Report Group 8

# Medical Image Analysis

# 8DC00

## Project 2: Image classification using different classification methods

Luuk Brouwer: 1899333

Karlijn Cornelissen: 1737104

Marit Janssen: 1841211

Manouk Mels: 1897497

## Introduction

One in seven Dutch women will get breast cancer in her lifetime. Over 3000 women in the Netherlands die of breast cancer every year*.* The 5-year relative survival rate is 99% if the cancer is detected early, which means a timely diagnosis can make all the difference, underscoring the critical importance of early detection (*Kans op Borstkanker*, 2023).

A crucial aspect of diagnosing breast cancer involves analyzing cancer cells, particularly their nuclei. The nuclei of cancer cells and how different they are from regular cells are indications for the stage of the cancer. Greater variations often signify more advanced and aggressive forms of cancer (Du Cancer, z.d.).

To effectively treat and save as many patients as possible, pathologists have to efficiently analyze potential cancer cells. However, manually analyzing these cells can be a highly challenging and time-consuming task. The complexity and variability of cancer cells make it difficult to consistently differentiate between benign and malignant cells, especially in the early stages. Subtle differences in cell morphology can be easily overlooked, leading to misdiagnoses. To help achieve as many accurate diagnoses as possible, image analysis can be a useful tool.

Various machine learning image classification analysis methods, such as logistic regression, neural networks, and k-nearest neighbors (k-NN) are available, in this case, specifically microscopic images of cancer cells. In this research, these methods will be tested on their ability to classify cancer cells, assessed by false negatives. The reason for choosing this assessment method is that in the opinion of the researchers, a false negative(missing cancer cases) is worse than a false positive (identifying cancer when it's not present) or overall accuracy when it comes to cancer. This study aims to answer to following research question:

“Which classification machine learning method— logistic regression, neural networks, or k-nearest neighbors—demonstrates better performance in classifying cancer cells based on the reduction of false negatives?"

## Method

### Linear regression

The microscopy image data consists of training and test data. Each dataset contains 24x24 pixel images with three color channels. The images are reshaped so that each pixel serves as an individual feature, transforming the images into vectors of the features, which are used as inputs for the model training.

The model is a linear regression model to predict the area (size) of nuclei based on the image data. Initially, the model is trained using the entire available training dataset, each pixel and color channel is treated as a feature. Once the model is trained it is used to predict the areas of nuclei in the test dataset. The predicted and actual areas are plotted, to evaluate the trained model. In addition to training the model on the full dataset, a second model is trained using a reduced subset of the training data. Specifically, every fourth sample from the training set is selected to form the reduced dataset. The reduced model is also used to predict the areas of nuclei in the test dataset. The effect of reducing the training dataset size on the performance of the model is evaluated.

To assess both models' accuracy, the predicted nuclei areas are compared with the true areas provided in the test set. The mean squared error is calculated for both models.

A similar dataset is used for classification, which can be done with the use of different models. In this report, we explore three classification models—Logistic Regression, Neural Networks, and k-nearest neighbors (k-NN)—to tackle the task of nuclei classification in microscopy. Each model offers unique strengths and methodologies for effectively categorizing images based on the underlying data patterns.

### Logistic regression

Nuclei classification in microscopy using a logistic regression model. A subset of training samples is used in each iteration to update the model parameters (batch size). The model is trained using gradient descent. Which allows the model to learn from the data by adjusting the parameters iteratively so that the loss function is minimized (Donges, 2024).

The batch size is the number of images used in each epoch to train the network. If this hyperparameter is set too large, the network may take too long to converge (no more increase in accuracy); on the other hand, if it is set too small, the network will bounce back and forth without achieving acceptable performance (Olamendy, 2024).

The logistic regression loss is calculated for the selected batch and next the parameters are updated for each iteration, each time both the training and validation loss are written down.

Using trial and error the initial values of the parameters were determined. The following parameters are used:

* Learning rate: 0.0001
* Batch size, 30
* Number of iterations: 5000

These seem to result in the best performance.

To determine when to stop the training process, we implement a stopping criterion based on the change in validation loss over succeeding iterations. Specifically, training will stop when the difference in validation loss between the last nine iterations is less than 0.001. This criterion indicates that the model is no longer experiencing significant performance improvements, suggesting that further training is unlikely to give better results.

Throughout the training process the loss for both the training and validation are visualized, this provides real-time feedback on the performance of the model. For example, if the model is overfitting the data.

Classification accuracy is calculated according to the formula 1:

(1)

The predicted values are calculated using the dot product of the input features x and the model parameters .

(2)

To reduce the size of the training set, a new function is written where first the index of every xth value is taken, depending on how much smaller the new data set has to be. Afterwards only the data on every xth row is copied into the smaller data set. However, reducing the dataset result

### Neural network

The next method we will be using for this classification problem is a neural network.

We will use a fully connected neural network with two layers (one hidden layer), each with a sigmoid activation function. These activation functions are necessary to add non-linearity to the neural network. Without these, it would be calculating a linear transformation. Because this is a classification problem, there is only one output variable. This output variable tells us in which class the image belongs: it will be 1 for a large nucleus and 0 for a small nucleus.

A layer of a neural network has the following workflow:

The input feature vector is multiplied by a vector weight vector, which contains a set of weights that determine the importance of each input feature. The result is a new set of features, which will then be multiplied by the sigmoid activation function. The resulting vector is the input for the next layer or the output of the neural network.

These weight vectors are the unknown variables of the neural network and can be determined with the help of the gradient descent method, and input data for which the output values are known.

Training of the model works as follows:

The model is trained using the gradient of the loss function. The loss function that is used is the squared difference between the predicted outcome and the desired outcome . See formula 3.

(3)

We use the training dataset and random initial values for the weights to determine the gradient of the loss function concerning the weights of each layer. We then update the weights towards a lower loss in the training dataset. See formula 4, we scale with the learning rate , to control this learning process.

(4)

Another way to control the learning process is to use only a batch of the training data instead of the whole dataset at once. This reduces the chance of ending up in a local optimum of the loss function.

Determining suitable training parameters:

The parameters of the model that impact the learning process are: the number of iterations, for which the weights are updated; the learning rate; the batch size; and the number of features in the hidden layer. We have chosen to limit the number of iterations to 150, because this gives a reasonable training time, and is usually enough to reach a steady loss function. The other parameters are chosen based on the results of a short experiment.

First, the effect of the learning rate and the batch size on the validation dataset's accuracy are investigated. This is done by comparing the accuracy of the predictions for the validation dataset that was reached in 150 iterations, for different learning rate and batch size combinations. A hidden layer that consisted of 1000 features was used. The best performing combination of learning rate and batch size was kept, and used for testing the effect of the size of the hidden layer.

### K-nearest neighbors

K - nearest neighbors or kNN is a supervised classification method based on the closest ‘neighbors’ of a test sample. The (training) data is clustered into several meaningful classes. When a test sample is put into the model, the model will calculate the k ‘nearest neighbors’, meaning the data samples that are closest to the test sample in terms of Euclidean distance. The class that occurs the most in the list of k neighbours, is the class that the test sample will be put into (Harrison, 2019).

In this research, k was chosen to be x, since blablabla

<https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761>

### Principle component analysis

Principal component analysis (PCA) can reduce the dimensionality of data, making it easier to classify. PCA is used in complex data structures with numerous correlated variables. A principle component is a linear combination of multiple variables. Principle components have less (/no) correlations with each other and are optimal when the maximum variance in the data is achieved. Usually, the first few principle components capture the majority of the variance of the dataset, so using these as variables will reduce the complexity of the data, without discarding a lot of important information. The data varies the most in the direction of certain eigenvectors, these eigenvectors form the basis of the new, principle component space, which has fewer dimensions. The eigenvalue can indicate the importance of a certain PC, if the eigenvalue is higher, the PC is more valuable, since it captures a large part of the variance. Principle component analysis makes classifying more efficient since it reduces the dimensions of the data and also reduces redundancy in the data since the PC’s are not correlated.

<https://datascience.codata.org/articles/10.5334/dsj-2021-026>

### Assessment methods

When evaluating machine learning classification models, it is important to assess their performance using different metrics. These metrics help in understanding the strengths and weaknesses of the model in various contexts, especially where different types of errors have different costs. The assessments will be done based on the performance of the model on the test set.

#### Accuracy

(5?)

(Bressler, 2024)

Accuracy is the most intuitive method, but it can give a misleading picture of the performance of the model especially in cases of imbalanced datasets. Accuracy assumes a binary classification context (Zvornicanin, 2024).

