A more active cell takes in more calcium than a less active one.  
Microcalcifications can raise suspicion of breast cancer (such as ductal carcinoma in situ - DCIS), especially when they appear alone or in clusters on mammography. If this is the case, the radiologist recommends a biopsy to make sure it is not cancer.

The objectif of this project is to be able to detect breast cancer from images. These images are the result of ct-scans, and the problem is a binary classification problem. The dataset is representative of wavelet analysis of 3562 images which represent 96 cases or 96 patients. The dataset containts 150 attributes or features representing radiomic data of the microcalcification.

Ein Bild, das Krater, Dunkel enthält.

Automatisch generierte Beschreibung

Based on 50 properties computed on the 3500 micros, classify them as benign or malignant. Problem : Every subject in the dataset presents multiple micros and we only know for sure whether the subject has breast cancer or not.

So 50 benign cases and 50 malignant cases result in 3700 micros in total.

The objectif is to classify individual micros assuming all micros per subject have the same label, and to classify whether a subject has cancer based on the classification of the multiple micros per subject.

Prepping data :

Part 2 :

* Checking if there’s no na value
* Splitting the data into 2 arrays :
  + positive labels array then turning them into a dataframe
  + negative labels array then turning them into a dataframe

Learning :

SVM :

SVM works by mapping data to a high-dimensional feature space so that data points can be categorized, even when the data are not otherwise linearly separable. A separator between the categories is found, then the data are transformed in such a way that the separator could be drawn as a hyperplane. Following this, characteristics of new data can be used to predict the group to which a new record should belong.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Precision | Recall | F1-score |
| 0 | 0.81 | 0.88 | 0.84 |
| 1 | 0.78 | 0.68 | 0.73 |

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

Mean training loss : 0.211 std: 0.047

Mean validation loss: 0.366 std: 0.184

XGBoost :

XGBoost is a supervised machine learning method for classification and regression and is used by the [Train Using AutoML](https://pro.arcgis.com/en/pro-app/3.1/tool-reference/geoai/train-using-automl.htm) tool. XGBoost is short for extreme gradient boosting. This method is based on decision trees and improves on other methods such as random forest and gradient boost. It works well with large, complicated datasets by using various optimization methods.

To fit a training dataset using XGBoost, an initial prediction is made. Residuals are computed based on the predicted value and the observed values. A decision tree is created with the residuals using a similarity score for residuals. The similarity of the data in a leaf is calculated, as well as the gain in similarity in the subsequent split. The gains are compared to determine a feature and a threshold for a node. The output value for each leaf is also calculated using the residuals. For classification, the values are typically calculated using the log of odds and probabilities. The output of the tree becomes the new residual for the dataset, which is used to construct another tree. This process is repeated until the residuals stop reducing or for a specified number of times. Each subsequent tree learns from the previous trees and is not assigned equal weight, unlike how [Random Forest](https://pro.arcgis.com/en/pro-app/3.1/tool-reference/geoai/how-random-trees-classification-and-regression-works.htm) works.

To use this model for prediction, the output from each tree multiplied by a learning rate is added to the initial prediction to arrive at a final value or classification.

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

|  |  |  |  |
| --- | --- | --- | --- |
|  | Precision | Recall | F1-score |
| 0 | 0.84 | 0.83 | 0.84 |
| 1 | 0.74 | 0.76 | 0.75 |

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

Mean training loss: 0.527 std: 0.145

Mean validation loss: 0.384 std: 0.367

----------------------------------

Log reg :

Logistic regression is a data analysis technique that uses mathematics to determine the relationships between two data factors. This relationship is then used to predict the value of one of these factors based on the other. The prediction usually has a limited number of outcomes, such as yes or no.  
  
For example, let's say you want to guess whether or not your website visitor will click the checkout button in their shopping cart. Logistic regression analysis examines the behavior of past visitors, such as time spent on the website and number of items in the cart. It finds that visitors who have spent more than five minutes on the website in the past and added more than three items to the cart have clicked the checkout button. Using this information, the logistic regression function can then predict the behavior of a new website visitor.  
  
Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

|  |  |  |  |
| --- | --- | --- | --- |
|  | Precision | Recall | F1-score |
| 0 | 0.84 | 0.84 | 0.84 |
| 1 | 0.75 | 0.75 | 0.75 |

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

Mean training loss: 0.222 std: 0.05

Mean validation loss: 0.384 std: 0.177

Bagging :

Bagging, also known as bootstrap aggregation, is the ensemble learning method that is commonly used to reduce variance within a noisy dataset. In bagging, a random sample of data in a training set is selected with replacement—meaning that the individual data points can be chosen more than once. After several data samples are generated, these weak models are then trained independently, and depending on the type of task—regression or classification, for example—the average or majority of those predictions yield a more accurate estimate.

As a note, the random forest algorithm is considered an extension of the bagging method, using both bagging and feature randomness to create an uncorrelated forest of decision trees.

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

|  |  |  |  |
| --- | --- | --- | --- |
|  | Precision | Recall | F1-score |
| 0 | 0.82 | 0.86 | 0.84 |
| 1 | 0.76 | 0.71 | 0.73 |

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

Mean training loss : 0.222 std: 0.05

Mean validation loss : 0.384 std: 0.177

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

Prepping data :

Part 1 :

* Visualise data

Svm :

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

|  |  |  |  |
| --- | --- | --- | --- |
|  | Precision | Recall | F1-score |
| 0 | 0.75 | 0.88 | 0.81 |
| 1 | 0.79 | 0.61 | 0.69 |

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

Mean training loss: 0.206 std: 0.004

Mean validation loss: 0.247 std: 0.017

Log reg :

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

|  |  |  |  |
| --- | --- | --- | --- |
|  | Precision | Recall | F1-score |
| 0 | 0.76 | 0.83 | 0.79 |
| 1 | 0.74 | 0.65 | 0.69 |

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

Mean training loss: 0.216 std: 0.005

Mean validation loss: 0.25 std: 0.016

Bagging :

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

|  |  |  |  |
| --- | --- | --- | --- |
|  | Precision | Recall | F1-score |
| 0 | 0.76 | 0.87 | 0.81 |
| 1 | 0.79 | 0.63 | 0.70 |

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

Mean training loss: 0.001 std: 0.0

Mean validation loss: 0.225 std: 0.018

**Adaboost**

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

|  |  |  |  |
| --- | --- | --- | --- |
|  | Precision | Recall | F1-score |
| 0 | 0.74 | 0.82 | 0.78 |
| 1 | 0.72 | 0.62 | 0.67 |

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

Mean training loss: 0.189 std: 0.006

Mean validation loss: 0.259 std: 0.016

XGBOOST :

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

precision recall f1-score support

0 0.74 0.84 0.79 335

1 0.74 0.61 0.67 253

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

Mean training loss: 0.332 std: 0.087

Mean validation loss: 0.315 std: 0.09

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung