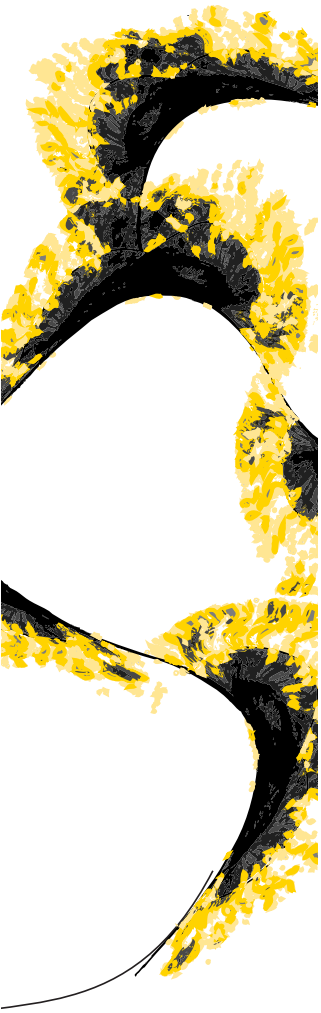


RAM ROBOTICS AND MECHATRONICS

A 3D DEEP LEARNING METHOD FOR THE PREDICTION OF BREAST TUMOR RESPONSE TO NEOADJUVANT CHEMOTHERAPY USING MR IMAGES WITHOUT THE NEED FOR A TUMOR SEGMENTATION

J. (Jaap) Lobeek

MSC ASSIGNMENT

Committee:
dr. ir. J.F. Broenink
dr. ir. M. Abayazid
dr. ir. L. Alic
dr. B. Dasht Bozorg

February, 2022

008RaM2022
Robotics and Mechatronics
EEMathCS
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

UNIVERSITY OF TWENTE. | TECHMED CENTRE UNIVERSITY OF TWENTE. | DIGITAL SOCIETY INSTITUTE

Summary

Introduction: Pathological complete response (pCR) is confirmed by the absence of residual tumor cells after pathological evaluation of the resected breast specimen after surgery. As about 46% of patients report any form of pain after surgery, and around 15% of the patients are still reporting moderate to severe pain 12 months after surgery, it would greatly improve the quality of life of these patients if pCR can be predicted and futile surgery can be left out in the treatment for these patients. For these reasons, it is desirable to predict pCR after neoadjuvant chemotherapy (NAC) in a non-invasive way.

Deep learning approaches have shown promising results to predict the response of breast tumors in breast cancer patients to NAC. However, these approaches need a manual tumor segmentation in the MR images. The segmentation is burdensome and takes much time since the radiologist needs to do this slice by slice. Especially after NAC, segmentation of the tumor is challenging due to unclear tumor boundaries.

In this study, it has been investigated to what extent it is possible to use a 3D CNN for predicting pCR without the need for tumor segmentation. For this, three sub-questions have been answered. First, the effect of the size of the region of interest (ROI) on the performance of the 3D CNN has been investigated. Second, it has been investigated which areas in the ROIs are most important for the prediction of the CNN. Lastly, it has been investigated if the usage of the molecular subtype can improve the performance of the CNN.

Methods: A large and a small ROI is drawn in the first post-contrast phase of the dynamic contrast-enhanced T1-weighted images, resulting in two datasets. Each ROI dataset is used to train a single-input 3D CNN on post-NAC MR images and a double-input 3D CNN on pre- and post-NAC MR images. Gradient-weighted Class Activation Mapping (Grad-CAM) has been used to visualize the most important regions in the MR images for predicting pCR. An additional input channel for the molecular subtype has been integrated into the single-and double-input CNN. Bayesian optimization has been used to find the optimal hyperparameters for all models above. The models are compared in mean accuracy, AUC, and mean Matthews correlation coefficient (MCC) after 10-fold cross-validation.

Results: The dataset contained 124 pCR patients and 55 non-pCR patients. The single-input CNN trained on the small ROI achieved the highest performance (a mean AUC, accuracy, and MCC of 0.74 ± 0.09, 69.90 ± 7.77, and 0.22 ± 0.25, respectively). No significant differences in the performance scores were found between training the CNNs with the large or small ROI dataset. Grad-CAM shows that the heart (visible in the large ROIs) and the nipple (visible in both ROIs), influence the prediction of the CNNs. Integration of the molecular subtype did not improve the performance of the CNNs.

Discussion and conclusion: It is possible to predict pCR with a 3D CNN without tumor segmentation with an AUC of 0.74 and mean MCC of 0.22. The main complication for all models was localizing the (former) tumor location in the MR images. Parts of the heart and the nipple influenced the performance of the prediction. To improve on the performance of the model, the main recommendations are to use smaller ROIs and to research which MR sequences to use as input for the model.

Table of content

List of abbreviations

1D	One-dimensional
2D	Two-dimensional
3D	Three-dimensional
AI	Artificial intelligence
AUC	Area under the curve
BCE	Binary cross-entropy
BCS	Breast-conserving surgery
CNN	Convolutional neural network
CR	Complete response
DCE	Dynamic contrast enhanced
DCIS	Ductal carcinoma in situ
DNA	Deoxyribonucleic acid
DWI	Diffusion-weighted imaging
ER	Estrogen receptor
FCL	Fully connected layer
GPU	Graphics processing unit
Grad-CAM	Gradient-weighted class activation mapping
HER2	Human epidermal growth factor receptor 2
IDC	Invasive ductal carcinoma
ILC	Invasive lobular carcinoma
LCIS	Lobular carcinoma in situ
MCC	Matthews correlation coefficient
MR	Magnetic resonance
MSE	Mean squared error
NAC	Neoadjuvant chemotherapy
NKI-AvL	The Netherlands Cancer Institute-Antoni van Leeuwenhoek Hospital
NN	Neural network
pCR	Pathological complete response
PR	Progesterone receptor
rCR	Radiologic complete response
RECIST	Response Evaluation Criteria in Solid Tumors
ReLU	Rectified linear unit
ROC	Receiver operating characteristic
ROI	Region of interest
TME	Tumor microenvironment
TN	Triple negative
VAB	Vacuum assisted biopsy

1 Introduction

1.1 Context

In The Netherlands, breast cancer is the cause of cancer with the highest prevalence and incidence in 2020, approximately 67,500 and 15,700, respectively (Sung et al., 2021). Over 50% of these breast cancer patients require chemotherapy as part of their treatment (Miller et al., 2019). The chemotherapy is increasingly administered before (neoadjuvant chemotherapy = NAC) instead of after breast surgery (adjuvant chemotherapy). Some benefits of NAC compared to adjuvant chemotherapy are the possibility of breast-conserving surgery (BCS) due to the downsizing of the tumor load and reducing the risk of recurrence (Thompson and Moulder-Thompson, 2012; Kaufmann et al., 2006).

If NAC is prescribed to a patient, magnetic resonance (MR) imaging is used to evaluate the effect of the therapy on the tumor (Mann et al., 2008; Liu et al., 2010). Multiple MR images are acquired during the therapy, one before the start of the chemotherapy (pre-NAC MR image), halfway through the course of the chemotherapy and the last MR image is acquired after the final course of chemotherapy (post-NAC MR image). Sometimes, also an MR scan is acquired after the first cycle of chemotherapy (Mann et al., 2008).

The effect of NAC can objectively be measured with the Response Evaluation Criteria In Solid Tumors (RECIST) criteria. With RECIST, a differentiation can be made between four groups: complete response (CR), partial response, progressive disease, and stable disease. CR means that all tumor cells have entirely disappeared after completion of NAC. For partial response, the tumor diameter needs to be shrunken at least 30%, and for progressive disease, the tumor diameter has increased at least 20%. All the other cases are categorized as stable disease(Eisenhauer et al., 2009).

Pathological complete response (pCR) is CR determined by pathological evaluation of the resected breast specimen after surgery. Patients who achieve pCR are associated with longer event-free survival and overall survival. (Kong et al., 2011; Cortazar et al., 2014; Prevos et al., 2012).

Another way to predict CR is by visually inspecting the pre-and/or post-NAC MR images. The absence of residual tumor mass on visual inspection of the MR images is called radiologic complete response (rCR) (Mann et al., 2008). Other studies have tried to use minimal invasive biopsy techniques for the pCR prediction (Heil et al., 2015, 2016). However, for both methods, the selection of patients with CR is unreliable (Prevos et al., 2012; Mann et al., 2008; Heil et al., 2015, 2016) meaning that surgery is still needed to remove the initial tumor bed and assess the specimen pathologically to confirm pCR.

As about 46% of patients report any form of pain after surgery (Wang et al., 2020), and around 15% of the patients are still reporting moderate to severe pain 12 months after surgery (Meretoja et al., 2014), it would greatly improve the quality of life of these these patients if pCR can be predicted and futile surgery can be left out in the treatment for these patients. For these reasons, it is desirable to predict pCR after NAC in a non-invasive way.

1.2 Problem statement

Using radiomics-based approaches have shown promise in the assessment of tumor response. More recently, deep learning approaches have emerged as promising tools for the prediction of the tumor response. Convolutional neural networks (CNNs) can discover visual patterns in images. These visual patterns can be used by the algorithm as imaging features for breast

cancer diagnosis, subtype classification, diagnosis for metastasis, or the prediction of pCR after NAC.

Some recently developed models to predict pCR after NAC are all using deep learning techniques (Braman et al., 2020; El Adoui et al., 2020; Qu et al., 2020; Ravichandran et al., 2018; Huynh et al., 2017; Ha et al., 2019). All studies need a manual segmentation of the tumor in the MR volume before making the response prediction. Five of these studies (Braman et al., 2020; El Adoui et al., 2020; Ravichandran et al., 2018; Huynh et al., 2017; Ha et al., 2019) use one 2D slice of the segmented tumor as input for the prediction model, only Qu et al. (2020) uses the 3D segmented tumor as input. All studies use one or multiple contrast phases from dynamic contrast-enhanced (DCE) T1 MR images as input.

One of the limitations of these state-of-the-art models is the usage of 2D slices of the MR images instead of the whole 3D volumes. If 2D slices instead of 3D volumes are used as input, the model can only learn from the information in one image. Possibly, there is more information along the third dimension of the MR volume. Also, a tumor may be absent in one slice, but it may still be present in the other slices of the 3D MR volume. So when the 2D slice of the 3D volume is wrongly determined, the tumor, and thus, important information might be missed.

Another limitation is that the manual segmentation is burdensome and takes much time since the radiologist needs to do the tumor segmentation slice by slice. Also, due to the adhesion of the tumor tissue to other tissues, the tumor boundaries are unclear and complicated to determine (Tian et al., 2021). Because of this, manual segmentation is prone to subjectivity and gives a lot of inter and intraobserver variability (Tian et al., 2021; Meyer-Baese and Schmid, 2014). Since these models are trained on images of segmented breast tumors, likely, the performance of these models is also influenced by the subjectivity and variability of the segmentation. Especially after NAC for some patients, the tumor boundaries are not visible. Segmentation or even localizing the tumor would then be very challenging.

Apart from being much quicker, an automatic tumor segmentation does not necessarily overcome these problems. Because due to the variability in the manual segmentation, it is hard to perceive correct and robust labels to train tumor segmentation algorithms (Tian et al., 2021).

By segmenting the tumor, the information in the surrounding healthy tissue of the tumor is removed. The healthy cells, molecules, and blood vessels surrounding and feeding into the tumor are called the tumor microenvironment (TME) (Whiteside, 2008). The TME and microvasculature in the area directly surrounding the tumor have much information on the effect of the chemotherapy on the tumor (Machireddy et al., 2019; Braman et al., 2017). So it seems that the information in the tissue surrounding the tumor is valuable for the response prediction of the tumor after NAC.

El Adoui et al. (2020) confirms this after comparing a CNN trained on segmented tumors and a CNN trained on images without segmented tumors. An model trained on non segmented 2D MR images (AUC = 0.91, sensitivity = 92.2, specificity = 79.1) outperforms an model trained on segmented 2D MR images (AUC = 0.74, sensitivity = 82.4, specificity = 68.7). This supports the statement that indeed the region surrounding the tumor has important information for the response prediction.

The different molecular subtypes of breast cancer respond differently to NAC. The molecular subtypes of breast cancer are based on the status of the receptors on the breast cancer cells. This means that the status of the receptors on the breast cancer cells may be another effective factor for predicting response to NAC (Rouzier et al., 2005; Subbiah et al., 2017). The receptors on breast cancer cells include estrogen receptor (ER), progesterone receptor (PR), human epidermal growth factor receptor 2 (HER2), and the nuclear antigen Ki-67. Qu et al. (2020) uses the ER, PR, and HER2 status as extra input for their model, improving the area under the receiver operating characteristic (ROC) curve (AUC) from 0.942 to 0.970. Ravichandran et al. (2018) also

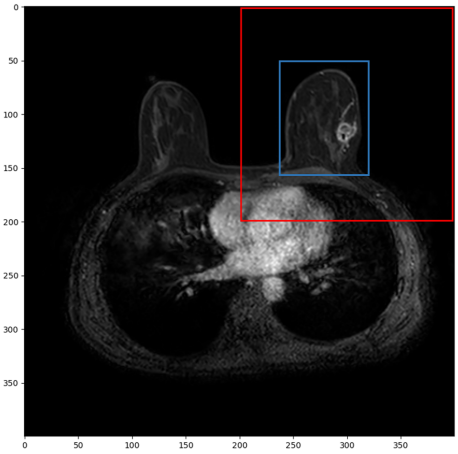


Figure 1.1: A slice of one of the pre-NAC MR images with an example of the large ROI (red rectangle) and the small ROI (blue rectangle).

investigated if there were significant age differences, lesion diameter, ER, PR and HER2 status between pCR and non-pCR patients. In this study, only adding the HER2 status improved the AUC of the model from 0.77 to 0.85.

1.3 Goal and research questions

To improve the assessment of NAC response prediction, the Image-Guided Surgery group in the Netherlands Cancer Institute-Antoni van Leeuwenhoek Hospital (NKI-AvL) wants to predict tumor response to NAC by analyzing pre-NAC and post-NAC MR images by the use of an automatic pipeline. To investigate if it is possible to use a 3D model for the prediction without the need for tumor segmentation in the MR volumes, the following main research question has been answered in this study:

To what extent is it possible to predict the response from breast tumors to neoadjuvant chemotherapy using a 3D multi-channel neural network and the pre-and post-NAC MR images without the need for tumor segmentation?

Since no tumor segmentation is performed, another way of determining a region of interest (ROI) is needed. Two possibilities determining the ROI have been conceived, resulting in a large and small ROI. The large ROI can be determined automatically by cropping one-quarter of the MR volume based on the affected breast, resulting in the red square in Figure 1.1. Possibly, an improvement in the performance of the model can be made by using a smaller ROI, containing only the breast tissue of each patient, as the blue square in Figure 1.1. The small ROI is determined manually for each patient.

To answer the main research question and to investigate the influence of the size of the ROI on the performance of the model, the following two sub-questions have been answered:

1. What is the effect of the size of the ROI on the performance of a 3D CNN for the prediction of the breast tumor responses to NAC?
2. Which areas in the ROI are most important for the model's prediction of the breast tumor response to NAC?

To try to improve the performance of the model and to investigate the influence of the integration of the receptor statuses on the performance of the model, a third sub-question has been answered:

3. To what extent can the integration of the ER, PR, and HER2 receptor status as extra input for the model improve the model's performance?

At the end of this study, it is expected to predict the response of breast tumors to NAC by using an automatic pipeline on pre-and post-NAC 3D MR volumes, without the need for tumor segmentation.

2 Background

2.1 Clinical background

Cell division is happening continuously in the human body and is important to replace worn-out cells and repair damaged tissue. It is estimated that in the order of 10^{16} cell division occur in a human body during a lifetime. One of the steps in cell division is making two identical copies of the deoxyribonucleic acid (DNA) molecule. This process is called DNA replication. Although very rare, mutations can be made in the DNA during DNA replication. Certain mutations can interrupt the normal cell division. The mutant cell can grow and divide (proliferate) out of control and form an expanding population of mutant cells. This mass of mutant unwanted cells is called a tumor. Benign tumors are encapsulated, localized and limited in size. Malignant tumors, however, are continuously growing and invade neighboring tissues (Parham, 2015).

When a malignant tumor is growing inside the breast, this is called breast cancer. The cells of the malignant tumor can also spread throughout the breast or to other organs via lymph or blood vessels. New tumor foci are called metastases. When the tumor or a metastasis is large enough, it can eventually disrupt the function of the affected organ and the body's physiology. Eventually, this leads to health problems and even death when not treated (Rubin and Reisner, 2014; Parham, 2015). Due to early detection and treatment, the risk of breast cancer death has significantly decreased in the Netherlands. Based on the breast cancer mortality rates of 2010, 1 in 27 women will die from breast cancer during their lifetime, based on the mortality rates in 2000, this was still 1 in 24 women (Van Der Waal et al., 2015).

Breast cancer can be classified into different kinds of sub-types based on histological features. A further classification of breast cancer types can be made based on the molecular biology of breast cancer (Kumar and Clark, 2012).

2.1.1 Breast cancer sub-types

The breast is primarily composed of glandular tissue. Glandular tissue is the functional tissue of the breast that produces the milk. This includes the lobes, that produce the milk and the ducts, that transport the milk to the nipple. Between all the lobes and ducts is adipose tissue (fat tissue) distributed (Figure 2.1). The adipose tissue makes up for most of the breast volume, around 50-70% (Gabriel and Maxwell, 2020).

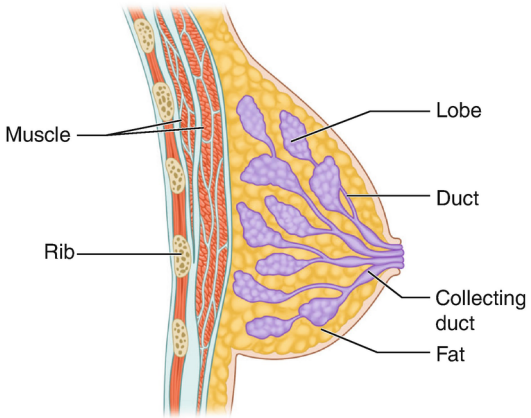
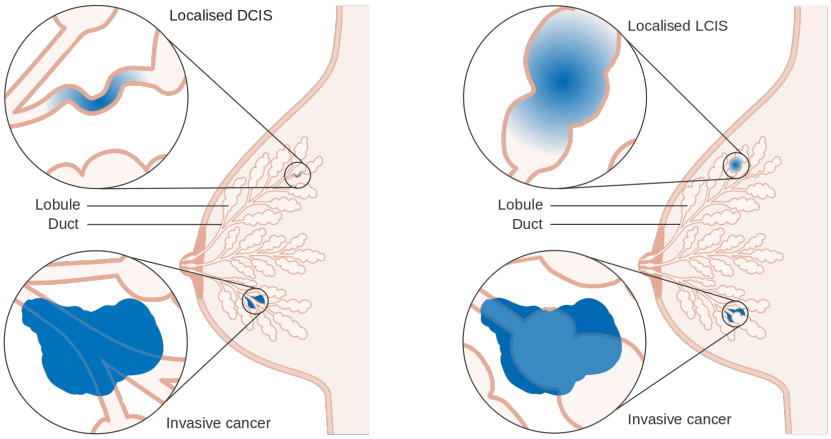


Figure 2.1: Anatomy of the female breast. The breast is mostly composed out of fat, milk producing lobules and milk ducts. Image from: Gabriel and Maxwell (2020)



(a) Schematic representation of DCIS and IDC, from: Breast Cancer Foundation NZ (2021)

(b) Schematic representation of LCIS and ILC, from: Irish Cancer Society (2021)

Figure 2.2: Schematic representation of ductal carcinoma in situ (DCIS) and invasive ductal carcinoma (IDC) (**a**) and lobular carcinoma in situ (LCIS) and invasive lobular carcinoma (ILC) (**b**). The blue coloring is a schematic representation of the cancer cells.

Most breast cancers originate from cells from the glandular tissue, either from the milk-producing lobes or the ducts. Accordingly, most breast cancers are either lobular carcinoma (originating from the lobes) or ductal carcinoma (originating from the ducts). Further classification can be made on if the malignant cells have infiltrated through the membrane of the tissue, into neighboring breast tissues (invasive) or if the tumor is still bound by the membrane of the tissue where the tumor originated (in situ). Based on this information a discrimination between four breast tumors can be made: ductal carcinoma in situ (DCIS), lobular carcinoma in situ (LCIS), invasive ductal carcinoma (IDC), and invasive lobular carcinoma (ILC) (Figure 2.2). More variants of invasive and in situ breast cancers are known, such as Paget disease, encapsulated papillary carcinoma, or tubular carcinoma), but these variants are rare and the first four mentioned breast cancer variants account for most of the breast cancers (Rubin and Reisner, 2014).

A further sub-division of breast cancers can be made based on the molecular subtype of breast cancer. The molecular subtype is based on the expression of the ER, PR, and HER2 on the cell (Rubin and Reisner, 2014; Gomes Do Nascimento and Otoni, 2020). Also the cell proliferation regulator (Ki-67), a nuclear protein, is used for the subdivision of molecular subtypes of breast cancer. Ki-67 is associated with tumor cell proliferation and growth and correlates with metastasis (Gomes Do Nascimento and Otoni, 2020; Li et al., 2015). Based on the gene expression patterns in breast cancer, four molecular subgroups can be made: Hormone receptor (ER or PR) positive tumors, called Luminal A and Luminal B, enriched HER2 (HER2+), and tumors lacking all 3 standard molecular markers, called triple negative (TN). Each molecular subtype has its own gene expressions and individual disease prognosis, see Table 2.1 (Rubin and Reisner, 2014; Gomes Do Nascimento and Otoni, 2020).

The molecular classification makes breast cancer a heterogeneous collection of diseases. Each molecular subtype of breast cancer has its own prognosis, treatment, and responses to treatment (Rubin and Reisner, 2014; Gomes Do Nascimento and Otoni, 2020). The different molecular subtypes of breast cancer also can have different responses to NAC (Rouzier et al., 2005; Subbiah et al., 2017). This means that the information on the molecular subtype can possibly improve the performance of the model for NAC response prediction.

Table 2.1: Classification and prognosis of molecular subtypes of breast cancer

	Luminal A	Luminal B		HER2+	TN
		HER2-	HER2+		
ER	+	+	+	-	-
PR	+	-	-/+	-	-
HER2	-	-	+	+	-
Ki67	Low	High	Low/high	High	High
Prognosis	Good	Intermediate		Poor	Poor

+ = positive, - = negative, -/+ = positive or negative
Note: adapted from Gomes Do Nascimento and Otoni (2020)

2.1.2 Breast cancer treatment

The choice of breast cancer treatment is determined by, among others, tumor size, lymph node status, distant metastases (TNM classification), and molecular classification (Kumar and Clark, 2012). For non-metastatic breast cancer and metastatic breast cancer, the goal of the treatment is different (Waks and Winer, 2019).

For non-metastatic breast cancer, the goal of the treatment is to remove the tumor from the breast and regional lymph nodes and prevent a (metastatic) recurrence. This can be done by local therapy, consisting of surgical resection and removal of the lymph nodes. Surgery can vary from BCS for masses smaller than 3 cm in diameter or mastectomy (complete removal of the breast) with or without reconstruction (Kumar and Clark, 2012). Additionally, postoperative radiation can be given. Additionally to local therapy, systemic therapy can be given, this can be preoperative (neoadjuvant), postoperative (adjuvant), or both. The molecular subtype of breast cancer determines which systemic therapy is given. (Waks and Winer, 2019)

For metastatic breast cancer, the goal of the treatment is to extend life and symptom palliation. Local therapy (surgery and radiation) is only used for palliation purposes. For systemic treatment, the same treatments are used in metastatic breast cancer as in non-metastatic breast cancer (Waks and Winer, 2019).

Since this project is concerned with the prediction of the response of breast cancer patients to NAC, only this treatment will be described in the next section. The other systemic and local treatments are not explained further.

Neo-adjuvant chemotherapy

Chemotherapy is an essential treatment for preventing recurrence in many patients with breast cancer and can provide good-quality palliation and prolongation of life (Waks and Winer, 2019; Kumar and Clark, 2012). It is also the only effective systemic treatment for TN breast cancer and is an important extra treatment for endocrine therapy or HER2-targeted therapy in patients with hormone receptor-positive or HER2+ breast cancer (Waks and Winer, 2019).

A lot of different treatment regimens exist for NAC. A gross division between these regimens can be made based on chemotherapy benefits and severeness of toxicities. Some chemotherapy regimens (like docetaxel/cyclophosphamide, adriamycin/cyclophosphamide, and cyclophosphamide/methotrexate/5-fluorouracil) are reasonable choices for lower-risk patients, where chemotherapy benefits are less important and severeness of the toxicities can weigh more in the decision making. Some chemotherapy regimens (anthracycline and taxane-containing regimens) achieve the highest risk reduction in recurrence from the tumor, but also have higher severeness of toxicities. These regimens are best for high-risk patients (Waks and Winer, 2019).

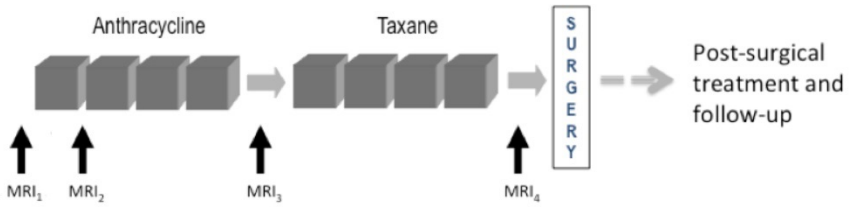


Figure 2.3: Example of a NAC schedule. Image adapted from: Newitt and Hylton (2016)

Neoadjuvant chemotherapy is chemotherapy given before surgery to make inoperable tumors operable or make large tumors smaller to provide BCS instead of mastectomy. NAC additionally reduces the risk of death from distant metastases (Kumar and Clark, 2012). The optimal outcome of NAC is the complete removal of the tumor in the breast and lymph nodes (Montemurro et al., 2020).

Figure 2.3 is an example of a NAC schedule. Each gray block in the figure represents one cycle of chemotherapy. After finishing the chemotherapy, surgery is following to remove the last remaining tumor lump or to confirm pCR. Most current chemotherapy regimens consist of anthracyclines and taxanes given either sequentially for up to 8 cycles, like in Figure 2.3. The time between the cycles of current commonly most used regimens is one up to three weeks (Rapoport et al., 2014).

MRI is used to monitor the response of the tumor to NAC (Mann et al., 2008; Prevos et al., 2012). A pre-NAC MR image is made before the start of NAC (MRI_1 in Figure 2.3). After NAC and before surgery a post-NAC MR image is made (MRI_4 in Figure 2.3). Often also an MR image after the first cycle of chemotherapy and in the middle (after four cycles of chemotherapy) is made ((MRI_2 and MRI_3 in Figure 2.3). For this project, the pre- and post-NAC MR images are used to develop the prediction model.

2.1.3 rCR and biopsy techniques

As already discussed shortly in the introduction, another way to determine if there is presence of residual tumor is rCR. This is defined as the absence of tumor on a MR image by visually inspecting this image. Although, patients achieving rCR are strongly associated with recurrence-free surival and overal survival, rCR can not predict pCR (Gampenrieder et al., 2019). According to the research of Gampenrieder et al. (2019), using rCR to predict pCR had a sensitivity of 75% and a corresponding specificity of 67%. These scores are not high enough to replace the pathology examination. In many cases small tumor foci could not be detected on MRI.

The low sensitivity and specificity for rCR in predicting pCR can be contributed to the MRI missing small tumor foci with regular vascularization (Gampenrieder et al., 2019). Also, missing the tumor on the MRI can be caused by reactive changes due to NAC, like inflammatory response or fibrotic reaction in the tumor bed (Woo et al., 2020). Inflammatory response or fibrotic reaction in the tumor bed can result in a strong decrease of contrast enhancement by the tumor and thus result in less visible on the MRI (Wasser et al., 2003).

Other studies tried to use minimal invasive biopsy techniques to diagnose pCR after NAC (Heil et al., 2015, 2016). Ultrasound-guided vacuum assisted minimal invasive biopsy (VAB) was performed on patients with rCR to diagnose pCR. Heil et al. (2015) reached a sensitivity of 50.7% and corresponding specificity of 93.5% to predict pCR. Although this looks like a small improvement on the predictive value of rCR, but are still too low for replacing the surgery and pathology examination. The high specificity and low sensitivity also means that it is more useful for ruling out the patients who don't have pCR, instead of determining which patients have pCR.

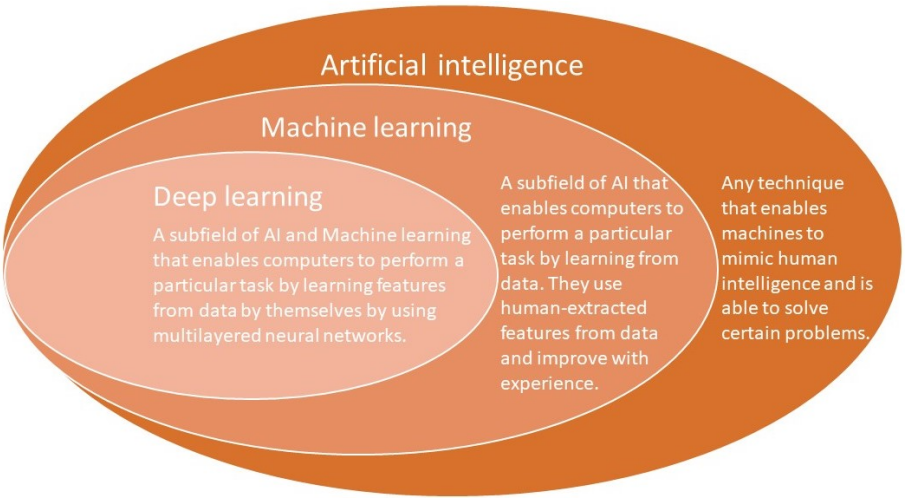


Figure 2.4: Venn diagram of the field of AI. It shows that deep learning is a subfield of machinelearning, which is a subfield of AI. Each subfield includes a small explanation of that subfield. Adapted from: Goodfellow et al. (2016).

The low sensitivity in the VAB for predicting pCR can be attributed to the high change of VAB missing the (former) tumor region due to suboptimal conditions in the surgical theatre (no high-end ultrasound machine, time pressure) and clip markers which are not always visible in ultrasound (Heil et al., 2016). The performance of a deep learning model for the prediction of pCR can possibly outperform the performances of rCR and VAB for this prediction. Also, using a deep learning model for the prediction would be not invasive, compared to minimal invasive with using VAB.

2.2 Technical background

We always say that we are the most intelligent species on earth. We have always dreamed to replicate our own intelligence and making a machine that can think. Artificial intelligence (AI) is the field of developing these intelligent machines. Today, this field covers a lot of practical applications and research topics. (McCorduck, 2004; Goodfellow et al., 2016)

The first working electro-mechanical computer was about 2.1 by 1.8 by 0.6 meters and weighed around 1000 kg. It was called The Bombe and was developed in 1940 by the English mathematician Alan Turing. During the Second World war, this machine could crack the Enigma code used by the German army. Before The Bombe, this code was almost impossible to break by the best human mathematicians and this made Turing think about the intelligence of these machines. This made him write his article 'Computing Machinery and Intelligence' in 1950. This article describes how to test if a machine can be considered intelligent or not. This test is called the Turing test. (McCorduck, 2004; Haenlein and Kaplan, 2019)

The first computers solved problems that can be described and solved by a list of formal and mathematical rules. Today, due to the development of AI, the problems that computers can solve are more and more problems that can not be solved by a list of rules but need some sort of previous knowledge. Among others, speech recognition, recognize faces or objects from images, credit card fraud detection, text translation, and making diagnoses from medical images or other medical data. (Goodfellow et al., 2016; Haenlein and Kaplan, 2019)

The field of AI can nowadays be divided into different subfields with every subfield its different approaches. Machine learning is one of the approaches to AI. Within the domain of Machine learning, the machine is trained by using large amounts of data and algorithms that give it the ability to learn how to perform a task, instead of coding a specific list of mathematical instruc-

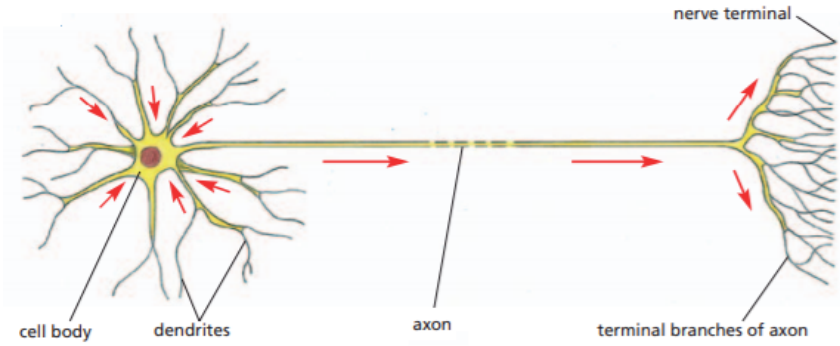


Figure 2.5: Schematic figure of a single neuron. Adapted from (Alberts et al., 2014)

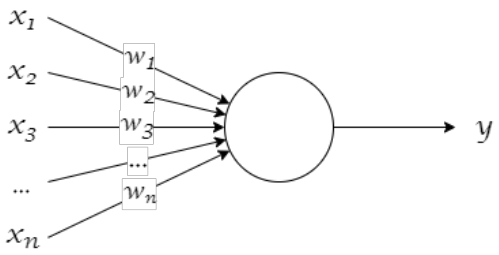


Figure 2.6: Schematic figure of a perceptron. Each input x_i is multiplied with a weight w_i and all these values are summed. If this summation exceeds a certain threshold, the perceptron will give an output of 1. Adapted from: (Nielsen, 2015)

tions. Machine learning approaches are, for instance, random forest, k-nearest neighbor, Naive Bayes, and linear regression.

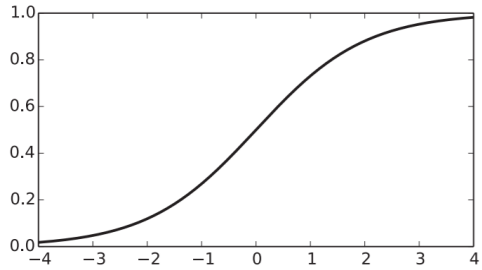
Deep learning is a domain within the domain of machine learning (see Figure 2.4). Deep learning methods are methods with multiple levels of representation of data. These representations are obtained by composing simple but non-linear transformations on the data. This is done by the use of multilayered neural networks (NN). The key aspect of deep learning is that all the different representations of data are not designed by human engineers, but they are learned from the data by the algorithm itself (LeCun et al., 2015; Goodfellow et al., 2016).

A NN is composed of multiple perceptrons stacked in multiple layers. A perceptron is the mathematical model of a neuron. A biological neuron typically has many inputs (dendrites), a cell body with a nucleus, and a single output (the axon), see Figure 2.5. The axon divides into many branches so that the neuron's message can be passed simultaneously to many other cells (Alberts et al., 2014; Goodfellow et al., 2016; Charniak, 2018; Nielsen, 2015).

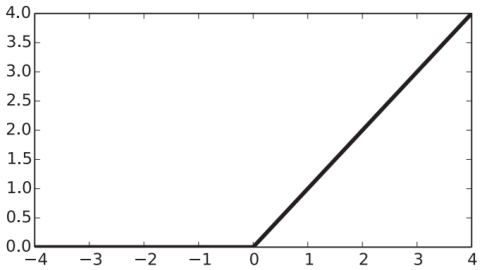
A perceptron is based on the same concept, it also has multiple inputs and a single output, see Figure 2.6. All inputs $\mathbf{x} = [x_1, x_2, ...x_n]$ are each multiplied with a weight. We can write these weights as a vector $\mathbf{w} = [w_1, w_2, ...w_n]$. After multiplicating each input with the weight, these values get summed. If the subsequent value exceeds a certain threshold θ, the perceptron returns one, otherwise, it returns zero (Equation 2.1)(Goodfellow et al., 2016; Charniak, 2018; Nielsen, 2015).

$$y = \begin{cases} 1, & \text{if } \sum_{j=1}^{n} w_j x_j > \theta \\ 0, & \text{if } \sum_{j=1}^{n} w_j x_j \leq \theta \end{cases} \tag{2.1}$$

We can simplify Equation 2.1 by subtracting both sides of the equation with θ and replacing it by the perceptron's bias b ($b = -\theta$). We can also write $\sum_j w_j x_j$ as the dot product of the



(a) The graph of a sigmoid function. From: Charniak (2018)



(b) The graph of a rectified linear unit (ReLU). From: Charniak (2018)

Figure 2.7: The graphs of two activation function examples.

two vectors **w** and **x**. We can write the equation as a function of **x**, with **x** the input of the perceptron and **w** and b the parameters for the perceptron. (Goodfellow et al., 2016; Charniak, 2018; Nielsen, 2015)

$$f_{\mathbf{w},b}(\mathbf{x}) = \begin{cases} 1, & \text{if } \mathbf{w}\cdot\mathbf{x}+b>0 \\ 0, & \text{if } \mathbf{w}\cdot\mathbf{x}+b\leq 0 \end{cases} \tag{2.2}$$

A NN is just a mathematical function mapping some input values to output values. This function is composed out of many much simpler functions, the mathematical function for the perceptron. A NN can be made deeper by just stacking more layers of perceptrons on each other, although there is no consensus on how much depth a NN requires to be qualified as deep. (Goodfellow et al., 2016; Charniak, 2018; Nielsen, 2015)

With training a NN, we change the weights and biases in the NN a little bit every time, so that the output of the network gets closer to the desired output. How the training and changing of the weights and biases exactly is done, will be explained further in the text.

Since we have only two possible outputs for a perceptron (a one or a zero), a small change in the weights and biases of a single perceptron in the NN can cause a large change in the output of the NN. This makes it very difficult to see how the weights and biases need to be modified to get the desired output. To overcome this problem a NN uses an activation function, like the sigmoid function, which is defined by Equation 2.3 with the corresponding graph in Figure 2.7a.

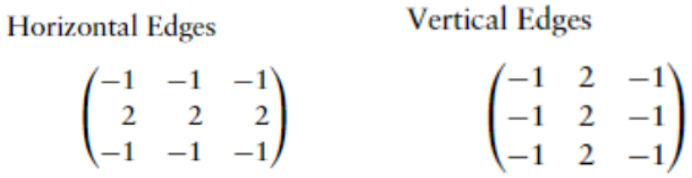
Due to the shape of the sigmoid function, it is possible to have small changes in the weights and biases and have small changes in the output from the neuron. Other activation functions are used often, like the rectified linear unit (ReLU), see Figure 2.7b.

$$\sigma(\mathbf{x},\mathbf{w},b) = \frac{1}{1+\exp(-\mathbf{w}\cdot\mathbf{x}-b)} \tag{2.3}$$

There are many NN architectures of which the fully connected feedforward NN is the most basic one. Many different architectures have been proposed for specific tasks. A specialized NN architecture for computer vision is called a convolutional neural network (CNN), these networks are partially connected NNs. A unit in a partially connected NN only feeds its output to some of the next layer's units. (Goodfellow et al., 2016) Since a CNN is the type of network that is used in this project, this network will be further explained in the next chapter.

2.2.1 Convolutional neural networks

When feeding data into an (ordinary) feed-forward fully connected NN, the network is looking at the absolute values of the data and making its decisions or solving its designated problem



(a) Convolution kernel for detecting horizontal edges

(b) Convolution kernel for detecting vertical edges

Figure 2.8: Two examples for convolution kernels for detecting horizontal and vertical edges. Adapted from McReynolds and Blythe (2005)

based on these absolute values. But sometimes you want to look at the differences between the values of the data points, rather than the absolute values of these data points.

For instance, to recognize an object in a picture, the relation of the light intensity (the pixel value) on one location to its neighboring locations is important. In other words: the differences in the pixel value of a certain pixel to its neighboring pixels. This is what CNN's are particularly good at, detecting local light intensity differences in images. But this doesn't need to be particularly images, a 2D grid of pixels. This can also be a 1D grid, for instance, time-series data, which are a grid of 1D samples at regular time intervals. (Goodfellow et al., 2016; Charniak, 2018)

Convolutional layers, nonlinearity and pooling layers

To find these differences between (local) pixel values a CNN makes use of a mathematical operation called convolution. The convolution is an integral that expresses the amount of overlap of one function g as it is shifted over another function f. We typically write a convolution between two functions with an asterisk:

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t-\tau)d\tau \tag{2.4}$$

In image terminology one can think about this as shifting a 2D filter kernel $K(i, j)$ over a 2D image $I(i, j)$ to get a feature map $S(i, j)$. Since the image and filter kernel are discretized data (an image is nothing more than data points at regular intervals in space), we can write the discrete convolution as a summation.

Also, Since both the image and filter kernel have a finite size and we assume that $K(i, j)$ and $I(i, j)$ are zero everywhere except within their boundaries, we can replace the infinite summation as a summation over a finite number of array elements. Lastly, the image and filter kernel are both 2D arrays instead of a 1D array as in Equation 2.4. These three arguments changes Equation 2.4 into Equation 2.5. (Goodfellow et al., 2016; Charniak, 2018; McReynolds and Blythe, 2005)

$$S(i, j) = (I * K)(i, j) = \sum_m \sum_n I(m, n)K(i-m, j-n) \tag{2.5}$$

What this equation does, is moving the filter over the image and computing an output. The output is saying something about how much different parts of the image corresponds to the filter kernel. The higher the value of the output, the more that part of the image looks like the filter kernel. With convolution kernels, you can detect certain structures in an image, for instance, horizontal or vertical lines (Figure 2.9) or edges.

Normally a filter kernel is designed by the programmer to detect certain features within an image. Within a CNN there are multiply convolutional layers with a lot of different filter kernels,

(a) Original image

(b) After convolution with an edge detection kernel

Figure 2.9: Example of an image after convoluting it with an edge detection kernel. Image (**a**) shows the original image and image (**b**) shows the convoluted image. In the convoluted image, the highlighted parts are parts where edges are present in the original image (**a**). Adapted from McReynolds and Blythe (2005)

but what kind of feature is picked out from the image is not designed by the programmer, it is learned by training the CNN. Figure 2.10 is an example of convolution kernels learned by the first layer of a CNN. What can be seen is that different kinds of edge detectors and colors are being learned to be detected by the CNN. The filter kernel $K(i, j)$ are the weights $\mathbf{w}$, talked about earlier, that are being trained by the network. The resulting feature map after the convolutional layer is then passed through a non-linearity function, such as sigmoid function or ReLu function to introduce non-linearity into the network. (Charniak, 2018; LeCun et al., 2015)

The role of the pooling layer is to reduce the dimensions of the feature maps. This provides two advantages. First, it reduces the number of parameters to learn and thus number of computations to be made. Second, a convolutional layer also detects the precise location of the different features. Due to pooling multiple pixels into one pixel, precisely positioned features are summarized, this makes the the network more robust for small translations of the features in the image. Pooling can be done in different ways, by taking the maximum pixel value of a small neighborhood of pixels (max pooling), or by taking the average of a small neighborhood of pixels, called average pooling. (Goodfellow et al., 2016; LeCun et al., 2015)

A CNN consists of stacks of convolutional layers, non-linearity functions (like a sigmoid or ReLU function), and (max) pooling layers. Making the network deeper can be done by stacking more of these layers. This does not have to be necessarily in this order, but typically one or two or three convolutional layer(s) is/are followed by a non-linearity function and a pooling layer. (Goodfellow et al., 2016; LeCun et al., 2015)

2.2.2 Training a network

There are multiple approaches to train a deep learning model. In this project supervised learning is used to train the CNN. Supervised learning is a form of learning where each image is labeled with its class. During training, the CNN is presented with an image and the CNN is producing an initial output. The output is in the form of a vector (with the length of the number of classes) with scores, a score of the likelihood of the input image being that class. An error is calculated between the initial output of the CNN (the vector of scores for each class) and the desired scores (the label of the image). The CNN then adjusts the internal parameters, the weights $\mathbf{w}$ and biases b to reduce the error. The function that calculates the error is called the loss function. Sometimes this function is also called objective function or cost function. (LeCun et al., 2015; Goodfellow et al., 2016; Nielsen, 2015)

Figure 2.10: Example of convolution kernels learned by the first layer of a CNN. Adapted from: Goodfellow et al. (2016)

A regularly used loss function is the mean squared error:

$$L(\mathbf{w},\mathbf{b}) = \frac{1}{2n}\sum_{x}||y(x) - y'(x,\mathbf{w},\mathbf{b})||^2 \tag{2.6}$$

Here, $\mathbf{w}$ is the collection of all the weights of the network, $\mathbf{b}$ of all the biases, n the number of training inputs, $y(x)$ the labels of all the inputs and $y'(x,\mathbf{w},\mathbf{b})$ the prediction of the network. The loss becomes equal to zero when $y(x)$ becomes equal to $y'(x,\mathbf{w},\mathbf{b})$. If the outcome is not equal to the label, the training algorithm adjusts weights and biases in the network to reduce the loss function. The goal of the training is to find the weights and biases which make the loss function as small as possible. To do that, the training algorithm computes a gradient vector. (Nielsen, 2015; LeCun et al., 2015)

The gradient vector ∇L indicates for each weight and bias by what amount the loss would change (and in what direction) if this weight or bias would be changed by a small amount. We have a gradient vector for the weights and a gradient vector for the biases:

$$\nabla L_w = \left(\frac{\partial L}{\partial w_{1,1}}, \frac{\partial L}{\partial w_{1,2}}, ..., \frac{\partial L}{\partial w_{m,n}}\right) \tag{2.7}$$

$$\nabla L_b = \left(\frac{\partial L}{\partial b_1}, \frac{\partial L}{\partial b_2}, ..., \frac{\partial L}{\partial b_m}\right) \tag{2.8}$$

The weight vector and bias vector are adjusted in the opposite direction to the gradient vector.

$$\delta w_{i,j} = -\eta\frac{\partial L_w}{\partial w_{i,j}} \tag{2.9}$$

$$\delta b_i = -\eta\frac{\partial L_b}{\partial b_i} \tag{2.10}$$

Here, η is the learning rate. The learning rate is an important parameter, which needs to be chosen correctly. The averaged loss function over all the training inputs can be seen as a wavy plane. In this multi-dimensional plane (set up by all the weights and biases), we want to find the minimum value for the loss function. The negative gradient vector is pointing into the direction of the steepest decline. By every update of the weights and biases according to the gradient descent, we will hopefully find the minimum of the loss function. When the loss function is dependent on only two variables, it is easy to visualize the loss function, this can be seen in

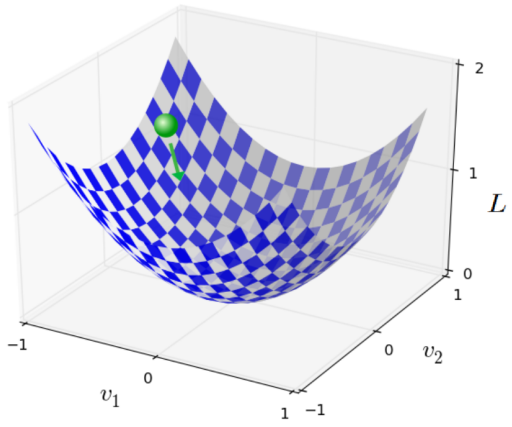


Figure 2.11: Visualisation of a two-dimensional loss function $L(v_1, v_2)$. The green dot is the position of the loss, and the gradient vector (the green arrow) points into the direction of the steepest descent, taking it closer to the minimum. Adapted from: Nielsen (2015)

Figure 2.11. In reality, the loss function is a complex multidimensional function (Nielsen, 2015; LeCun et al., 2015; Charniak, 2018)

The gradient vector is calculated for every training sample and shows how we would change each parameter to minimize the loss. The parameters are not actually changed until for each sample, a gradient vector is calculated. Each parameter gets modified by the sum of the changes from each individual sample. The problem with this is that it can be very slow, especially when the training set is very large. To coop with this a stochastic gradient descent is used. The stochastic gradient descent updates the parameters for every β samples. The size of β is much smaller than the size of the training set and is called the batch size. (Goodfellow et al., 2016; Charniak, 2018; Nielsen, 2015)

When a batch size β is small, the learning rate η needs to be smaller as well. When for instance a batch size of one is chosen, the gradient descent is going to change the weights and biases to classify this sample correctly at the expense of others. If the learning rate is low, this effect is smaller, since the parameters change with a very small value. With a large batch size, a higher learning rate can be chosen, since the change of parameters is averaged over a larger batch size, so the danger of pushing the parameters to a certain direction to satisfy one sample is smaller. (Goodfellow et al., 2016; Charniak, 2018; Nielsen, 2015)

The process where you use Equation 2.6 and calculate the loss for a mini-batch, is called the forward pass. You are going from input to the output from the network and calculating the loss, forward in the network. The process where you use Equation 2.8, Equation 2.7, Equation 2.9 and Equation 2.10 is called the backward pass or backpropagation. The just calculated loss is 'going backward' into the network and calculates how the weights and biases should change in order to minimize the loss.

Commonly, for training a model, three sets of data are used. One training set, for doing the training and updating the weights and biases and another smaller validation set, to calculate the accuracy after each epoch of training. A third set is used after completing the training to determine the accuracy (or other performance scores) on samples which are never seen before by the network. This third set can also be used to find the hyperparameters of the network. These are parameters other than the weights and biases, which can not be learned by the network through training. The following four steps summarize the algorithm of training a network. (Charniak, 2018; Goodfellow et al., 2016; Nielsen, 2015)

1. For j from 0 to m Set b_j randomly, but close to zero

2. For j from 0 to m and i from 0 to n set $w_{i,j}$ randomly, but close to zero
3. Until validation loss stops decreasing
   (a) for each training sample in the mini batch of size β
      i. Do the forward pass and calculate the loss using Equation 2.6
      ii. Do the backward pass and calculate the weight updates and bias updates using Equation 2.8, Equation 2.7, Equation 2.9 and Equation 2.10
      iii. Every mini batch, after β samples, modify the weights and biases using the summed updates
   (b) Compute the validation loss and validation accuracy of the model by doing a forward pass of all the samples in the validation set
4. Save all the weights and biases from the epoch scoring the lowest loss on the validation set.
5. Use the samples from the test set as input for the model. Calculate the accuracy and/or other performance scores for the

3 State-of-the-art response prediction methods

State-of-the-art literature can be divided into literature that describes deep learning models and conventional machine learning models other than deep learning. An analysis of the state-of-the-art models on the second category can be found in Appendix A. The rest of this section focuses on the state-of-the-art literature on deep learning models for the prediction of NAC response.

In Table A.2 the extracted data from the reviewed deep learning techniques for breast cancer response prediction can be found. During the literature review, a total of six studies on deep learning models for the response prediction have been found (Braman et al., 2020; El Adoui et al., 2020; Qu et al., 2020; Ravichandran et al., 2018; Huynh et al., 2017; Ha et al., 2019). These studies are all retrospective studies.

Braman et al. (2020) developed a double-input CNN (Figure 3.1) for the prediction of pCR. Two contrast phases from the DCE-T1 MR images are used. An ROI is placed around the tumor in the slice with the largest tumor area. This is done for both the pre-and post-contrast phases and both ROIs are used separately as input to their convolutional branch. The features from each convolutional branch are then combined into a dense layer to give the relationship between these features and to make a response prediction.

All possible combinations for the pre-and post-contrast phases (pre-contrast phase and first to third post-contrast phase) were used as input for the CNN. Using the pre-contrast phase and the third post-contrast phase gave the best AUC of 0.85 (see Table A.2). This score has been obtained on a test set of 28 patients.

In El Adoui et al. (2020) a single-input CNN (Figure 3.2) and a double-input CNN (Figure 3.3) has been developed. A total of 723 axial slices from 42 patients are used to train the model. The first post-contrast phase of the DCE T1 sequence of the pre-NAC MR images and the post-NAC MR images are used as input. The tumor is manually segmented and zero-padded to an input size of 120 × 120. Using segmented tumors from only the post-NAC MR images for the single-input CNN resulted in an accuracy of 0.69 and an AUC of 0.71. The double-input CNN trained on the segmented tumors from both the pre-and post-NAC MR images achieved an accuracy of 0.70 and an AUC of 0.74.

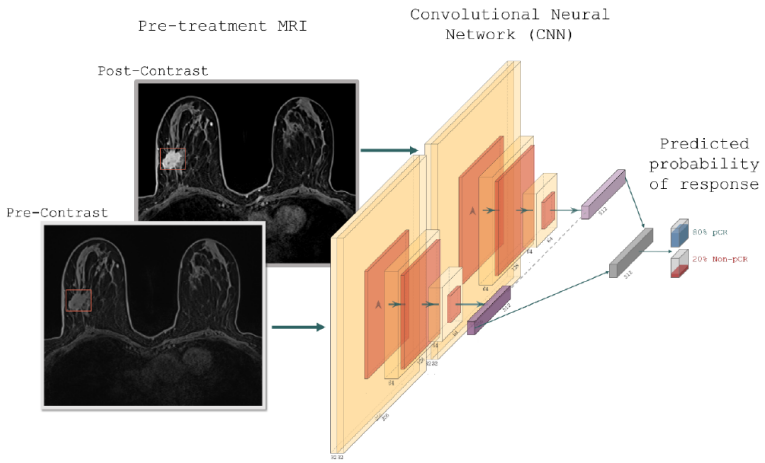


Figure 3.1: The double-input network from Braman et al. (2020)

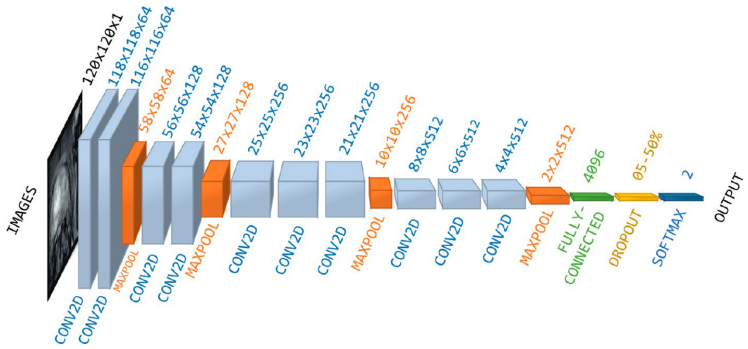


Figure 3.2: The single-input network from El Adoui et al. (2020)

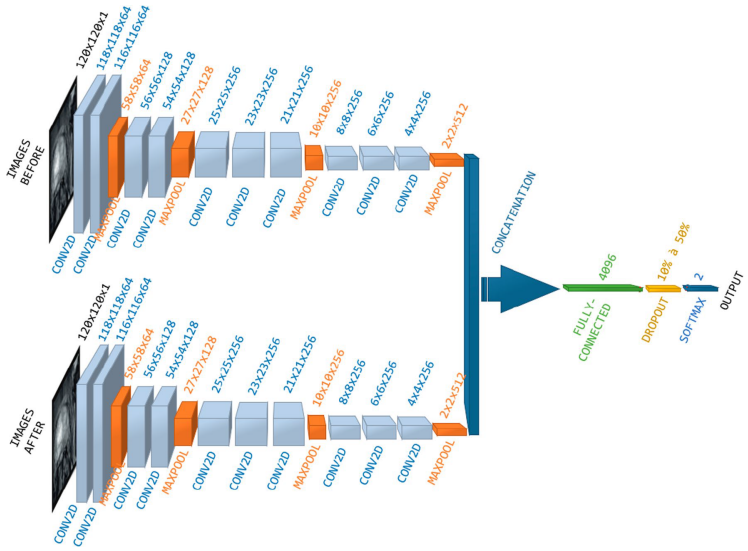


Figure 3.3: The double-input network from El Adoui et al. (2020)

To investigate the role of the TME around the tumor for the prediction of pCR in breast cancer patients, the CNN is also trained on a second dataset. Instead of segmenting the tumor, the MR images are cropped to a size of 120 × 120 by manually drawing an ROI around the tumor. The CNN trained on the dataset without segmentation achieved the best results with an accuracy of 0.88 and an AUC of 0.91 for the double-input CNN and an accuracy of 0.80 and an AUC of 0.79 for the single-input CNN.

In this study, a gradient-based localization evaluation is performed to visualize the tumor regions providing the most important features to classify pCR and non-pCR patients. It turned out that for non-pCR patients, the CNN emphasized its surrounding region in the DCE-MRI images. For pCR patients, it focuses on all the tumor regions. This means that the surrounding tissue of the tumor can play an important role in the classification of pCR patients.

Qu et al. (2020) developed a multi-input CNN for the classification of pCR patients (Figure 3.4). This CNN has 12 different input channels, one for each of the 6 phases of the pre-NAC MRI and post-NAC MRI. Each input channel has five repetitions of convolution and max-pooling layers to detect the features. Three indices for the status of the ER, PR, and HER2 receptor are used as extra input for the dense layers. The tumor was segmented manually on each slice of the tumor and zero-padded to a size of 128 × 128 × 16. The trained CNN reached an AUC of 0.970 on a validation set of 58 patients.

Ravichandran et al. (2018) used a CNN consisting of 6 blocks with convolutional layers with batch normalization and ReLu activation followed by one fully-connected layer. The tumor was

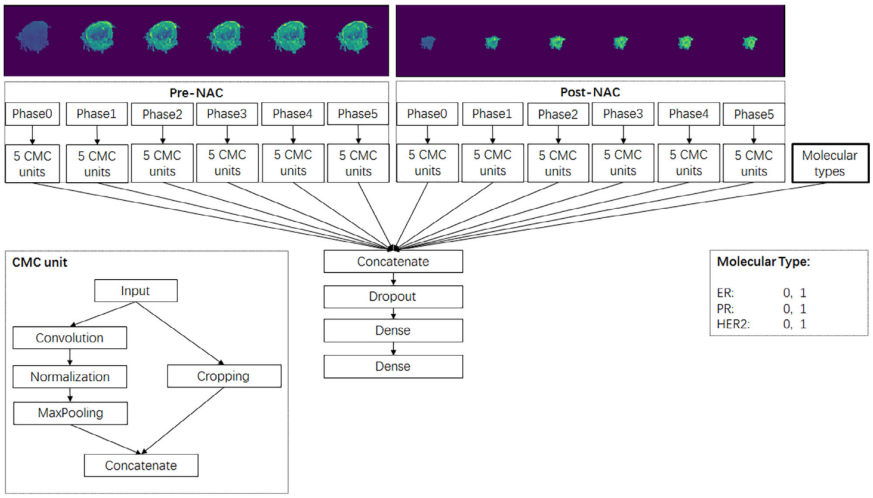


Figure 3.4: The multi-input CNN from Qu et al. (2020)

segmented by thresholding semi-quantitative pharmacokinetic parameters and was further refined by morphological operations. From the tumor mask, patches of 64 × 65 were randomly selected to use as input for the CNN. Both pre-and post-contrast images are used as input for the CNN. The model gave each pixel in the tumor a probability of response. Majority voting was used to determine predictions for a given patient.

Integrating the HER2 status in the model improved the model from an accuracy of 0.82 and AUC of 0.77 to an accuracy of 0.85 and AUC of 0.85. The pixel-wise probabilities were used to overlay the original MRI images as a heat map, showing the response likelihood of different areas of the tumor. The heat maps suggested that the center of the tumor is highly predictive in response to NAC compared to the sides of the tumor.

In Huynh et al. (2017) a VGGNet was trained on 561 ROIs of 64 patients. Patches of 111 x 111 were used as input of the VGGNet to extract features from the image. The color channels of the VGGNet were used for the different time points of the DCE MRI images (pre, post-contrast 1, post-contrast 2). The resulting feature vector of 1472 features was used as input of a linear discriminant analysis classifier.

Using only the pre-contrast MRI images resulted in the best predictive score. After leave-one-out cross-validation this resulted in an AUC of 0.85 (σ=0.033). Adding more contrast time points reduced the standard error, but had a worse predictive performance.

Ha et al. (2019) used a VGG 16 network for a tumor response prediction. A total of 3107 MR slices from 141 patients were used for training this network. A 64 × 64 crop of the segmented tumor was used as input for the network. The model classified the patients into three categories: complete response, partial response, and no response. Using pre-treatment MRI images, the CNN algorithm achieved an overall mean accuracy of 0.88 and an AUC of 0.98 in the three-class prediction of NAC treatment response, using five-fold cross-validation.

In most of the mentioned studies, the ROI determination is done manually and 2D slices of the MRI are used as input for the model. Only Qu et al. (2020) are using a 3D input of 128 × 128 × 16 for the model. Three studies (Qu et al., 2020; Ravichandran et al., 2018; Ha et al., 2019) are segmenting the contours of the tumor and excluded the tissue around the tumor for the response prediction. Two studies (Braman et al., 2020; Huynh et al., 2017) manually drew an ROI around the tumor.

El Adoui et al. (2020) investigated the role of the influence of the healthy tissue around the tumor on the prediction performance. A model was once trained on ROIs with the healthy tissue

surrounding the tumor and once on segmented tumors without the surrounding tissue. The model trained on the ROIs including the surrounding healthy tissue outperformed the model trained only on the segmented tumors.

Most studies included patients with different breast cancer subtypes for training their model. Only Braman et al. (2020) included only patients with HER2+ breast cancer. All studies are using only the DCE T1 sequence MR images for the prediction of breast cancer response. Two studies (Qu et al., 2020; El Adoui et al., 2020) used the pre-and post-NAC MR images, the other studies only used the pre-NAC MR images.

The scores of all deep learning models are outperforming or comparable to the best scores of the other machine learning classifiers in Table A.1. The highest performance is achieved by Qu et al. (2020).

3.1 White spots

All six studies required some form of tumor localization in the MR images. El Adoui et al. (2020), Qu et al. (2020), and Ha et al. (2019) did this by manually segmenting the tumor. Braman et al. (2020) and Huynh et al. (2017) did this by manually drawing an ROI around the tumor. Doing this manually requires experienced radiologists and takes a lot of time. Only Ravichandran et al. (2018) is using an automatic segmentation tool, which includes classical segmentation methods, such as thresholding and morphological operations.

Apart from being time-consuming, errors are easily made in a manual (but also an automatic) tumor segmentation. Due to the adhesion of tumor tissues to other tissues, the tumor boundaries are unclear and complicated to determine (Tian et al., 2021). Since the deep learning models are trained on the tumor segmentation results, likely, the performance of these deep learning models for response prediction is influenced by errors produced during the segmentation step.

Especially after NAC, the tumor boundaries are hard to determine. After NAC, for some patients, the tumor has almost or completely disappeared. Segmenting or even localizing the tumor would then be very challenging.

Only El Adoui et al. (2020) and Qu et al. (2020) were using the pre-and post-NAC MR images. From the pre-and post-NAC MR images, the same tissue location needs to be feed to the network as input (the tumor location in the pre-NAC MR image and the former tumor location in the post-NAC MR image). Since it is required to have the same locations in the breast in both sets of images, this adds an extra registration step to the whole pipeline.

In El Adoui et al. (2020) the images were aligned using an affine registration algorithm. In Qu et al. (2020) when no tumor was seen on the post-NAC MR image, the ROI was manually placed on the former tumor region. The affine registration algorithm makes the image pipeline extra complex and both methods can cause more errors in the ROI determination in the post-NAC MR images. This can, especially by doing the registration manually, even result in using the wrong locations in the post-NAC MR images.

Only Qu et al. (2020) used a 3D CNN. All other studies are using a 2D CNN and only one 2D slice of the 3D MR volume. By doing this, information along the z-axis is discarded. Possibly, there is more information along this third dimension of the MR volume since 2D images are not representative of a change in the tumor volume. With using 2D slices, there is a possibility of choosing the wrong slice (with no tumor presence), while the tumor is still present in the breast of the patient.

Regarding the added value of patient's clinical data, only Qu et al. (2020) and Ravichandran et al. (2018) have researched the influence of the clinical information (such as breast cancer subtype) on the prediction of breast cancer response. Ravichandran et al. (2018) only used the

HER2 status as additional input for the model. Qu et al. (2020) used HER2 status, ER status, and PR status as additional input for the model.

Only a few studies investigated the importance of ROI selection in response prediction. Ravichandran et al. (2018) used a pixel-wise probability map for this and concluded that the center of the tumor is most important for the prediction of the tumor response. The researchers did not include the healthy tissue around the tumor. El Adoui et al. (2020) used a gradient-based localization evaluation to visualize the tumor regions providing the most important features for the prediction and concluded that the healthy tissue outside the tumor can also give important features for the classification. However, it was not investigated how much of the healthy tissue around the tumor needs to be in the ROI for the best result.

Other limitations of these studies are: the studies are all retrospective studies and all studies use single-institutional data.

4 Methods

4.1 Materials

4.1.1 Software and hardware

For this project, Python 3.7 (Van Rossum and Drake, 2009) will be used to develop the prediction models. This programming language is chosen since Python has a lot of libraries that can be used for developing machine learning and deep learning algorithms. The deep learning models are developed using Keras (2.3.1) (Chollet, 2015) framework and Tensorflow (2.1.0) (Abadi et al., 2015) in the backend. Training of the models was executed using four graphics processing units (GPUs) (NVIDIA GeForce GTX 1080 Ti) with a memory of 11 GB each. IBM SPSS Statistics 26 (IBM Corp., 2019) is used for the statistical analysis.

4.1.2 Data

For this study, a dataset was collected retrospectively. Patients were included if a patient was diagnosed with breast cancer at the NKI-AvL or a referring hospital in the Netherlands between January 2019 and July 2020 and received a NAC therapy with a pre-NAC and post-NAC MR-scans as part of their treatment plan, followed by surgical resection of the (former) tumor location.

Patients were excluded if one of the following conditions applied: if the patient had silicone implants in the affected breast, no DCE T1 MR scans available in post- or pre-NAC timepoint, no pre-NAC MR scans available, no post-NAC MR scans available, MR scans were not accessible, and if no immunohistochemical information was available. The included patients were randomly divided into a train set (70%), validation set (15%), and test set (15%) while keeping the ratio between pCR and non-pCR the same over each set.

The MR images of the patients in the dataset include different image sequences: T1 DCE, T2, and DWI. The T1 DCE sequence contained for all patients a pre-contrast phase and a first post-contrast phase (3 minutes after injecting contrast). Some patients also had a second post-contrast phase image (7 minutes after injecting contrast). An overview of other MR acquisition parameters of the MR images can be found in Table 4.1.

The outcome of the pathological examination is used as a label to train the CNN. The outcome is categorized into two groups, pCR, and non-pCR. pCR is defined as the absence of invasive and in situ carcinoma in the resected breast tissue (ypT0).

4.2 Data pre-processing

To use the MR data, first, the MR data needs to be pre-processed. Pre-processing the data is necessary since the data feed into the CNN need to be from the same MR sequence and need to have the same size.

To answer the first sub-question, a dataset containing a large ROI and a dataset containing a small ROI have been obtained after the pre-processing steps. Figure 4.1 shows a flowchart of the pre-processing steps for the two datasets. A more detailed pseudo-code for both the pre-processing methods can be found in Appendix B.

Table 4.1: MR acquisition parameters of the MR images

Timepoint	Manufacturer	Scannertype	Magnetic strength (T)	TR (ms)	ET (ms)	Flip angle (°)	Slicethickness (mm)	Number of slices	Number of rows × columns
Pre-NAC	Philips Medical Systems	Achieva dStream (n=123)	3.0 (n=123)	3.215 (n=1), 3.240 (n=51), 3.243 (n=13), 3.244 (n=3), 3.665 (n=52), 3.667 (n=1), 3.999 (n=1), 4.248 (n=1)	1.633 (n=1), 1.637 (n=51), 1.639 (n=16), 1.728 (n=53), 1.857 (n=1), 1.979 (n=1)	10 (n=69), 12 (n=54)	1 (n=1), 1.5 (n=1), 1.8 (n=121)	200 (n=123)	400×400 (n=53), 480×480 (n=1), 528× (n=67), 640×640 (n=1), 720×720 (n=1)
		Ingenia (n=32)	3.0 (n=32)	3.296 (n=11), 3.313 (n=4), 3.662 (n=16), 3.787 (n=1)	1.628 (n=4), 1.632 (n=11), 1.727 (n=16), 1.969 (n=1)	10 (n=16), 12 (n=16)	1.8 (n=32)	200 (n=32)	400×400 (n=16), 528×528 (n=15), 560×560 (n=1)
		Achieva (n=24)	1.5 (n=24)	3.267 (n=8), 3.305 (n=16)	1.639 (n=8), 1.712 (n=16)	10 (n=24)	2 (n=8), 2.3 (n=16)	157 (n=1) 200 (n=23)	320×320 (n=16), 480×480 (n=8)
Post-NAC	Philips Medical Systems	Achieva dStream (n=137)	3.0 (n=137)	3.240 (n=29), 3.243 (n=2), 3.244 (n=6), 3.665 (n=99), 3.780 (n=1)	1.637 (n=29), 1.639 (n=8), 1.728(n=99), 1.772 (n=1)	10 (n=37), 12 (n=100)	1.8 (n=137)	240 (n=1) 157 (n=16) 180 (n=8) 200 (n=112)	400×400 (n=99), 448×448 (n=1), 528×528 (n=36), 640×640 (n=1)
		Ingenia (n=41)	3.0 (n=41)	3.296 (n=15), 3.662 (n=25), 3.663 (n=1)	1.632 (n=15), 1.727 (n=26)	10 (n=15), 12 (n=26)	1.8 (n=41)	200 (n=41)	400×400 (n=26), 528×528 (n=15)
		Achieva (n=1)	1.5 (n=1)	3.305 (n=1)	1.712 (n=1)	10 (n=1)	2.3 (n=1)	200 (n=1)	320x320 (n=1)

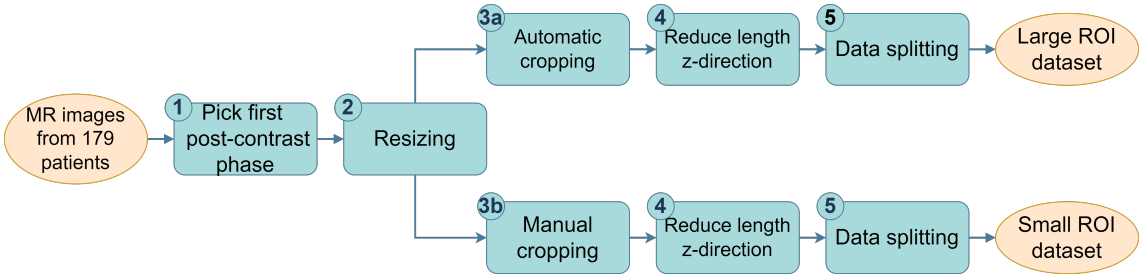


Figure 4.1: Flowchart data pre-processing steps to obtain the small ROI and large ROI dataset.

Sequence extraction

The first step of the pre-processing is to get the correct sequence for each patient. The first contrast phase (3 min after injecting contrast) of the DCE T1 MR images is used as input for the training and evaluation of the model. Compared to the pre-contrast phase (before injecting the contrast bolus), the first contrast phase MR image contains information related to blood flow and vascular permeability. Because of this, potentially the first contrast phase image contains important information for the model.

The tumor has a higher contrast with the rest of the image in the post-contrast phase compared to the pre-contrast phase. Because of this, it is hypothesized that the tumor can more easily be located by the model in the post-contrast phase image compared to the pre-contrast phase images. Since no segmentation of the tumor has been used, it can thus be beneficial to use the post-contrast phase images.

Because of these reasons, it seemed reasonable to choose the first contrast phase MR image. The second contrast phase (7 min after injecting contrast)is not used, because for a lot of patients the second contrast phase is not available.

For some patients, the different slides of each contrast phase of the DCE T1 images were not saved as separate scans but stacked one after the other in the same scan. This resulted in an MR volume with the pre-contrast phase, first post-contrast phase, and second post-contrast phase stacked one after each other. To be clear, this means that the first slice was a pre-contrast phase image, the second slice a first post-contrast phase image, the third slice, a second post-contrast phase image. The fourth slice was a pre-contrast phase image again, the fifth a post-contrast phase image, and so on. For these patients, the first contrast phase needs to be extracted and saved as separate scans. This can be done by selecting every third slice starting from the second slice.

Resizing

Not all MR volumes had the same size, so the second pre-processing step includes resizing some of the MR volumes. Most MR volumes had a size of $400 \times 400 \times 200$ voxels. The MR volumes that had a different size were resized to this size. This is needed since the model can only accept MR images as input of the same size.

For the resizing, a bicubic interpolation method is used. Compared to two other common interpolation techniques, bilinear, and nearest-neighbor interpolation, bicubic interpolation has a much higher level of computation complexity and thus takes more time, but also results in much sharper images and it has the smallest error rate for image resizing (Triwijoyo and Adil, 2021).

ROI determination

The third pre-processing step is determining the ROI and cropping the MR volumes in the xy-plane. Step 3a in Figure 4.1 is an automatic cropping method by using the information on the

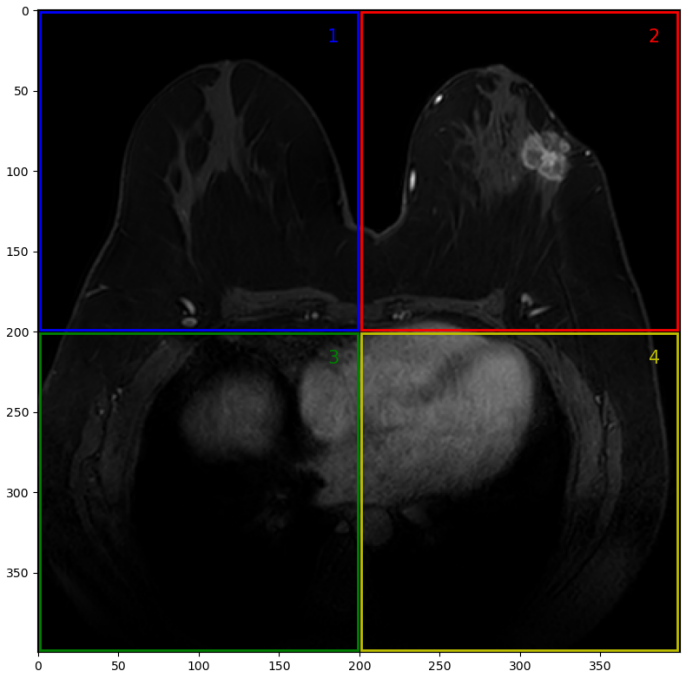


Figure 4.2: An MR image (xy-plane) of one of the MR volumes. The image can be divided into four quadrants, depending on the laterality the right area of the image will be cropped. In this image, in quarter 2 a tumor can be seen in the breast. Cropping will be done for every slice in the MR volume, the created volume will be used as input for the model.

affected breast of the patient (the laterality). When taking one slice from the MR volume, the slice can be divided into 4 equal quarters, like in Figure 4.2. Depending on the laterality, the affected breast will be in quarter 1 or 2. When cropping this area in all adjacent slices of the MR volume, this results in a volume of size $200 \times 200 \times 200$, containing one-quarter of the original MR volume.

Step 3b is the manual cropping of only the breast tissue in the MR volumes. This will result in a crop, only containing the breast tissue and tumor of the patient. The four coordinates defining the (square) crop will be used to crop the breast in all the MR slices of the patient. The blue rectangle in Figure 1.1 is an example of this crop. The size of the largest crop after this step is $184 \times 186 \times 200$. All smaller cropped volumes are zero-padded to this size so that all MR volumes in this dataset also have the same input size. Due to the zero-padding, the image size from the small ROI dataset is not that different compared to the size of the large ROI dataset.

Length reduction z-direction

Due to the GPUs memory limitation, it was not possible to use all 200 slices in the MR volume. For an MR volume of 200 slices, the GPU can only handle a batch size of one. Otherwise, the batch size will be too large for the memory of the GPU. When sticking to a batch size of one, training a CNN would take a very long time. For these reasons, pre-processing step four includes reducing the length of the z-direction of the MR volumes.

When taking a subset of 30 slices of the total MR volume, the GPU can handle to train the model with batch sizes up to eight. Since the first 25 and last 25 slices of the MR volume, do not contain imaged breast tissue, the MR volume can be cropped in the z-direction to a size of 150 slices. When taking every 5th slice, a total of 30 slices remains, leaving a total input size for the MR volume of $200 \times 200 \times 30$ for the large ROI dataset and $184 \times 186 \times 30$ for the small ROI dataset.

Of course, by doing this, information from some slices is lost. An alternative would be to resize the MR volumes to a size of $200 \times 200 \times 30$ and $184 \times 186 \times 30$, but then some form of interpolation technique is needed, altering the data. Interpolation can deform patterns in an around the tumor, negatively effecting performance of the CNN. To avoid interpolation as much as possible, it has been chosen to take a subset of the slices.

Data splitting

In the fifth and last step for both preprocessing methods, the dataset is divided into a train, validation, and test set, while keeping the ratio between pCR and non-pCR labels the same for each set. For the training set, 70% of the dataset is used, while for the validation and test set 15% of the dataset is used. After splitting the data, the MR images of each of the three datasets are stored in separate folders. The data generator can feed the MR data from these locations into the CNN.

4.3 Models

Since the MR volumes are data with three dimensions, a 3D CNN has been used for the prediction of the tumor response. This study is not to obtain a new architecture for a CNN, so known 2D CNN architectures are transformed to 3D architectures so that the three-dimensional data can be used as input for the CNN.

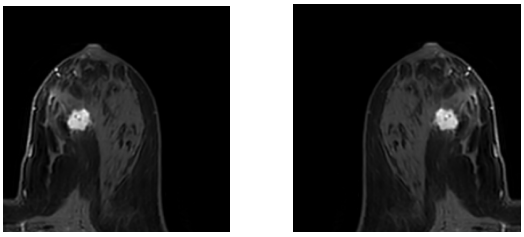
First, the 2D single-input CNN from El Adoui et al. (2020) (Figure 3.2) has been converted to a 3D model. It is chosen to start with this model because it got only one input for one MR volume. This makes it easier to program the architecture and the data generator and, later on in the project, add extra inputs to the model and change the setup and data generator accordingly.

Second, the architecture from Braman et al. (2020) (Figure 3.1) and El Adoui et al. (2020) (Figure 3.3) have been converted to a 3D double-input CNN. Both studies are using the same 2D architecture for their CNN. This architecture consists of four blocks of 2D convolutional layers. The convolutional layers are followed by a ReLu activation function and max-pooling layer. It is chosen to convert this 2D model to a 3D model since these models achieve the highest performance on the 2D classification task from the double-input models of the studies mentioned in **??**.

Both the single-input and double-input 2D architectures can be converted to a 3D architecture by changing the 2D convolutions in the architecture to 3D convolutions. Also, the max-pooling operation for 2D spatial data had been converted to a max-pooling operation for 3D volumetric data. For both models, every convolutional layer has a kernel initializer and has filter sizes of $3 \times 3 \times 3$. The 3D single-input CNN consists of one branch with four blocks of (3D) convolutional layers. The double-input consists of two branches with both four blocks. For both models, each block is followed by a ReLu layer and max-pooling layer.

For both models, the last convolutional block is followed by a fully connected layer (FCL). To prevent overfitting, this layer is followed by a drop-out layer. The final layer is an FCL with a softmax activation function and two nodes. In this way, the output of the model consists of two probabilities, a probability for the patient being that class (either pCR or non-pCR). To further prevent overfitting, L2 regularization is used in the convolutional and fully connected layers. A figure of the architecture of both models can be found in Section C.1.

Qu et al. (2020) is achieving the highest performance on the response prediction and is showing promising results. However, the 2D architecture from Qu et al. (2020) will not be converted to a 3D architecture since it requires all six phases of the DCE T1-weighted images. The used dataset for this study did not include six contrast phases of the DCE T1 MR scan for all the patients.



(a) Image **(b)** Flipped image

Figure 4.3: The resulting MR images before and after horizontal flipping

4.4 Data generator and data augmentation

After pre-processing the MR data and programming the model, a data generator must be programmed to feed the MR data in the model. The default data generator in Keras (Chollet, 2015) is only suitable for 2D image classification tasks and can thus not be used for this project. To overcome this, two custom data generators have been programmed for this project, one for the 3D single-input model and one for the 3D double-input model.

A data generator can load the saved MR volumes and labels and feed them into the CNN. It does this by getting a list of IDs of the current batch and feeding the corresponding MR volumes and labels into the CNN. The single-input data generator loads one MR volume per patient and the double-input data generator loads two MR volumes per patient. Pseudocode for the data generators can be found in Section B.3.

In the data generator, the MR volumes are scaled between zero and one. It is important to scale the input values between zero and one since the weights in a CNN are initialized to small numbers. Using very large numbers would slow down or lead to an unstable learning process.

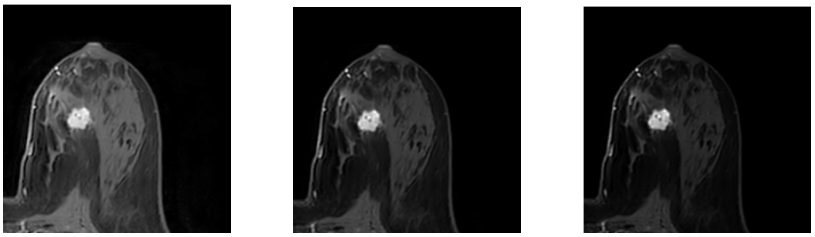
4.4.1 Data augmentation

Data augmentation has also been done in the data generator. Since the training set that is used in this project to train the models only consists of MR volumes from 125 patients, the diversity of the training examples is very limited. The lack of quantity and diversity in the data negatively influences the performance of the models. Data augmentation is used to generate new training data from the existing training data, increasing the number of training examples and variability in the dataset. This reduces the overfitting of the model on the training data and helps improve the performance of the model.

Five augmentation techniques are applied on the MR volumes: flipping, gamma correction, translation, rotation, and scaling. Applying the augmentation on the data during the training of the model is called on the fly data augmentation. This means that before loading the batch into the model during training, augmentation is applied to the MR volumes. Every epoch new augmentation parameters are determined, resulting in slightly different images used during training every epoch.

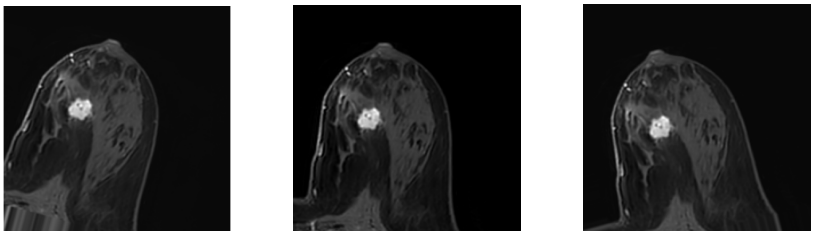
Flipping

What needs to be taken into account is that the MR images won't get unrealistic after applying the augmentation techniques. When flipping is done in the vertical direction (along the horizontal axis), it would look like the MR scan (and thus the breasts) are upside down. Since the breasts are symmetrical along the vertical axis, flipping the MR images along the horizontal axis would not give a realistic representation of the breasts. So flippling is only done in the horizontal direction (along the vertical axis). An example of an MR image before and after horizontal flipping can be seen in Figure 4.3



(a) $\gamma = 0.65$ **(b)** $\gamma = 1$ **(c)** $\gamma = 1.25$

Figure 4.4: The resulting MR images after gamma correction: $im_{out} = im_{in}^{\gamma}$, for $\gamma = 0.65$, $\gamma = 1.25$ (the outer two values) and $\gamma = 1$ (no gamma correction)



(a) -15^o **(b)** no rotation **(c)** 15^o

Figure 4.5: The resulting MR images after rotating the image with -15^o and 15^o (the outer two values) and no rotation

Gamma correction

For gamma correction the following formula has been used: $MR_{out} = MR_{in}^{\gamma}$. For γ larger than one, the output image will be darker than the input image and for γ smaller than one, the output image will be lighter than the input image. γ is randomly chosen for every patient every epoch during training between 0.65 and 1.25. The resulting MR images after gamma correction with these two boundary values for γ can be seen in Figure 4.4

Rotation

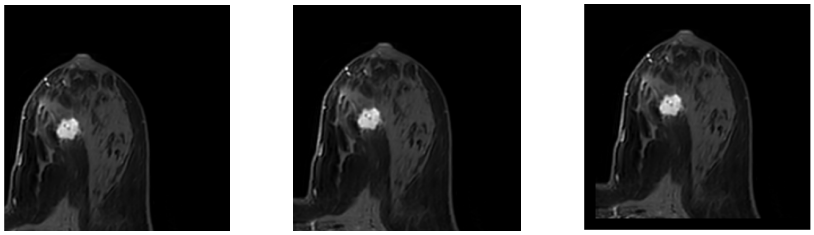
For rotation, too large angles will result in breasts that are not oriented correctly, for instance, breasts that are oriented sideways when rotated with 90^o. So too large rotation angles will not result in realistic-looking breast MR images. For these reasons rotation angles are between -15^o and 15^o are chosen. Rotating the images with these angles, the breasts are still oriented in a way that they can be imaged in an MR scanner. The MR volumes are rotated in the xy-plane. The resulting MR images after rotating the image with these boundary values can be seen in Figure 4.5

translation

Translating the MR volumes is also done in the xy-plane. For translating the volume, the translation in the x-direction and the y-direction needs to be chosen, there is no translation in the z-direction. To prevent tumors close to the border of the MR image fall out of the image, the translation in the x- and y-direction is randomly chosen between -10 and 10 voxels. After the translation the image is zero-padded. An example for two translations can be seen in Figure 4.6

Scaling

Scaling the MR volumes is the fifth and last used augmentation technique. For a scale factor larger than 1, the MR volume is enlarged and for a scale factor smaller than 1, the MR volume is shrunken. After enlarging, the MR volume is cropped to its original size and after shrinking, the MR volume is zero-padded, so that the input sizes stay the same. For the tumors close to

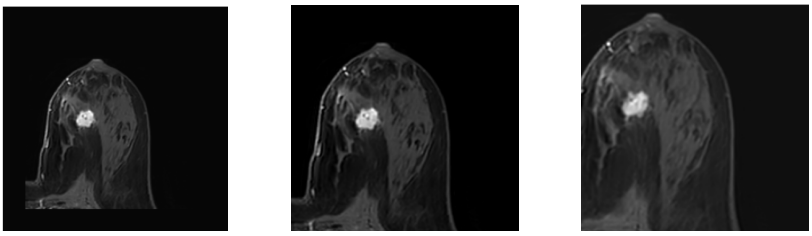


(a) translation = (-10, 10)

(b) no scaling

(c) translation = (10, -10)

Figure 4.6: The resulting MR images after translating the image with (-10, 10), (10, -10) (the outer two values) and (0,0) (no translation)



(a) scalingfactor = 0.8

(b) no scaling

(c) scalingfactor = 1.2

Figure 4.7: The resulting MR images after scaling the image with a factor of 0.8, 1.2 (the outer two values) and 1.0 (no scaling)

the border of the MR volume not to fall off the MR volume, the maximum scaling factor is 1.2. For the breast not getting too small, the minimum scaling factor is 0.8. An example of an image after scaling with these two scaling factors can be seen in Figure 4.7

Scaling and rotation both need an interpolation method to determine the new voxel values. Within the used methods, the interpolation techniques available are the nearest-neighbor, bilinear, bicubic, quadratic, and quintic interpolation. Also for these interpolation methods, a bicubic interpolation had been chosen, since this gives better interpolation results than for nearest-neighbor and bilinear (Triwijoyo and Adil, 2021), but doesn't cost as much time as quadratic or quintic interpolation.

Due to the image interpolation used with the two augmentation techniques, it takes a lot of time to train the model if all the augmentation techniques are applied to the volumes every epoch. To save time during training, randomly two augmentation techniques are chosen for every patient every epoch. For the two randomly chosen augmentation techniques, the parameters are chosen randomly between the set boundaries for these parameters.

4.5 Experimental setup

4.5.1 ROI size

To make a comparison between both the two ROI sizes, both the generated 3D single-input CNN and 3D double-input CNN will be trained on both datasets.

For getting the best performance from the models, the optimal hyperparameters for the model will be trained first. The hyperparameters of each model will be trained with a Bayesian optimization algorithm, resulting in an optimal set of hyperparameters for each model. Bayesian optimization is chosen since this algorithm uses a Gaussian process to approximate the function to optimize. This can speed up the hyperparameter optimization compared to other optimization algorithms like random or grid search.

Table 4.2: The trained hyperparameters and their boundaries. The hyperparameters are trained with Bayesian optimization. For each hyperparameter, the kind of hyperparameter (real or categorical) and boundaries or categories are shown.

Hyperparameter	**Sort**	**Boundaries/Categories**
Learning rate	Real	$[1 \times 10^{-6}$ - $1 \times 10^{-3}]$
Decay	Real	$[1 \times 10^{-7}$ - $1 \times 10^{-2}]$
Batch size	Categorical	[4, 8]
drop-out	Real	[0.0 - 0.5]
Loss function	Categorical	[Weighted BCE, weighted MSE]
Optimizer	Categorical	[ADAM, SGD, RMSprop]
L2 learning rate	Real	$[1 \times 10^{-7}$ - $1 \times 10^{-4}]$
Weight initializer	Categorical	[He normal, Random normal]
Number of filters block 1	Categorical	[4, 8, 16, 32]
Number of filters block 2	Categorical	[8, 16, 32, 64, 128]
Number of filters block 3	Categorical	[8, 16, 32, 64, 128]
Number of filters block 4	Categorical	[8, 16, 32, 64, 128, 256]
Number of nodes FCL	Categorical	[128, 256, 384, 512, 640, 768, 896, 1024]

For Bayesian optimization, two things need to be determined, thy hyperparameters to train and the total number of evaluations. The total number of evaluations is set to 200. The hyperparameters to be trained and the boundaries of each hyperparameter can be seen in Table 4.2.

