**Introduction**

Breast cancer is one of the most common cancers and the second most common cause of cancer deaths. The chance that a woman will die from breast cancer is about 1 in 39 (about 2.6%) (ACS, 2019). Since 2007, breast cancer death rates have been steady in women younger than 50 but have continued to decrease in older women. From 2013 to 2018, the death rate decreased by 1% per year. These decreases are believed to be the result of finding breast cancer earlier through screening and increased awareness, as well as better treatments.

The early diagnosis of breast cancer can improve the prognosis and chance of survival notably, as it may be a help to promote early treatment to the affected person. In its early stages, breast cancer may not cause any symptoms. In many cases, a tumor may be too small to be felt, but an abnormality can still be seen on a mammogram. If a tumor can be felt, the first sign is usually a new lump in the breast that was not there before. However, not all lumps are cancer.

The Decision Support System aims to analyze how the different features correlates in order to predict whether a person have a breast cancer. To achieve this, we used a Machine Learning classification method to attain a highly accurate prediction using the programming language python.

**Functions**

**def \_\_init\_\_(self, lr=0.001, n\_iters=1000)**

**Description**

this function is used when an object is created from the class and it will allow the class to initialize the attributes of a class.

**Parameters**

Self -> represent the instance

lr -> Learning rate of the program

n\_iters -> number of times the program learns

**def fit(self, x, y)**

**Description**

This function involves the training step and the gradient descent

**Parameters**

Self -> represent the instance

x -> training sample

x -> value of the training labels

**def predict(self, X)**

**Description**

This function uses the sigmoid model to predict the outcome needed to be produced by the program.

**Parameters**

Self -> represent the instance

X -> new test samples to predict

**def \_sigmoid(self, x)**

**Description**

This function represent the mathematical equation of the sigmoid model that is used by the program.

**Parameters**

Self -> represent the instance

x -> training sample

**Programming Language Libraries**

**Pandas** is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language.

**NumPy** is a Python library used for working with arrays. It also has functions for working in domain of linear algebra, Fourier transform, and matrices.

**Plotly** is Python graphing library makes interactive, publication-quality graphs. Examples of how to make line plots, scatter plots, area charts, bar charts, error bars, box plots, histograms, heatmaps, subplots, multiple-axes, polar charts, and bubble charts.

**Matplotlib** is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack.

**Training**

The training of the model is where it learns the good values of the weights and bias while also increasing the accuracy while predating the results.. We used 80% of the data to train the program to acquire a better learning for the program

**Test**

We used the 20% of the data to test the model to check the accuracy and validate the model prediction. Testing data is a bit different from training data, as it is a kind of sample of data used for an unbiased evaluation of a final model fit on the training dataset to check model functioning.