Overall accuracy can be misleading, especially for cases where the class distribution is unbalanced and correctly predicting the minority class is important. In such situations, the model may achieve a high accuracy score by correctly predicting the majority class but consistently misclassifying the minority class, thus creating a false impression of good performance. Take, for example, a model for predicting cancer Misclassifying malignant as benign has serious consequences (Bressler, 2024).

Accuracy is most of the time used as an assessment method, since it’s the most intuitive. It is used as a rough indicator of the model training for balanced datasets. And is often used in combination with other metrics for model performance (Classification: Accuracy, Recall, Precision, And Related Metrics, z.d.). When there is an unbalanced dataset of misdiagnosis had a higher cost other evaluation metrics could be considered.

#### False positive rate (FPR)

(6?)

Used when false positives are more expensive than false negatives (Classification: Accuracy, Recall, Precision, And Related Metrics, z.d.).

#### Recall

(Jacobi, 2024)

Recall, or in other words false negatives, sensitivity or true positive rate, measures the ability of a model to correctly identify all the relevant instances. (Jacobi, 2024) False negatives occur when a model incorrectly classifies a positive instance as negative. In binary classification, this means the model fails to detect a positive class where it exists. If this is prioritized then missing positives is costly (Bressler, 2024). Recall focuses on minimizing false negatives. Used when false negatives are more expensive than false positives (Classification: Accuracy, Recall, Precision, And Related Metrics, z.d.).

#### Precision

(*Accuracy Vs. Precision Vs. Recall in Machine Learning: What’s The Difference?*, z.d.)

Focusses on minimizing false positives. (Jacobi, 2024) use when positive predictions need to be accurate (Classification: Accuracy, Recall, Precision, And Related Metrics, z.d.).

It works very well for a problem with unbalanced classes because it shows the accuracy of the model in identifying the target class. Precision is useful when the cost of a false positive is high. In this case, you usually want to be sure you can identify the target class (*Accuracy Vs. Precision Vs. Recall in Machine Learning: What’s The Difference?*, z.d.).

#### F1 score

Is a metric combining precision and recall. Used when a single metric to summarize the performance is needed. (Bressler, 2024)

The values of true positives, false positives, and false negatives all influence the F1 score. An increase in false positives reduces precision, while a rise in false negatives lowers recall, both contributing to a decline in the F1 score. Conversely, an increase in true positives improves both precision and recall, thereby enhancing the F1 score (*F1 Score in Machine Learning | Deepgram*, z.d.).

In essence, the F1 score helps strike a middle ground between catching all cancer cases and minimizing over-diagnosis, offering a more balanced metric compared to accuracy.

#### ROC-curve and AUC

The ROC curve (Receiver-operating characteristic curve) is a visual representation of model performance across all thresholds, during the training. In the curve the true positive rate (TPR) on y-axis, and false positive rate (FPR) on the x-axis are plotted. [What Is ROC Curve in Machine Learning? | Coursera](https://www.coursera.org/articles/what-is-roc-curve)

The area under the ROC curve is the AUC which measures the accuracy of the performance of machine learning models. This score ranges from 0.0 to 1.0, with 1.0 being a perfect classifier. [What Is ROC Curve in Machine Learning? | Coursera](https://www.coursera.org/articles/what-is-roc-curve)

““The ROC curve has no bias towards classifiers and remains independent of the conditions it works under, making it useful for predictions with both balanced and imbalanced problems. [What Is ROC Curve in Machine Learning? | Coursera](https://www.coursera.org/articles/what-is-roc-curve)””

[Receiver operating characteristic curve: overview and practical use for clinicians - PMC (nih.gov)](https://pmc.ncbi.nlm.nih.gov/articles/PMC8831439/)

#### Validation loss

This is the loss function used is the binary cross-entropy or log loss function (Godoy, 2022).

Another metric measures the error between the predicted outputs of a model (X) and the actual target values on a validation set(Y) (data unseen by the model during training), calculated using the mean squared error. It helps tracking a models performance over time. It can especially say when a model is overfitting or underfitting.

A confusion matrix can help pinpoint where the model is making mistakes.