To determine the boundaries of each hyperparameter, first, the models are trained by setting the hyperparameters manually. Reasonable boundaries are set around the manually determined hyperparameters resulting in the best performance for each model after training.

Since the distribution of the patients along the two classes is not equal, the dataset is slightly imbalanced. In the dataset, there are more cases of pCR than there are non-pCR cases. The imbalanced distribution of the two classes can affect the performance of the model. It is possible that the model simply for all cases predicts the majority class for every patient. To compensate for this, a weighted loss function had been used. Two weighted loss functions are used as categories in the hyperparameter optimization, a weighted binary cross-entropy (BCE) loss function and a weighted mean squared error (MSE) loss function.

The number of epochs is not treated as a trainable hyperparameter, since an early stopping argument is used to stop the training. An early stopping argument determines when the model starts to overfit on the training data and stops the training when this happens. The early stopping argument is monitoring the validation loss and when this is not decreasing for over 10 epochs, it stops the training of the model. The maximum number of epochs for the training is set at 200.

The training set, validation set, and test set are used for training the hyperparameters. The best set of hyperparameters is determined based on the highest AUC score on the test set.

After finding the optimal set of hyperparameters for the single-input and double-input model, and for both the small and large ROI dataset, the models have been compared using a stratified 10-fold cross-validation. For the 10-fold cross-validation, the complete dataset (179 patients) has been used. Since the hyperparameters are already trained, no hold-out dataset is used and all patients are used for the 10-fold cross-validation.

Every fold, the CNN is thus trained on 90% of the data and validated on 10% of the data. After every fold, performance scores are calculated on the 10% validation set. For every fold, a ROC curve has been drawn. From these ROC curves, a mean ROC curve has been visualized and the corresponding mean AUC had been determined. The effect of the ROI size on the performance

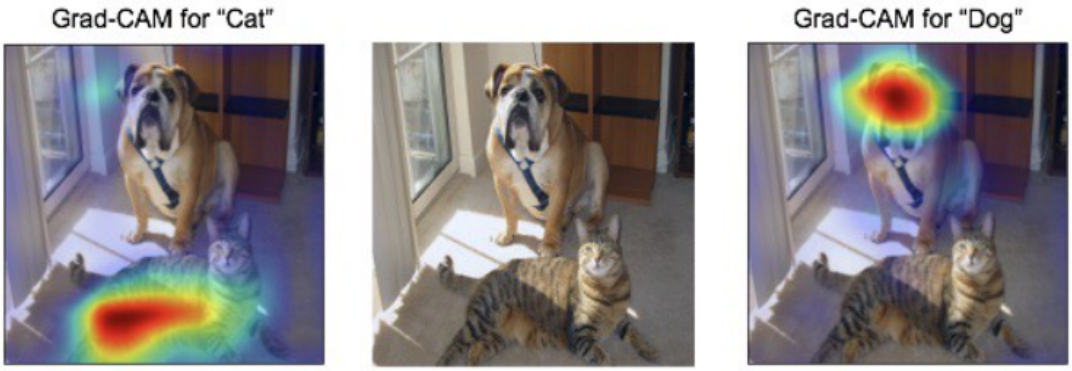


Figure 4.8: Example of Gradient-weighted Class Activation Mapping applied on a network trained on dogs and a network trained on cats. It can be seen in the images on which part of the image the model-based its decision on. From: Selvaraju et al. (2019)

of the single-input and double-input model will be determined by comparing mean AUC, mean accuracy, mean Matthews correlation coefficient (MCC), sensitivity, and specificity. Absolute differences are calculated for the mean AUC, accuracy, and MCC and a t-test is used to determine if these differences are statistically significant. The optimal sensitivity and specificity are determined by using Youden's index (Youden, 1950).

4.5.2 Gradient visualization

CNN's and other deep learning techniques are often treated as a black box. These techniques give little insight into how they make their classification, which makes these techniques hard to interpret. Also, it is hard to know where the network is 'looking' in the image. To make the prediction of the network more explainable and to answer subquestion 2, a technique called Gradient-weighted Class Activation Mapping (Grad-CAM) has been used. Grad-CAM is a technique that visualizes which parts of the image are important for the prediction of the network (Selvaraju et al., 2019). Figure 4.8 is showing an example of such a heat map. It can be seen which part of the images is important for the decision of the model.

By applying Grad-CAM a heat map will be produced. This heat map can be placed over the input image, highlighting which areas of the image have the most influence on the prediction of the model. With this information, it can be shown which areas of the MR volumes are important for the prediction of the NAC response. Grad-CAM can be applied on a trained network and there is no need for retraining the network.

For every 10-fold cross-validation, the algorithm is training and testing 10 models. To use the Grad-CAM algorithm, one model needs to be chosen from these 10 models. This is done by choosing the model with the highest AUC after one fold. This is done for the single-input CNN and double-input CNN, and every ROI size.

For the selected models, for every patient in the test set, a heat map has been made using Grad-CAM. The heat map has been overlayed on every slice of the MR volume. By visually inspecting the created heat maps for every patient it had been determined which areas of the MR images are most important for the model's prediction of the response of the breast tumor to NAC.

4.5.3 Extra input: Cancer subtype

To improve upon the model and to answer subquestion 3, the ER, PR, and HER2 status have been used as extra input for the model. For this, the model needs to have an extra input for the molecular subtype.

Both the single-and double-input models have been adapted to use the receptor statuses as an extra input. To do so, the input of the molecular subtype is concatenated with the output from the convolutional branches. Also, the data generator needs to have a small adaptation to be able to feed this extra information into the model during training. A figure of the architecture of the model can be found in Appendix C.

The dataset with the ROI size which gives the best performance in the previous section has been used to train the two models. As in the previous section, the optimal hyperparameters of these two models have been determined with Bayesian optimization. To compare the models a stratified 10-fold cross-validation had been performed. The mean ROC curve will be visualized and the models have been compared in mean AUC, mean accuracy, mean MCC, sensitivity, and specificity.

5 Results

5.1 Patient characteristics

As indicated in Figure 5.1 a total of 179 out of 328 patients were included in this study. In this patient group, 124 patients reached pCR, and 55 did not reach pCR (non-pCR). Splitting the dataset resulted in a training set of 125 patients, and a validation and test set of both 27 patients.

Table 5.1 shows a summary of the patient characteristics of these patients in each group. A t-test is used to check if there is a significant difference between the pCR and non-pCR groups within each set. A chi-squared test is used to check if the other categorical variables (the clinical staging, ER, PR, and HER2 statuses) give significant differences between the pCR and non-pCR groups within each set.

No significant difference was found between the mean age of the pCR group and the non-pCR group (49.10 years and 52.13 years, respectively, p=0.107). Also within the training, validation, and test set no significant differences were found in the mean age between the pCR and non-pCR group (p=0.199, p=0.595, and p=0.113, respectively).

Figure 5.2 shows the distribution of the age of the patients in the complete dataset. The youngest patient is 25 years and the oldest patient is 77 years old at the moment of diagnosis. The mean age of the complete dataset is 50.03 years. Figure 5.3 shows the age distribution for each dataset for the pCR and non-pCR groups.

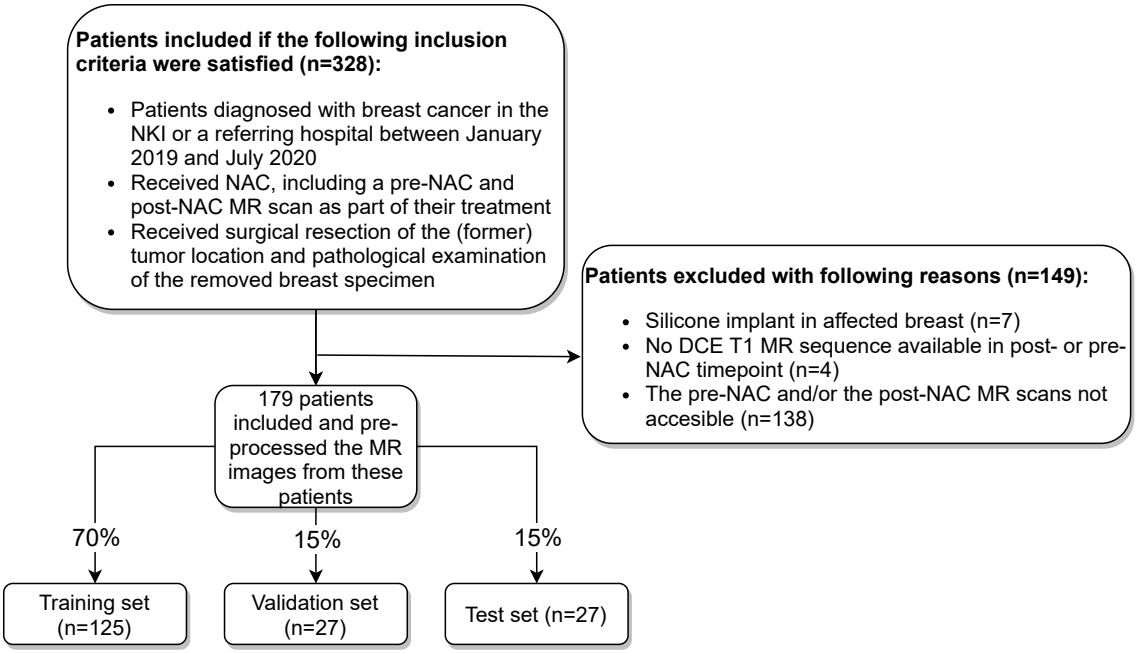


Figure 5.1: Flowchart showing the inclusion and exclusion criteria. A total of 328 patients with breast cancer between January 2019 and July 2020 were included initially in this study. All these patients received NAC and the breast tumor is removed surgically. 149 Of these patients were excluded.

As Table 5.1 indicates, only for the HER2 receptor, a significant difference is found between the pCR and non-pCR group in the complete dataset (p=0.037). After splitting the complete dataset in the training, validation, and test set, no significant differences were found between the pCR and non-pCR group for the HER2 receptor. For the ER and PR receptor, no significant differences were found between the pCR and non-pCR group in all datasets.

Table 5.1: Patient characteristics for the complete dataset, training set, validation set, and test set. A t-test is used to check if there is a significant difference between the pCR and non-pCR groups within each set. A chi-squared test is used for the other (categorical) variables. Clinical staging is graded with the TNM staging system.
ER: Estrogen receptor, **PR**: Progesterone receptor, **HER2**: Human Epidermal growth factor receptor 2, **T**: Tumor, **N**: Node, **M**: Metastasis
(1) Complete clinical staging is not known for one patient in the training set.
(2) Tumor status could not be assessed for one patient in the training set.
(3) Lymph node status could not be assessed for one patient in the training set.

Characteristics	Complete dataset			Training set			Validation set			Test set		
	pCR (n=124)	non-pCR (n=55)	p	pCR (n=87)	non-pCR (n=38)	p	pCR (n=18)	non-pCR (n=9)	p	pCR (n=19)	non-pCR (n=8)	p
Mean age±SD (years)	49.10±11.77	52.13±10.88	0.107	49.47±11.73	52.45±12.09	0.199	48.39±10.54	50.56±8.23	0.595	48.11±13.47	52.38±7.75	0.113
ER			0.102			0.058			0.766			0.732
Positive	72	39		46	27		13	6		13	6	
Negative	52	16		41	11		5	3		6	2	
PR			0.068			0.057			1.000			0.561
Positive	43	27		30	20		8	4		5	3	
Negative	81	28		57	18		10	5		14	5	
HER2			0.037			0.322			0.083			0.135
Positive	59	17		38	13		8	1		13	3	
Negative	65	38		49	25		10	8		6	5	
Clinical staging(1)												
T(2)			0.268			0.778			0.070			0.469
1	37	14		23	8		7	3		7	3	
2	65	28		46	21		10	2		9	5	
3	19	8		15	6		1	2		3	0	
4	2	4		2	2		0	2		0	0	
N(3)			0.000			0.012			0.105			0.067
0	73	14		51	10		10	1		12	3	
1	41	27		29	18		5	6		7	3	
2	2	1		1	1		1	0		0	0	
3	8	11		6	7		2	2		0	2	
M			0.001			0.001			0.150			0.508
0	122	47		86	31		18	8		18	8	
1	2	7		1	6		0	1		1	0	

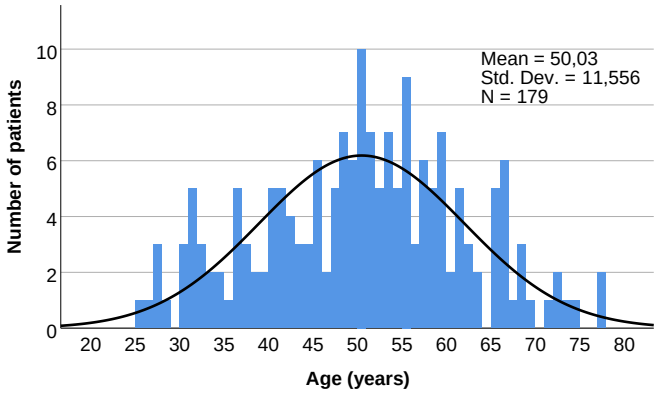


Figure 5.2: Histogram of the age distribution of the complete dataset (n=179). The minimum age at the time of diagnosis is 25 years. The maximum age is 77 years. The mean age is 50.03 years with a standard deviation of 11.556. The black line shows a normal distribution with the same mean and standard deviation.

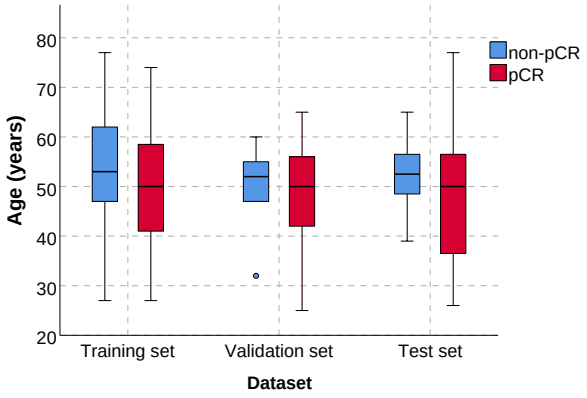


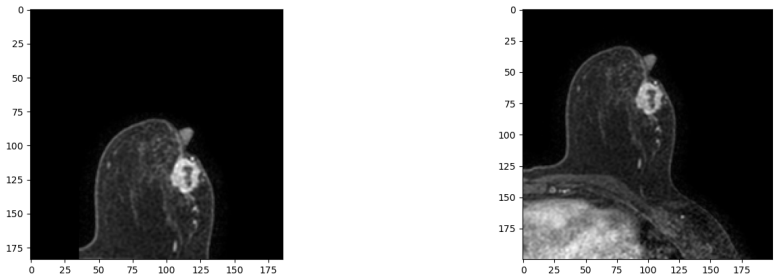
Figure 5.3: Boxplot of the patient's age for the train set, validation set, and test set for both the pCR and non-pCR group

Significant differences are found between the pCR and non-pCR group in the complete dataset for the lymph node status and metastasis status (p=0.000 and p=0.001 respectively). In the training set also a significant difference is found for these variables (p=0.012 and p=0.001), but the differences between these two groups were not statistically significant in the validation and test set. For the tumor status variable, no statistical difference between the pCR and non-pCR group were found in all the datasets.

5.2 ROI size

After pre-processing, the resulting size for the large ROI dataset is $30 \times 200 \times 200$ voxels. The MR volumes in the small ROI dataset had a size of $184 \times 186 \times 30$ voxels. Due to zero-padding the small ROI images, the sizes of the MR volumes do not differ much. Although not being much different in size, the information in the images is different, as can be seen in Figure 5.4.

Table 5.2 shows the optimal hyperparameters using the Bayesian optimization for both the single-input and double-input model for the large ROI dataset and the small ROI dataset. The last row of Table 5.2 shows the total trainable parameters of each model after compiling the model with the found hyperparameters. The single-input model (small ROI) has the largest amount of trainable parameters. The double-input model (small ROI) has the least amount of trainable parameters. A schematic image of the architecture of each model can be found in Appendix C.



(a) small ROI **(b)** large ROI

Figure 5.4: Example of a small ROI after zero-padding and a large ROI of the same MR slice and same patient.

Table 5.2: The optimal hyperparameters after the Bayesian optimization for the single-input and double-input models for the small and large ROI. The last row shows the total trainable parameters of each model after compiling the model with the found hyperparameters.

Hyperparameters	**Single-input (large ROI)**	**Single-input (small ROI)**	**Double-input (large ROI)**	**Double-input (small ROI)**
Learning rate	2.937×10^{-4}	2.460×10^{-5}	3.892×10^{-5}	1.501×10^{-6}
Decay	1.587×10^{-7}	2.305×10^{-5}	6.317×10^{-3}	8.939×10^{-4}
Batch size	4	4	4	4
Loss function	Weighted binary crossentropy	Weighted binary crossentropy	Weighted binary crossentropy	Weighted binary crossentropy
Optimizer	Adam	RMSprop	SGD	RMSprop
Drop-out	0.2	0.2	0.4	0.5
L2 learning rate	4.818×10^{-7}	8.457×10^{-5}	9.663×10^{-7}	5.840×10^{-6}
Number filters block 1	16	16	8	16
Number filters block 2	64	32	32	8
Number filters block 3	16	128	16	32
Number filters block 4	8	8	64	8
Number of nodes FCL	128	384	512	512
Weight initializer	He initialization	He initialization	He initialization	He initialization
Total parameters	203 802	1 087 358	628 914	170 834

Table 5.3: The mean performance scores after the 10-fold cross validation for the single-input models for both the small and large ROI. The last column shows the absolute difference between the two models for the mean AUC, mean accuracy, and mean MCC.

Performance scores	Single-input (large ROI)	Single-input (small ROI)	Absolute difference
Mean AUC ± std	0.75 ± 0.06	0.74 ± 0.09	0.01 (p=0.773)
Mean accuracy ± std (%)	62.61 ± 13.74	69.90 ± 7.77	7.29 (p=0.161)
Mean MCC	0.16 ± 0.15	0.22 ± 0.25	0.06 (p=0.523)
Sensitivity (%)	56	74	
Specificity (%)	83	66	

Table 5.4: The mean performance scores after the 10-fold cross validation for the double-input models for both the small and large ROI. The last column shows the difference between the two models for the mean AUC, mean accuracy, and mean MCC.

Performance scores	Double-input (large ROI)	Double-input (small ROI)	Absolute difference
Mean AUC ± std	0.68 ± 0.05	0.69 ± 0.08	0.010 (p=0.741)
Mean accuracy ± std (%)	65.26 ± 7.20	70.39 ± 7.03	5.130 (p=0.124)
Mean MCC	0.03 ± 0.14	0.15 ± 0.25	0.120 (p=0.202)
Sensitivity (%)	66	75	
Specificity (%)	72	66	

Table 5.3 shows the mean AUC, mean accuracy, mean MCC, sensitivity, and specificity of the single-input model trained on the large ROI and small ROI dataset after the 10-fold cross-validation. This table also shows the differences between the AUC, accuracy, and MCC for the two datasets. Table 5.4 shows these results for the double-input model trained on the large ROI and small ROI dataset.

In Figure 5.5 the mean ROCs for the double-input model trained on the large ROI and small ROI are plotted. The mean AUC for the double-input model trained on the small ROI is 0.69 ± 0.06. For the double-input model trained on the large ROI, this is 0.68 ± 0.05.

In Figure 5.6 the mean ROCs for the single-input model trained on the large ROI and small ROI are plotted. The mean AUC for the single-input model trained on the small ROI is 0.74 ± 0.09. For the single-input model trained on the large ROI, this is 0.75 ± 0.06.

For the models, the ROCs and corresponding AUCs of each fold can be found in Appendix C.

5.3 Gradient visualization

The Grad-CAM algorithm has been applied on the test set from the best folds from 10-fold cross-validation from the single-input and double-input models trained on the large and small ROI datasets. The best fold has been determined by the highest AUC. The AUC of each best fold can be found in Table 5.5. Figure 5.7 shows 8 generated heat maps for the single-input model. Figure 5.8 shows this for the double-input model.

Table 5.5: The AUC of the best fold of the 10-fold cross-validation for each model and each ROI size.

Performance scores	Single-input (large ROI)	Single-input (small ROI)	Double-input (large ROI)	Double-input (small ROI)
AUC	0.82	0.94	0.75	0.79

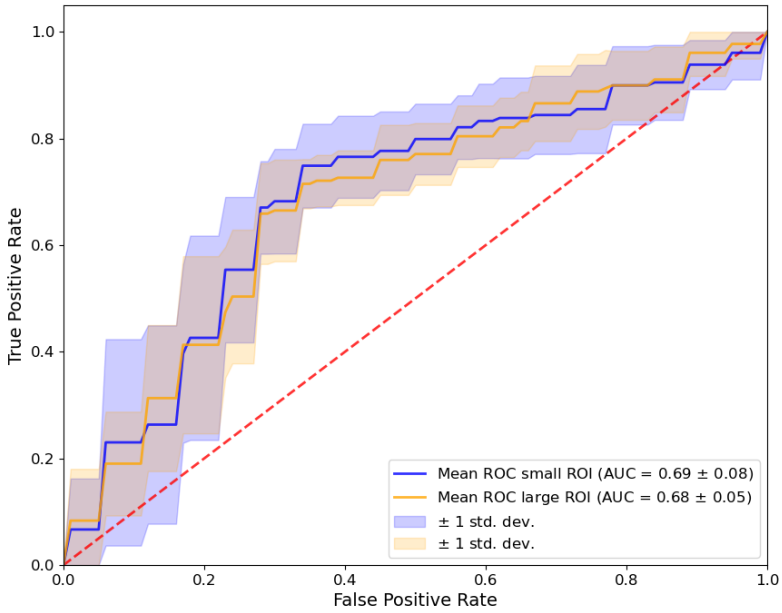


Figure 5.5: The mean ROC curve for the large and small ROI input for the double input model. The mean AUC and corresponding standard deviation for the small ROI is 0.69±0.08. For the large ROI, this is 0.68±0.05

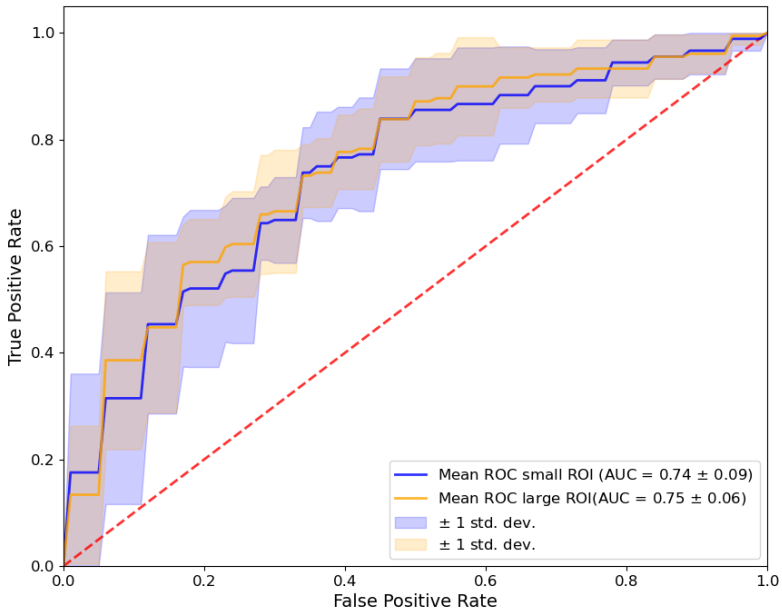
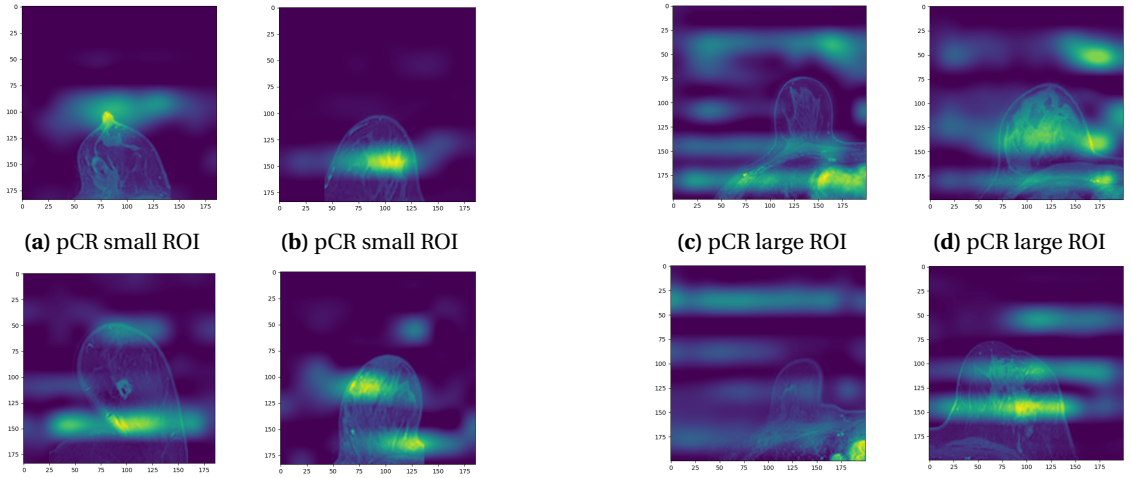


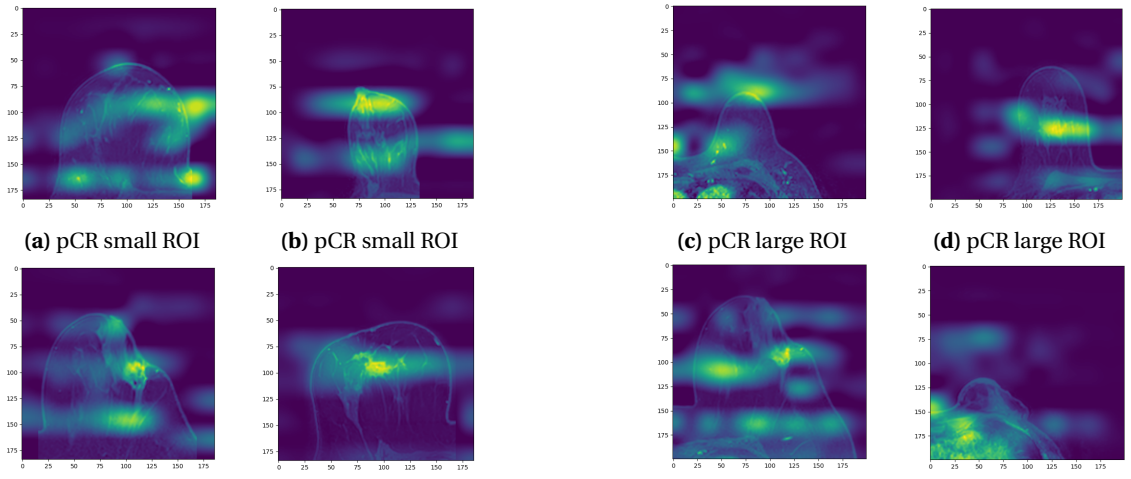
Figure 5.6: : The mean ROC curve for the single input model trained on the large and small ROI dataset. The mean AUC and corresponding standard deviation for the small ROI is 0.74±0.09. For the large ROI this is 0.75±0.06



(a) pCR small ROI **(b)** pCR small ROI **(c)** pCR large ROI **(d)** pCR large ROI

(e) non-pCR small ROI **(f)** non-pCR small ROI **(g)** non-pCR large ROI **(h)** non-pCR large ROI

Figure 5.7: Heat map examples for the single-input model



(a) pCR small ROI **(b)** pCR small ROI **(c)** pCR large ROI **(d)** pCR large ROI

(e) non-pCR small ROI **(f)** non-pCR small ROI **(g)** non-pCR large ROI **(h)** non-pCR large ROI

Figure 5.8: Heat map examples for the double-input model

Table 5.6: The optimal hyperparameters after the Bayesian optimization for the single-input and double-input models with the extra molecular subtype as input. The last row shows the total trainable parameters of each model after compiling the model with the found hyperparameters.

Hyperparameters	**Single-input (small ROI)**	**double-input (small ROI)**
Learning rate	6.231×10^{-4}	9.233×10^{-4}
Decay	8.666×10^{-6}	1.490×10^{-7}
Batch size	4	4
Loss function	Weighted binary crossentropy	Weighted binary crossentropy
Optimizer	SGD	SGD
Drop-out	0.45	0.2
L2 learning rate	9.371×10^{-7}	1.221×10^{-7}
Number filters block 1	32	32
Number filters block 2	128	64
Number filters block 3	64	64
Number filters block 4	32	8
Number of nodes FCL	384	512
Weight initializer	He initialization	He initialization
Total parameters	1 135 624	1 088 018

Table 5.7: The mean performance scores after the 10-fold cross validation for the single-input and double-input model with the molecular subtype as extra input for the model. The last column shows the difference between the two models for the mean AUC, mean accuracy, and mean MCC.

Performance scores	**Single-input (small ROI)**	**Double-input (small ROI)**	**Absolute difference**
Mean AUC ± std	0.72 ± 0.08	0.74 ± 0.08	0.02 (p=0.5830)
Mean accuracy ± std (%)	67.61 ± 6.38	$67.12 \pm 9,92$	0.49 (p=0.8969)
Mean mcc	0.05 ± 0.18	0.05 ± 0.28	0.00 (p=1.000)
Sensitivity (%)	74	78	
Specificity (%)	66	66	

5.4 Extra input: molecular subtype

Table 5.6 shows the optimal hyperparameters using Bayesian optimization for both the single- and double-input model with the molecular subtypes as an extra input. The last row of this table shows the total trainable parameters of both the models after compiling the models with the found hyperparameters. A schematic image of the architecture of both the models can be found in Appendix C.

Table 5.7 shows the mean AUC, mean accuracy, mean MCC, sensitivity, and specificity of the single-input and double-input model with the molecular subtypes as an extra input after the 10-fold cross-validation. Both models are trained with the small ROI dataset.

Table 5.8 shows the difference in the performance of the single-input model from Table 5.7 and the single-input model (small ROI) from Table 5.3. It also shows the difference in the performance of the double-input model from Table 5.7 and the double-input model (small ROI) from Table 5.4.

In Figure 5.9 the mean ROCs for the single-and double-input model with the molecular subtype as an extra input. Both models are trained on the small ROI dataset. The mean AUC for the double-input model is 0.74 ± 0.08. For the single-input model, this is 0.72 ± 0.08. For both models, the ROCs and corresponding AUCs of each fold can be found in Appendix C.

Table 5.8: The differences in performance scores between the single-input model with molecular subtype as extra input and the single-input model (small ROI) from Table 5.3 and the double-input model with molecular subtype as extra input and the double-input model (small ROI) from Table 5.4.

Performance scores	Single-input models	Double-input models
Absolute difference mean AUC	0.02 (p=0.606)	0.05 (p=0.179)
Absolute difference mean accuracy	2.29 (p=0.481)	3.27 (p=0.406)
Absolute difference mean MCC	0.17 (p=0.098)	0.10 (p=0.416)

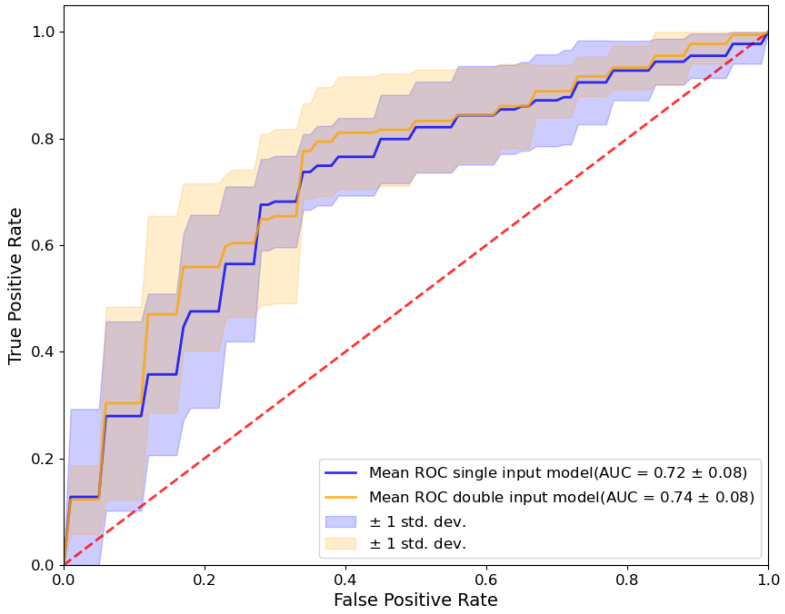


Figure 5.9: The mean ROC curve for the single-and double-input model with the clinical variables as an extra input. The mean AUC and corresponding standard deviation for the double-input model is 0.74 ± 0.08. For the single-input model, this is 0.74 ± 0.08

6 Discussion

6.1 Interpretation of the results

Table 5.1 shows that there is no significant difference in the mean age of the patients between the training set, validation set, and test set. Figure 5.3 shows that also the median age of the patients of each set is almost the same. This means that there is no age bias between the three datasets.

6.1.1 ROI size

When comparing the performance from the double-input models in Table 5.4, it can be seen that both the large ROI and small ROI almost give the same performance in terms of mean AUC. In terms of mean accuracy and the mean MCC, the small ROI has an increased performance compared to the large ROI, although both differences are statistically not significant (p=0.124 and p=0.202). The mean MCC of 0.03 for the double-input model trained on the large ROI dataset indicates that this model has almost no predictive value. The mean MCC of 0.15 for the double-input model trained on the small ROI dataset, indicates that the predicted labels and true labels are weakly correlated.

Comparing the performance of the two single-input models in Table 5.3 the performance of the two models are almost equally performing on the classification in terms of the mean AUC. In terms of the mean accuracy and mean MCC the model trained on the small ROI dataset is slightly outperforming the model trained on the large ROI dataset. These differences, however, are not statistically significant (p=0.161 and p=0.523).

The standard deviation for the four trained models in Table 5.3 and Table 5.4 for the mean accuracy and mean MCC is relatively high. This means that there are large differences between the performance of the model in every fold, meaning that the models are not very robust to changes in training and/or test data. The instability of the model across the different training sets also indicates that there is a high probability of overfitting of the model on the training data (Cawley and Talbot, 2010).

Comparing the single-and double-input model trained on the small ROI dataset, the single-input model is outperforming the double-input model based on mean AUC and mean MCC. Based on the mean accuracy, the two models are performing almost identical. Beforehand, it was expected that the double-input model could make a better prediction than the single-input model since it has the pre-and post-NAC MR scans as input and thus more information to base a prediction on. Table 5.2 shows that after the hyperparameter tuning, the number of convolutional filters in the single-input model is larger than for the double-input model. This means that the single-input model is able to get more (deep) features out of the image. These features might be important to be able to make a better prediction.

Also for the large ROI dataset, the single-input model is outperforming the double-input model based on mean AUC and mean MCC. Again, based on the accuracy, the models are almost having the same performance.

6.1.2 Gradient visualization

Figure 5.7 and Figure 5.8 show some examples of the generated heat maps by the Grad-CAM algorithm. The more a region lights up, the more important it is for the decision of the model.

It can be noticed that for some cases the decision of the model is (partly) determined by (former) tumor locations in the breast, like Figure 5.7b, Figure 5.7h, Figure 5.8d, Figure 5.8e, Figure 5.8f, and Figure 5.8g. However, not in all these mentioned images the classification of the

model is solely made by the (former) tumor location. For instance, in Figure 5.7f, Figure 5.8e, and Figure 5.8g, apart from the (former) tumor location, other locations light up aswell. This means that these other locations influence the prediction of the models.

It can be noticed in the heat maps of the large ROI images that some kind of wavy pattern is visible, like in Figure 5.7c, Figure 5.7g, and Figure 5.7h. This can mean that the model trained on the large ROI images is basing its decision (partly) on some kind of noise that is present in the images. This wavy pattern is much less present in the heat maps for the small ROI images.

Also, in the large ROI images, the model bases its decision a lot of times on structures outside the breast, like parts of the heart in Figure 5.7c, Figure 5.7g, Figure 5.8c, and Figure 5.8h. These structures have thus (a negative) influence on the performance of the model.

Other structures that influence the performance of the model, for both the small ROI and large ROI, are the nipple area (Figure 5.7a, and Figure 5.8b) and the skin fold between the breast and thorax (Figure 5.7f and Figure 5.7e). Both these areas have a large contrast with their surroundings and thus are easily detected on the MR images.

It can be concluded that structures with a large contrast with their surroundings have the largest influence on the prediction of the model. This can be the tumor, but also (a part of) the heart, the nipple and the skin fold between the breast and thorax. When no tumor is visible in the MR image (as for pCR patients in the post-NAC MR images), it is difficult for the model to determine on what location it needs to base its decision. Smaller ROIs can help the model determine the (former) tumor location of the tumor.

6.1.3 Extra input: molecular subtype

Looking in Table 5.7, comparing the single-input model and double-input with the extra input, there is no statistical difference in one of the performance scores between the two models.

In Table 5.8 the models with the molecular subtype as extra input are compared with the single- and double-input model trained on the small ROI. For both the single-and double-input models no significant differences in any of the three used performance scores are found. Expected was that the receptor statuses would improve the model, because previous studies have demonstrated the correlation of pCR and the ER, PR, and HER2 statuses (Liu et al., 2019; Cortazar et al., 2014; Braman et al., 2019). Also, Qu et al. (2020) and Ravichandran et al. (2018) showed an increase in the performance of the model after using the molecular subtype as extra input for their model.

A possible explanation for having no improvement in performance after integrating the receptor statuses in the model can be induced from the information in Table 5.1. No statistical difference is found in the training, validation, and test set for the number of ER, PR, and HER2 receptor status between the pCR and non-pCR groups. Also in the complete dataset, no statistical difference is found between the pCR and non-pCR group in the ER and PR receptor status. Only for the HER2 status, a statistical difference is found between the pCR and non-pCR group in the complete dataset. Although, previous studies have demonstrated the correlation of pCR and the molecular receptor statuses, for the dataset used in this study this correlation could only be found in the HER2 status for the complete dataset.

Because of this, a possible improvement can be made to only use the HER2 status as extra input for the model. Ravichandran et al. (2018) also left out the ER and PR status and only used the HER2 status as extra input for the model. This improved their model from an AUC of 0.77 to an AUC of 0.85 for classifying the pCR and non-pCR patients.

For this study, the information on the expression of the nuclear protein Ki-67 on the tumors was not available. In the review from (Li et al., 2015) it is stated that this protein can possibly be used as a marker to classify patients who are likely to respond to cancer therapy. When

using the expression of the Ki-67 protein on the tumor as extra input for the model, a possible improvement in the model's performance can be made.

In Table 5.1, it can be seen that significant differences are found between the lymph node status (N) and metastasis status (M) of the clinical staging between the pCR and non-pCR patient group (respectively, p=0.000 and p=0.001) in the complete dataset. In the training set these differences are also significant (p=0.012 and p=0.001). This indicates that there might be a correlation between the clinical staging and pCR status. It might be useful to investigate if this can be used as extra input for the model to improve the performance.

6.2 Comparison with literature

Comparing the performance of these models with the models from the studies in Table A.2 (Braman et al., 2020; El Adoui et al., 2020; Qu et al., 2020; Ravichandran et al., 2018; Huynh et al., 2017; Ha et al., 2019), all models are outperforming the models from this study in terms of accuracy, AUC, sensitivity, and specificity. The reason for this could be that all other studies are segmenting the tumor before making the prediction. This makes it unnecessary for the model to locate the tumor first. When using the whole breast as input for the model, the model first needs to localize the tumor and then make the response prediction. This can negatively affect the performance of the model.

Also, using 3D volumes as input for the model increases the number of parameters to be learned. The number of patients that is used to train the model is approximately the same compared to the other studies. When the number of parameters is increased and the number of training data is kept the same, this has a negative influence on the final performance.

The scores from the models from this study are comparable with the predictive performance of rCR in (Gampenrieder et al., 2019). The sensitivity and specificity from Gampenrieder et al. (2019) for predicting pCR are 75% and 67%, respectively. This sensitivity and specificity are almost the same as for the models used in this study.

When comparing the sensitivity and specificity with the minimally invasive biopsy to predict pCR from Heil et al. (2015), the sensitivity from almost all models in this study is higher than the sensitivity from Heil et al. (2015) (50.7%). But Heil et al. (2015) also reached a much larger specificity (93.5%) compared to the specificity of the models used in this study.

Other studies (Qu et al., 2020; Braman et al., 2020; Ha et al., 2019; Gampenrieder et al., 2019; Ravichandran et al., 2018) are using different definitions for pCR. In this study, pCR is defined as the absence of invasive cancer in the resected breast specimen, regardless of lymph nodes (ypT0). Using a different definition for pCR has an influence on the total number of pCR cases in a patient group (Choi et al., 2017) and influences what the model learns during training. To make a better comparison, the same definitions for pCR should be used.

6.3 limitations

This study has several limitations. First, the dataset contained only 179 patients, this is a very limited size for developing a deep learning method. This also makes the model easily prone to overfitting. Although using different kinds of regularization techniques, like data augmentation, drop-out, L2 regularization, and early stopping, overfitting of the model could not be prevented. Also, the data is acquired in only one institution.

Secondly, this study was a retrospective study. Due to this, the MR acquisition parameters could not be set the same for every MR image, as can be seen in Table 4.1. The difference in the MR acquisition parameters can influence the quality of the MR images and the performance of the model. Keeping these the same will probably improve the performance of the model.

Due to the different MR acquisition parameters, some of the MR images had different sizes in the x-and y-direction of the MR image. These images are all resized to the same size as the other images. For resizing, interpolation is used. Due to the interpolation, patterns in and around the tumor are getting deformed or can vanish. Since the information in the micro-vasculature around the tumor can be important for the response prediction, deforming this imaged micro-vasculature (and other imaged structures) can affect the performance of the model. Using MR data with the same image sizes means that less resizing is needed during pre-processing, this can improve the performance of the model.

During the data pre-processing, for every MR volume, every 5th slice is taken to get to a total of 30 slices. Since the tumor might be very small in the post-NAC images, it can be that the tumor is removed from the MR volume, while the tumor is still present. Reducing the number of slices in the MR scans was needed for the GPU being able to handle larger batch sizes than one.

Also, in the dataset that was used the majority of the patients (124 out of 179) reached pCR. In real life only 10% to 30% of the patients receiving NAC achieve pCR (Kong et al., 2011; Cortazar et al., 2014; Prevos et al., 2012). This means the models from this study are probably more inclined to predict pCR. This can result in many false-positive predictions when such a network is used in real-life.

The goal of this study was to predict pCR without the need for segmentation of the tumor in the MR images. This is also directly a limitation since the tumor is thus not localized in the MR image before making the prediction. It gets harder in this way for the model since the model is expected to localize the tumor and make the prediction of the response. From the results from Section 5.3 it can be seen that this is a complicated task to perform for the models.

Another limitation of this study is the fact that it is not researched which sequences of the MR images give the most information for the model to make the prediction. In this study for the double-input network, only the first post-contrast phase of the DCE T1 MR image of the pre-and post-NAC MR images are used. The performance of the model can maybe be improved by using the pre-contrast phase of the DCE T1 MR images or using the pre-and post-contrast phase MR images from only one MR timepoint (only the pre-or post-NAC images).

Lastly, all models are trained with the use of post-NAC MR images. With a model trained on only the pre-NAC and/or the MR scans after the first cycle of chemotherapy, it is possible to predict the response of the patients in an early stage of the chemotherapy. Since a large group of patients does not respond well to the chemotherapy (Kong et al., 2011; Cortazar et al., 2014; Prevos et al., 2012), these patients unnecessarily suffer from the side effects and toxicity of the chemotherapy (Reinisch et al., 2013).

A prediction in an early stage of the chemotherapy may allow for a more personalized treatment plan and change of treatment plan for patients not categorized as CR(Liu et al., 2010; Tudorica et al., 2016). To use a model in an early phase of the therapy, it seems also important to not only distinghuish patients with pCR, but also the patients with partial response, progressive disease and stable disease. Since in this study, for every model, post-NAC images are used, a prediction in an early phase of the treatment is not possible.

7 Conclusions and Recommendations

The goal of this study is to predict the response of breast tumors to neoadjuvant chemotherapy with a 3D CNN without the need for tumor segmentation. If this were to be realized, the radiologist does not need to make the segmentation slice by slice and this would rule out the subjectivity and variability in the segmentations. Also, it would potentially improve the performance of the prediction by using a 3D volume instead of a 2D slice. Lastly, by excluding the tumor segmentation, potentially the performance of the prediction can be improved due to the information in the TME. To reach this goal, a main research questions and three sub-questions were made up.

The first sub-question is: *What is the effect of the size of the ROI on the performance of a 3D CNN for the prediction of the breast tumor responses to NAC?*

It is expected that the models trained on the small ROI dataset would have a better performance than the models trained on the large ROI dataset. This is hypothesized since the large ROIs contain other anatomical structures by which the model can be 'distracted'. The results from this study do not support this hypothesis completely. A smaller size of the ROI slightly improves the mean accuracy and mean MCC for the single-and double-input model, although these differences are not statistically significant. For the two different ROI sizes, almost no difference is found in the mean AUC for both models.