## Results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Method | False negatives | Recall | Accuracy | Precision |
| Logistic regression |  |  |  |  |
| Neural network | 2164 | 0.781 | 0.806 | 0.805 |
| KNN |  |  |  |  |

## Discussion

* Not every assessment method made use of the same parameters, because different values for these parameters such as …. Delivered the optimal result for each method.
* Verschillende parameters bij verschillende methodes
* Cut off point bij positive, negative, na sigmoid.
* Metric is erg afhankelijk van wat voor data set je hebt, wat je zoekt: voorbeeld uit bron*: 1) A model that labels all instances as positive will have a perfect Recall of 1.0 because it doesn’t miss any positive instance. However, this model will have poor Precision because it falsely classifies many negatives as positives, thereby decreasing its Precision. 2) Conversely, a model that labels very few instances as positive but is correct whenever it does will have high Precision but low Recall. That’s because it misses many actual positive instances.* [*Recall: Machine Learning Evaluation Metric (aporia.com)*](https://www.aporia.com/learn/recall-a-key-metric-for-evaluating-model-performance/#:~:text=Recall%2C%20also%20known%20as%20Sensitivity,positives%20our%20model%20correctly%20identified.)
* It is useful to use multiple assessment metrics for a model, depending on the desired function of the model. For example, a model can have 0 false negatives, but still not be very accurate because it classifies almost everything as positive. The model might seem perfect because of the zero false negatives, but this is misleading information. To prevent this type of misleading, multiple assessment metrics should be taken into account for the assessment of any model.

## References

*Accuracy vs. precision vs. recall in machine learning: what’s the difference?* (z.d.). https://www.evidentlyai.com/classification-metrics/accuracy-precision-recall

Bressler, N. (2024, 10 juni). *How to Check the Accuracy of Your Machine Learning Model | Deepchecks*. Deepchecks. https://www.deepchecks.com/how-to-check-the-accuracy-of-your-machine-learning-model/

*Classification: Accuracy, recall, precision, and related metrics*. (z.d.). Google For Developers. https://developers.google.com/machine-learning/crash-course/classification/accuracy-precision-recall

Donges, N. (2024, 1 augustus). *Gradient Descent in Machine Learning: A Basic Introduction*. Built In. https://builtin.com/data-science/gradient-descent

Du Cancer, C. C. S. /. S. C. (z.d.). *Grading breast cancer*. Canadian Cancer Society. https://cancer.ca/en/cancer-information/cancer-types/breast/grading

*Evidently AI - Classification metrics guide*. (z.d.). https://www.evidentlyai.com/classification-metrics#:~:text=Accuracy%20reflects%20the%20overall%20correctness,class%20detected%20by%20the%20model.

*F1 score in Machine Learning | Deepgram*. (z.d.). Deepgram. https://deepgram.com/ai-glossary/f1-score-machine-learning

Godoy, D. (2022, 10 juli). Understanding binary cross-entropy / log loss: a visual explanation. *Medium*. <https://towardsdatascience.com/understanding-binary-cross-entropy-log-loss-a-visual-explanation-a3ac6025181a>

Harrison, O. (2019, 14 juli). Machine Learning Basics with the K-Nearest Neighbors Algorithm. *Medium*. https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761

Jacobi, O. (2024, 8 september). *Recall: A Key Metric for Evaluating Model Performance*. Aporia. https://www.aporia.com/learn/recall-a-key-metric-for-evaluating-model-performance/#:~:text=Recall%2C%20also%20known%20as%20Sensitivity,positives%20our%20model%20correctly%20identified.

*Kans op borstkanker*. (2023, 15 juni). RIVM. https://www.rivm.nl/bevolkingsonderzoek-borstkanker/over-borstkanker/kans-op-borstkanker#:~:text=Borstkanker%20is%20de%20meest%20voorkomende,in%202021%3A%203059%20vrouwen).

Olamendy, J. C. (2024, 15 maart). The Power of Batch Size: A Comprehensive Guide to Gradient Descent. *Medium*. https://medium.com/@juanc.olamendy/the-power-of-batch-size-a-comprehensive-guide-to-gradient-descent-23f25a2183ff

Zvornicanin, E. (2024, 18 maart). *Accuracy vs AUC in Machine Learning* (G. Piwowarek, Red.). Baeldung. https://www.baeldung.com/cs/ml-accuracy-vs-auc#:~:text=Explainability%20and%20Intuition,analyze%20the%20prediction%20more%20deeply.