This is followed by the second question: *Which areas of the MR images are most important for the model's prediction of the breast tumor response to NAC?*

From the data from this study, it is hard to say which areas of the MR images are most important for predicting pCR. What can be concluded is that the visible structures in the thorax in the large ROI images (like parts of the heart) are in most cases distracting the model and negatively influencing the performance of the model. Other structures that negatively influence the performance of the model are the area around the nipple and the skin fold between the breast and thorax.

In order to improve the performance of the model, the third sub-question is: *To what extent can the integration of the ER, PR, and HER2 receptor status as extra input for the model improve the model's performance?*

It is expected that the integration of the receptor statuses would improve the performance of the model. In this study, no statistically significant increase was reported in the mean AUC, mean accuracy, or mean MCC for the single-and double-input model after integration of the cancer subtypes. So it can be concluded that for this study, no improvement in the performance of the models is made after integration of the cancer subtypes. A possible reason is that no statistical differences were found in the ER, PR, HER2 receptor statuses between the pCR and non-pCR groups in this dataset.

Finally, to conclude the study, the main research question was:

To what extend is it possible to predict the response from breast tumors to neoadjuvant chemotherapy using a 3D multi-channel neural network and the pre-and post-NAC MR images without the need for tumor segmentation?

The best model after 10-fold cross-validation was able to predict pCR with a mean AUC of 0.74 and mean MCC of 0.22. This performance did not improve on the performance of state-of-the-art models. It seemed that one of the reasons for this is that it is hard for the model to localize the tumor. Especially when no tumor is visible in the MR image (as for pCR patients in the post-NAC MR images), it was hard for the model to localize the former tumor location. Only in

a few cases, the best model was able to solely make its prediction based on the (former) tumor location.

7.1 Recommendations

A few recommendations can be made for future studies. Since the main bottleneck of the model for the prediction of the pCR response is the localization of the (former) tumor in the MR volume, it is recommended to use smaller ROI areas. By using smaller ROI areas, it is easier for the model to localize the (former) tumor. A downside of using smaller ROI areas is that a radiologist should look at the MR images and determine the ROI. Somebody else cannot easily localize the tumor in the (post-NAC) MR images.

Another valuable option might be to use a segmentation CNN for transfer learning. A CNN trained for segmenting tumors can localize tumors in an MR image. When such a network is used for transfer learning, the models for the response prediction can possibly more easily localize the (former) tumor location and make a better prediction.

For the extra input of the model, it might be useful to research the added value of the TNM staging. In the dataset used for this study statistical differences were found between the pCR and non-pCR groups. Maybe this information can improve the performance of the model. Also, in this study, the status of the Ki-67 protein for each patient was not available. The Ki-67 protein can possibly be used as a marker to classify patients who are likely to respond to cancer therapy (Li et al., 2015), so this information might improve the model's performance.

Furthermore, adding more MR images to the dataset will increase the performance of the model. When increasing the dataset, it will also be useful to keep the MR acquisition parameters the same. Especially keeping the size of the MR images the same, to avoid resizing the MR images as much as possible. Also, a more realistic pCR/non-pCR ratio in the dataset will help improve the model.

Finally, it is highly recommended to research which contrast phases of the DCE T1-weighted MR image to use as input for the model. For this study, only the first contrast phase of the DCE T1 MR image is used to train the models. Using more or different contrast phases can possibly improve the performance of the models. The different contrast phases contain different kinds of information, so using more or other contrast phases can possibly improve the model's performance. Also, researching using other sequences than the DCE T1 images, like T2 or DWI, can be valuable. In literature, to the best of my knowledge, it is not tried to use other sequences than the DCE T1 images for training a CNN.

A Appendix: Results literature review

On the following pages the description of the state-of-the-art machine learning models and two tables with extracted information from the reviewed articles. Table A.1 shows the extracted information from the studies on machine learning techniques other than deep learning techniques. Table A.2 shows the extracted information from the studies on deep learning techniques for the prediction of breast cancer response.

A.1 state-of-the-art machine learning models

In Table A.1, a summary of the reviewed articles on response prediction using machine learning techniques can be found. Articles running multiple experiments, only the result of the experiment achieving the best performance is included in Table A.1. All results in the table are reported on a validation set. If the article only reports training results, the results are not included in this table. All studies in Table A.1 are retrospective studies.

Looking at Table A.1, it can be seen that most studies are using a heterogeneous dataset. Based on breast cancer subtype, only Banerjee et al. (2017), Braman et al. (2019), and Liu et al. (2019) are using a homogeneous dataset. Cain et al. (2019) used a heterogeneous dataset, including all the three breast cancer subtypes. Apart from this, they also tested their algorithm on a subset containing only HER2+ and TN cancer types, this improved their results from an AUC of 0.59 to an AUC 0.71.

Based on MRI specifications, most studies are using heterogeneous data sets, with MRI images from different manufacturers, different field strengths, and different amount of breast coil channels. Most studies are using MRI scanners with a field strength of 1.5 or 3.0 T. The most used MRI sequence is DCE T1. DCE T1 can give additional information, since this sequence has an extra (time) domain compared to other sequences. Most studies used breast coil channels are 8 or 16 coils.

Most ROI segmentations are done manually by a radiologist. Some studies are using semi-automatic methods, such as a fuzzy c-means algorithm. Giannini et al. (2017) and Wu et al. (2016) are using an automatic and semi-automatic in-house developed algorithm respectively.

Judging on the AUC, the best result is obtained by Xiong et al. (2020) with an AUC of 0.94, but no sensitivity and specificity are reported. Based on sensitivity and specificity, the best results are obtained by Henderson et al. (2017) (sensitivity: 0.875 and specificity: 0.847) and Bian et al. (2020) (sensitivity: 0.882 and specificity: 0.909), with corresponding AUC of 0.845 and 0.91 respectively. Getting close to these results is Zhou et al. (2020) (AUC: 0.888, sensitivity: 0.762, specificity: 0.845) and Chen et al. (2020) (AUC: 0.837, sensitivity: 0.714, specificity: 0.952). The accuracy (0.893) of Chen et al. (2020) is the highest accuracy after Xiong et al. (2020). The best performing classifiers are: Multivariable logistic regression analysis, Mann-Whitney U-test, and random forest algorithm.

A.2 Extracted data literature review

Table A.1: Extracted data from the included studies on machine learning techniques other than a deep learning techniques for the prediction of breast cancer response.

Author (year)	Year	Number of patients	Breast cancer types	Time of MRI	*MRI specifications*					Algorithm	ROI segmentation method	Size ROI	Image pre-processing	Number of features	*Results*			
					Manufacturer	Field strength	Breast coil channels	Sequence	Phase DCE T1 used						Acc.	AUC	Sens.	Spec.
Liu et al. (2019)	2019	414	HR+/HER2-, HER2+, TN	Pre	Philips, Siemens, GE	1.5 T, 3.0 T	Double	T2, DWI, DCE T1	2	MLR analysis	Manually	3D, C	N	13950	-	0.79, 0.71, 0.80[1]	-	-
Eun et al. (2020)	2020	136	HR+, HER2+, TN	Mid	Philips, GE	3.0 T	4	DCE T1	Pre, 1,2,3,4,5	Random Forest	Manually	2D, C	IS	18	0.831	0.82	0.625	0.917
Sutton et al. (2020)	2020	273	HR+/HER2-, HR+/HER2+, HR-/HER2+, TN	Pre, Post	GE	1.5 T, 3.0 T	8, 16	DCE T1	Pre, 1, 2, 3	Random Forest	GCGMM	3D, C	-	255	-	0.83	0.77	0.69
Machireddy et al. (2019)	2019	55	-	Pre, F	Siemens	3.0 T	4	DCE T1	Pre, 2	SVM	Manually	3D, C	-	34	-	0.78	-	-
Xiong et al. (2020)	2020	125	HR+, HER2+, TN	Pre	Philips, GE	1.5 T, 3.0 T	-	T2, DWI, DCE T1	2	MLR analysis	Manually	3D, C	-	1941	0.9355	0.94	-	-
Yoon et al. (2019)	2018	83	HR+, HER2+, TN	Pre	Philips	3.0 T	7	DWI	n/a	ADC ha	Manually	2D, C	-	46	-	-	-	-
Cain et al. (2019)	2019	288	HR+, HER2+, TN	Pre	Siemens, GE	1.5 T, 3.0 T	-	DCE T1	1	SVM	Fuzzy C-means[2] (SA)	3D, C	-	529	-	0.59[3]	-	-
Banerjee et al. (2017)	2017	41	TN	Pre, Post	-	1.5 T, 3.0 T	-	DCE T1	-	SVM	Manually	2D, C	N	442	-	0.85	-	-
Chamming's et al. (2018)	2018	85	HR+, HER2+, TN	Pre	GE	1.5 T	8	T2, DCE T1	1	MWU	Manually	2D, C	SSF (2,4,6)	6	0.58	0.67	0.83	0.49
Giannini et al. (2017)	2017	44	HR+, HER2+, TN	Pre, Mid, Post	GE	1.5 T	8	DCE T1	1	BC	Automatic (in-house)	3D, C	N, IR	27	0.7	-	0.67	0.72
Henderson et al. (2017)	2017	88	HR+, HER2+, TN	Pre, mid	Siemens	3.0 T	7	T2	n/a	MWU	Manually	2D, C	N	1 (image entropy)	0.852	0.845	0.875	0.847
Fan et al. (2017)	2017	103	HR+, HER2+, TN	Pre, Post	Siemens	1.5 T, 3.0 T	8	DCE T1	Pre, 1, 2	MLR analysis	SA	3D, C	-	158	-	0.713	-	-

Continued on next page

Table A.1 – continued from previous page

Author (year)	Year	Number of patients	Breast cancer types	Time of MRI	*MRI specifications*					Algorithm	ROI segmentation method	Size ROI	Image pre-processing	Number of features	*Results*			
					Manufacturer	Field strength	Breast coil channels	Sequence	Phase DCE T1 used						Acc.	AUC	Sens.	Spec.
Braman et al. (2017)	2017	117	HR+, HER2+, TN	Pre	Siemens, Philips	1.5 T, 3.0 T	-	DCE T1	1	DLDA classifier	Manually	3D, C[4]	-	10	0.67	0.74	-	-
Thibault et al. (2017)	2017	38	-	Pre, F, Mid, Post	Siemens	3.0 T	4	T2, DCE T1	2	RR model	Manually	3D, C	-	1043	-	-	-	-
Wu et al. (2016)	2016	35	-	Pre, F, Post	Philips	3.0 T	-	DCE T1	3	LR model	SA (in-house)	3D, C	N	4	-	0.79	0.75	0.78
Braman et al. (2019)	2019	28	HER2+	-	Siemens, Philips	1.5 T, 3.0 T	-	DCE T1	-	DLDA classifier	Manually	3D, C[5]	N	495	0.79	0.80	0.94	0.58
Chen et al. (2020)	2020	91	-	Pre	Siemens	3.0 T	16	DCE T1, ADC map	-	MLR analysis	Manually	3D, C	-	792	0.893	0.837	0.714	0.952
Bian et al. (2020)	2020	152	HR+, HER2+, TN	Pre	GE	3.0 T	8, 16	T2, DWI, DCE T1	1,2,3,4,5,6,7,8	MLR analysis	Manually	2D, C	-	396	0.889	0.91	0.882	0.909
Zhou et al. (2020)	2020	55	HR+, HER2+, TN	Pre	Siemens	3.0 T	16	DCE T1	Pre, 1,2,3,4,5	Random Forest	SA	3D,C	-	616	0.810	0.888	0.762	0.845
Drukker et al. (2019)	2019	158	HR+, HER2+, TN	Pre	Philips	1.5 T, 3.0 T	-	DCE T1	1	LDA classifier	SA	3D, C	-	49	-	-	-	-
Drukker et al. (2018)	2018	162	HR+, HER2+, TN	Pre, F	-	1.5 T	-	DCE T1	Pre, 2,7	CR model	Fuzzy C-means (SA)	3D, C	-	1 (METV)	-	0.72	-	-

Abbreviations: n/a: not applicable, - : not performed/specified
Breast cancer types: HR+: hormone receptor-positive, HR+/HER2-: hormone receptor positive and human epidermal growth factor receptor 2 negative, HER2+: human epidermal growth factor receptor 2 positive, TN: triple negative
Time of MRI: Pre: pre-treatment, Mid: Mid-treatment, Post: Post-treatment, F: after first cycle
Phase DCE T1 used: pre: pre-contrast phase, 1: 1 minute after contrast, 2: 2 minutes after contrast, etc.
Algorithm: MLR: Multivariable logistic regression, SVM: Support Vector Machine, ADC ha: ADC histogram analysis, MWU: Mann-Whitney U-test, BC: Bayesian classifier, DLDA: Diagonal linear discriminant analysis, RR: Ridge regression, LR: logistic regression, LDA: linear discriminant analysis, CR: Cox regression
ROI segmentation method: GCGMM: Grow Cut Gaussian Mixture Model, SA: Semi-automatic
Size ROI: C: Contour of tumor, R: Rectangular box
Image pre-processing: N: normalization, IS: Image Subtraction on all postcontrast phases, SSF (2,4,6): filtering using spatial scaling factor 2, 4 and 6, IR: Image Registration
Number of features: METV: Most enhancing tumor volume
[1] For each of the subgroup (HR+/HER2-, HER2+, TN)
[2] Applying a fussy C-means algorithm inside a manually placed 3D box around the tumor
[3] on a subset of HER2+ and TN tumors an AUC of 0.71 was obtained
[4] Slice with largest tumor area and the two neighbouring slices. Delineation 2.5-5.0 mm around tumor contour.
[5] delineation maximum of 15 mm around tumor contour on 3 adjacent slices.

Table A.2: Extracted data from the included studies on deep learning techniques for the prediction of breast cancer response

Author (year)	Year	Number of patients	Breast cancer types	Time of MRI	*MRI specifications*					Algorithm	ROI segmentation method	Size ROI	Image pre-processing	Number of features	*Results*			
					Manufacturer	Field strength	Breast coil channels	Sequence	Phase DCE T1 used						Acc.	AUC	Sens.	Spec.
Braman et al. (2020)	2020	157	HER2+	Pre	-	1.5 T, 3.0 T	-	DCE T1	Pre, 1,2,3	Multi input CNN	Manually	2D, R	N, Z	Rot, flip, trans, resiz	0.86	0.85	0.81	0.92
El Adoui et al. (2020)	2020	42	HR+, HER2+, TN	Pre, post	Siemens	1.5 T	4	DCE T1	1	Multi input CNN	Manually	2D,R	N, Z	Rot, trans, flip, zoom	0.88	0.91	0.922	0.791
Qu et al. (2020)	2020	302	HER2+, HR+, TN	Pre, post	GE	1.5 T	4	DCE T1	Pre, 1,2,3,4,5	Multi input CNN	Manually	3D, C	N, Z	Rot	-	0.97	0.96	1.00
Ravichandran et al. (2018)	2018	166	HR+, HER2+, TN	Pre	-	1.5 T	4,8	DCE T1	Pre, 1,2	CNN	Thresholding and morph. operations	2D, C	N	Rot, multiple slices/pat	0.85	0.85	-	-
Huynh et al. (2017)	2017	64	-	-	-	-	-	DCE T1	Pre,1,2	CNN (VGGNET)	Manually	2D,R	-	-	-	0.85	-	-
Ha et al. (2019)	2019	141	HR+, HER2+, TN	Pre	GE	1.5 T, 3.0 T	8	DCE T1	1	CNN (VGG 16)	Manually	2D, R	N	Rot. Flip, shear, zoom	0.877	-	0.739	0.951

Abbreviations: n/a: not applicable, - : not performed/specified
Breast cancer types: HR+: hormone receptor-positive, HR+/HER2-: hormone receptor positive and human epidermal growth factor receptor 2 negative, HER2+: human epidermal growth factor receptor 2 positive, TN: triple negative
Time of MRI: Pre: pre-treatment, Mid: Mid-treatment, Post: Post-treatment, F: after first cycle
Phase DCE T1 used: pre: pre-contrast phase, 1: 1 minute after contrast, 2: 2 minutes after contrast, etc.
Algorithm: CNN: Convolutional Neural Network
ROI segmentation method: morph: morphological
Size ROI: C: Contour of tumor, R (x,x): Rectangular box (length, width)
Image pre-processing: N: normalization, Z: zero-padding
Augment. Technique: Rot: Rotating, trans: translating, flip: flipping, resize: resizing, zoom: zooming, multiple slices/pat: multiple slices per patient.

B Appendix: Pseudocode pre-processing methods and data generator

B.1 Pseudocode large ROI dataset

Algorithm B.1: Pseudocode of the data processing resulting in the large ROI dataset

input : X - Dataset containing the DCE T1 volume x_n of each patient n,
L - Containing the information on laterality l_n for each patient
output: X_{train} - training set containing 0.7 of the pre-processed images x'_n,
X_{val} - validation set containing 0.2 of the pre-processed images x'_n,
X_{test} - test set containing 0.1 of the pre-processed images x'_n

```
Load input data;
forall x_n in X do
    if length(x_n) > 200 then
        x_n = x_n[1:3:end];                    // If the MR scan has more than
                                                   200 slices, starting from the
                                                   second slice, every third slice
                                                   is a slice of the first
                                                   contrast phase of the DCE T1
                                                   scan
    end
    if size(x_n) ≠ (200, 400, 400) then
        resize x_n to 200 × 400 × 400 voxels ; // Bicubic interpolation is used
                                                   as interpolation method
    end
    if l_n = 'LEFT' then
        x'_n = x_n[:, 0:200, 200:400] ;        // Volume containing the left
                                                   breast of the patient
    else if l_n = 'RIGHT' then
        x'_n = x_n[:, 0:200, 0:200] ;          // Volume containing the right
                                                   breast of the patient
    end
    x_n = x_n[25 : 175, :, :] ;                // Get slices containing breast
                                                   tissue
    x_n = x_n[0 :: 5, :, :] ;                  // Take every 5th slice, to get a
                                                   total of 30 slices
end
Split X in X_train, X_val and X_test ;         // With ratio (0.7, 0.15, 0.15)
                                                   and while keeping the ratio
                                                   pCR/non-pCR the same over each
                                                   set
Save X_train, X_val and X_test;
```

B.2 Pseudocode small ROI dataset

Algorithm B.2: Pseudocode of the data processing resulting in the small ROI dataset

input : X - Dataset containing the DCE T1 volume x_n of each patient n,
L - Containing the information on laterality l_n for each patient
output: X_{train} - training set containing 0.7 of the pre-processed images x'_n,
X_{val} - validation set containing 0.2 of the pre-processed images x'_n,
X_{test} - test set containing 0.1 of the pre-processed images x'_n

Load input data;
forall x_n *in* X **do**
  if $length(x_n) > 200$ **then**
    $x_n = x_n$[1:3:end]; // If the MR scan has more than 200 slices, starting from the second slice, every third slice is a slice of the first contrast phase of the DCE T1 scan
  end
  if $size(x_n) \neq (200, 400, 400)$ **then**
    resize x_n to $200 \times 400 \times 400$ voxels ; // Bicubic interpolation is used as interpolation method
  end
  if l_n = *'LEFT'* **then**
    $x'_n = x_n$[:, 0:200, 200:400] ; // Volume containing the left breast of the patient
  else if l_n = *'RIGHT'* **then**
    $x'_n = x_n$[:, 0:200, 0:200] ; // Volume containing the right breast of the patient
  end
  $x_n = x_n[25:175, :, :]$; // Get slices containing breast tissue
  $x_n = x_n[0::5, :, :]$; // Take every 5th slice, to get a total of 30 slices
  $coord_{breast} = x_{start}, x_{end}, y_{start}, y_{end}$; // Manually determine the image coordinates defining the box that fits around the breast
  $x_n = x_n[:, x_{start}:x_{end}, y_{start}:y_{end}]$; // Use these coordinates to crop the breast in every slice of the MR volume
end
Split X in X_{train}, X_{val} and X_{test} ; // With ratio (0.7, 0.15, 0.15) and while keeping the ratio pCR/non-pCR the same over each set
Save X_{train}, X_{val} and X_{test};

B.3 Pseudocode data generator

Algorithm B.3: Pseudocode for the data generator used in this project

input : $list_{paths}$ - csv-file containing the paths to the MR volume x_n of each patient n,
$labels$ - dictionary with the labels y_n of each patient n. When calling $labels[path]$, you'll get the corresponding label of these
b_size - Size of the batch,
n_epochs - The number of epochs,
dim - Dimensions of the MR volumes,
$n\_channels$ - Number of channels (1 for grayscale),
aug - Parameter to apply Augmentation (True or False)
output: X_{batch} - The MR volumes of a batch,
y_{batch} - The corresponding labels of the batch

```
n_batches = round down(len(list_paths) / batch_size);        // Determination of the number of batches by '__len__'
indexes = [0,1,2 ... len(list_paths)];
for epoch = 0 to n_epochs do
    for i = 0 to n_batches do
        b_indexes = indexes [i × b_size : (i + 1) × b_size];   // Get the correct patient indexes for the batch
        b_paths = [list_paths [k] for k in b_indexes];         // And get the corresponding paths to the MR volumes
        X_batch = zeros(b_size, dim, n_channels);              // Place to store the MR volumes
        y_batch = zeros(b_size);                               // Place to store the labels
        for j = 0 to b_size do
            x_n = load(b_paths[j]);                            // '__data_generation' function is called to load the MR volumes...
            if aug == True then
                x_n = augment x_n ;                            // ...and to apply the augmentation methods
            end
            x_n = normalize(x_n);                              // scale the voxel values between 0 and 1
            X_batch[j] = x_n ;                                 // Store the MR volume
            y_batch[j] = labels[batch_path[j]] ;               // Store the corresponding label
        end
        return X_batch, y_batch;                               // The '__getitem__' function returns the MR volumes and corresponding labels for the batch. The '__getitem__' function is called for every batch, over all epochs
    end
    indexes = shuffle(indexes);                                // At the end of every epoch, patients indexes are shuffled by 'on_epoch_end' function
end
```

C Appendix: Results

This appendix shows the model architectures in Section C.1. Section C.2 shows the ROC curves of each model of each fold (with corresponding mean ROC curve) of the 10-fold cross validation.

C.1 Model architectures

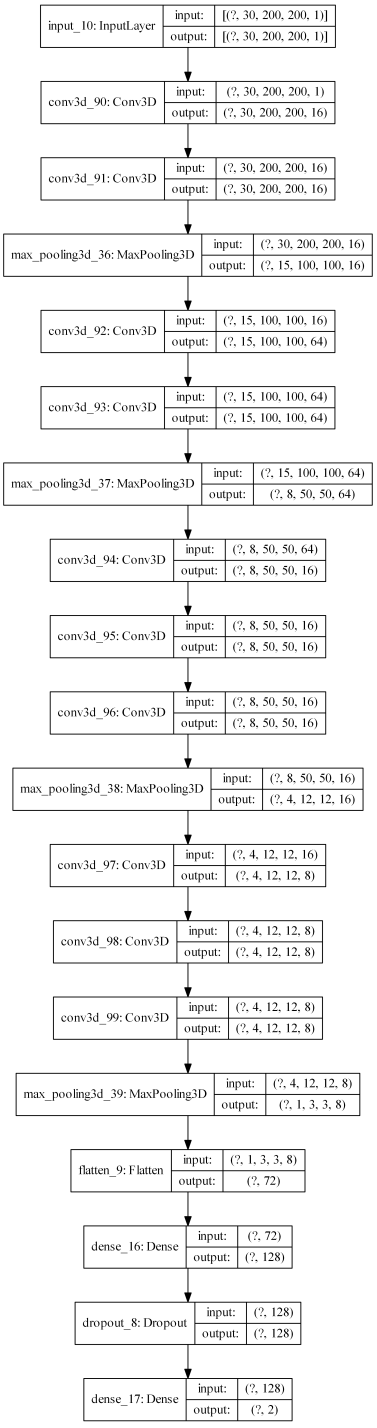


Figure C.1: The architecture of the single-input model for the dataset containing the large ROIs. The input size of this model for the MR volume is $200 \times 200 \times 30$

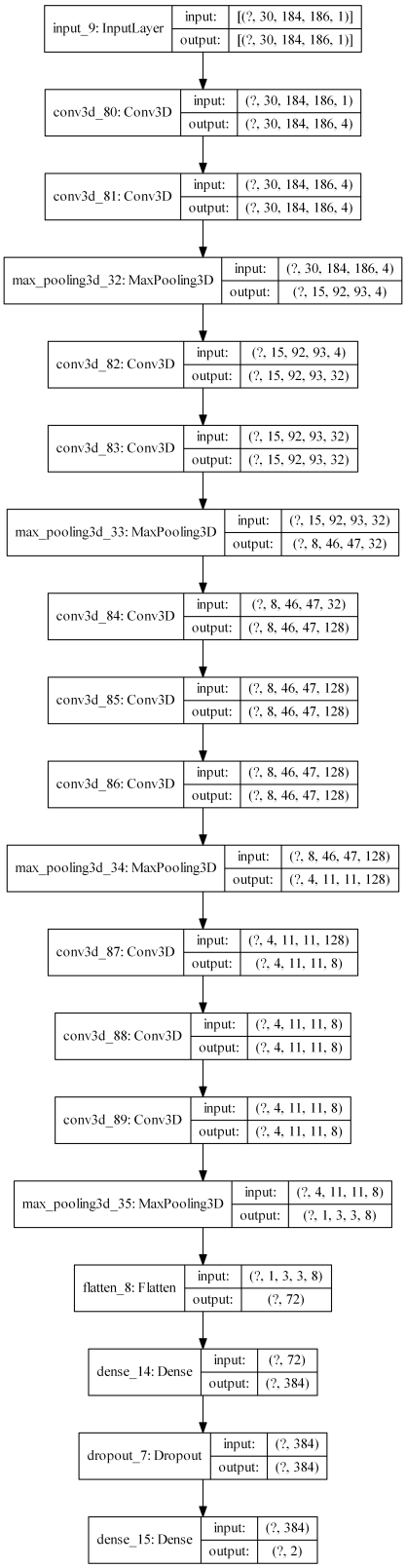


Figure C.2: The architecture of the single-input model for the dataset containing the small ROIs. The input size of this model for the MR volume is $184 \times 186 \times 30$

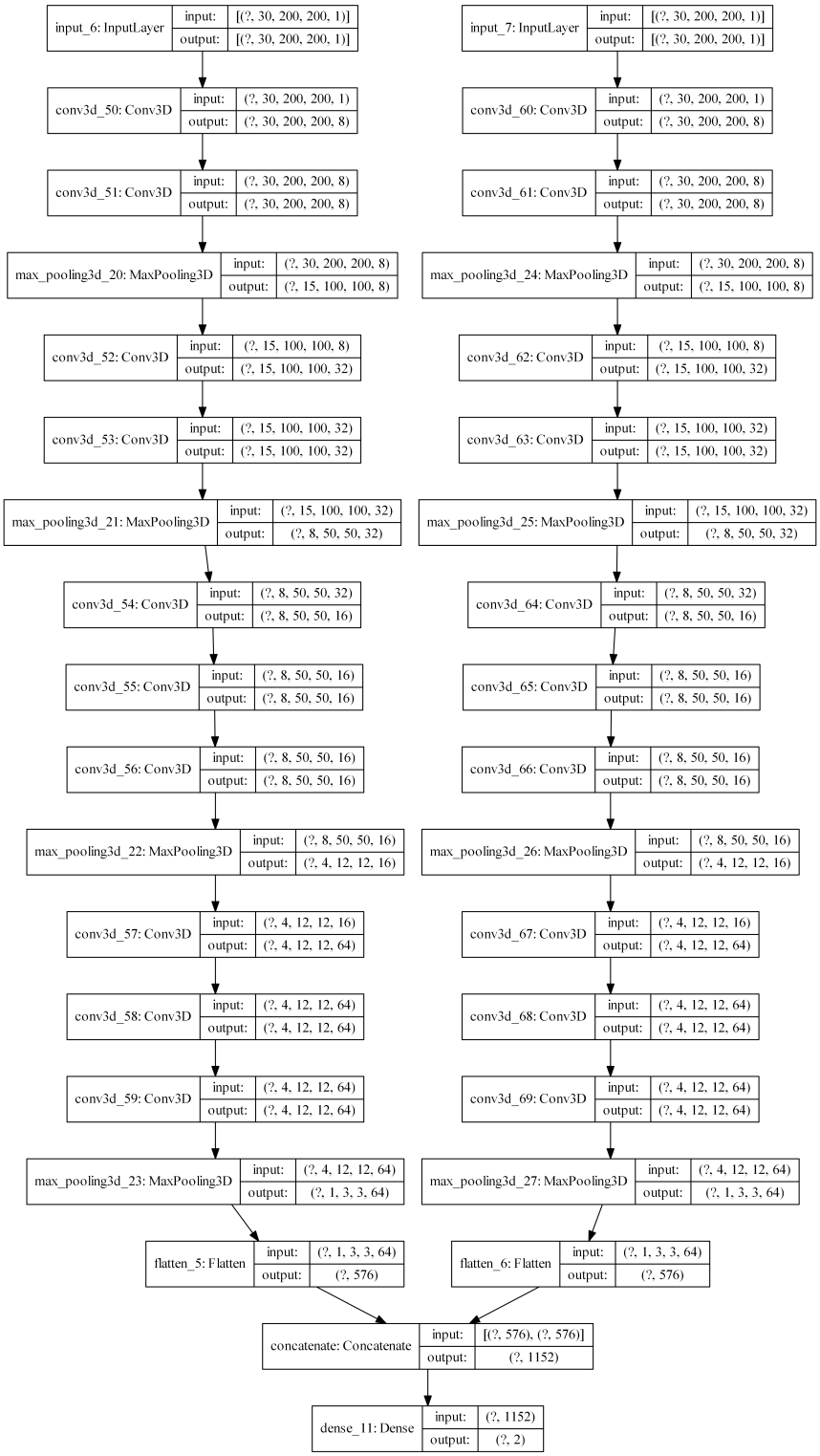


Figure C.3: The architecture of the double input model for the dataset containing the large ROIs. The input size of this model for the MR volumes are 200 × 200 × 30

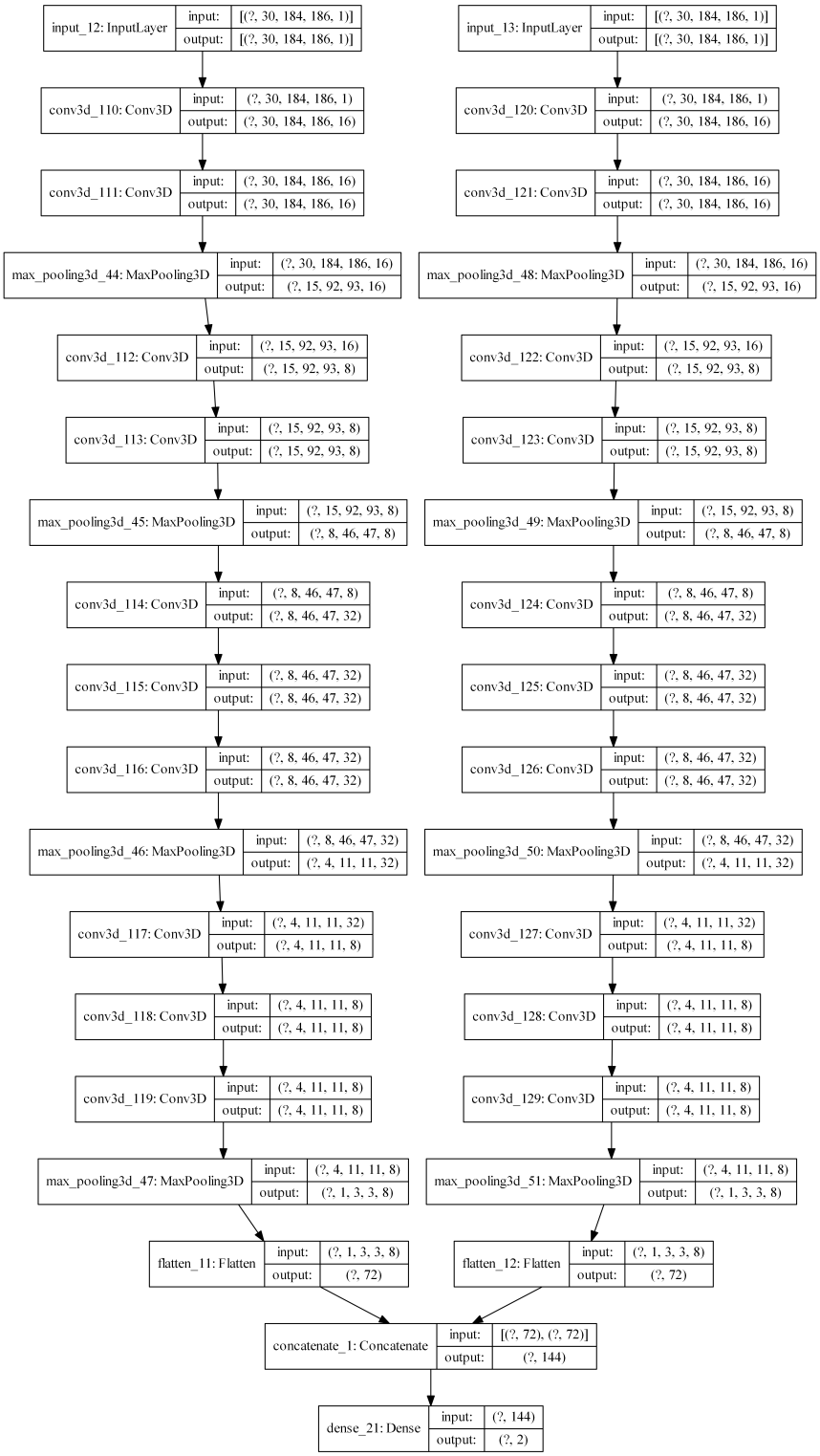


Figure C.4: The architecture of the double input model for the dataset containing the small ROIs. The input size of this model for the MR volumes are 184 × 186 × 30

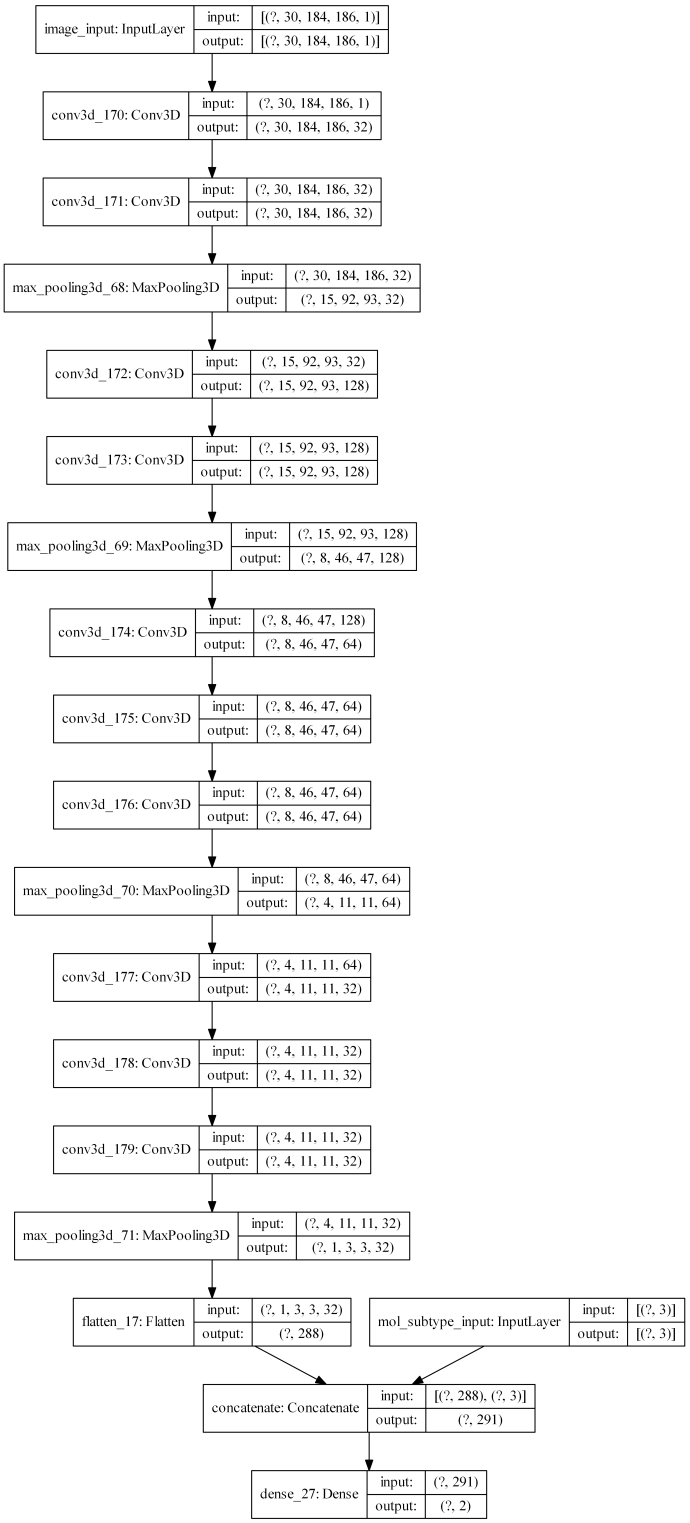


Figure C.5: The architecture of the single-input model for the dataset containing the small ROIs with extra input for the molecular subtype. The input size of this model for the MR volume is $184 \times 186 \times 30$

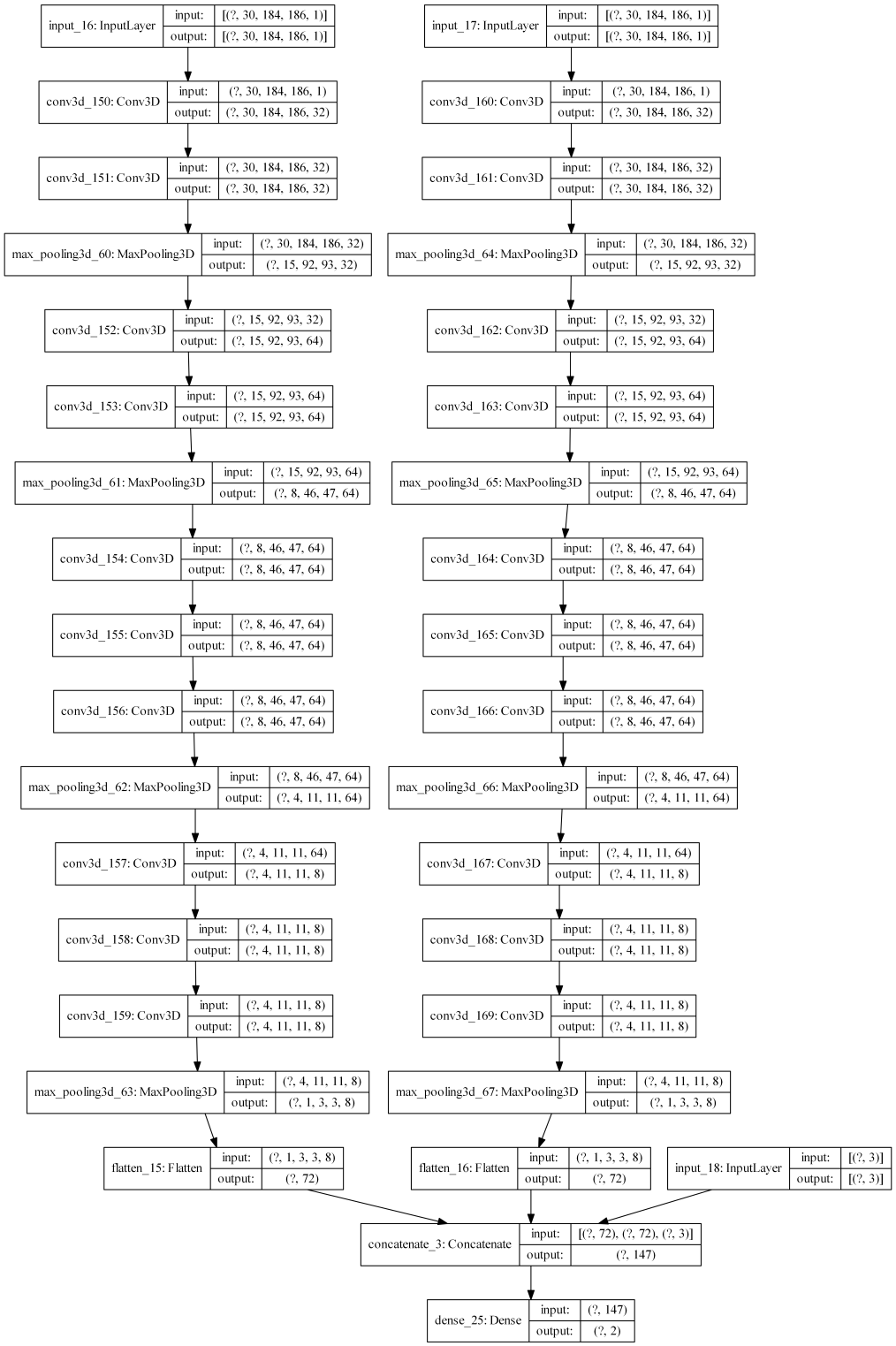


Figure C.6: The architecture of the double-input model for the dataset containing the small ROIs with extra input for the molecular subtype. The input size of this model for the MR volume is 184 × 186 × 30

C.2 10-fold cross validation

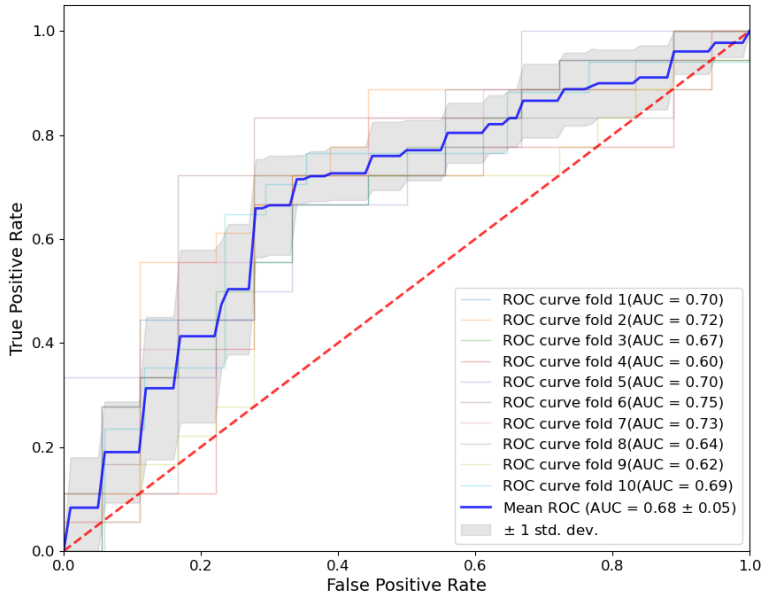


Figure C.7: The ROC for each fold of the 10-fold cross validation and the resulting mean ROC for the double input CNN (Large ROI). The mean AUC and corresponding standard deviation is 0.68 ± 0.05

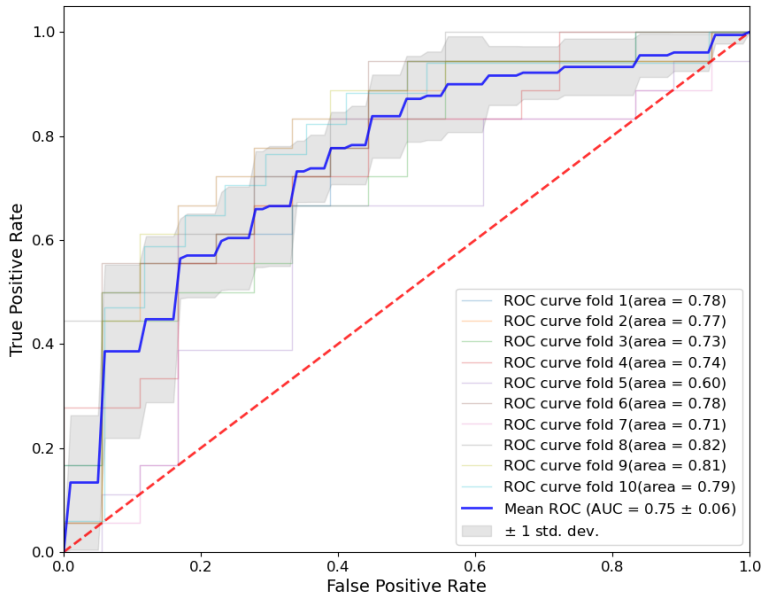


Figure C.8: The ROC for each fold of the 10-fold cross validation and the resulting mean ROC for the single-input CNN (small ROI). The mean AUC and corresponding standard deviation is 0.75 ± 0.06

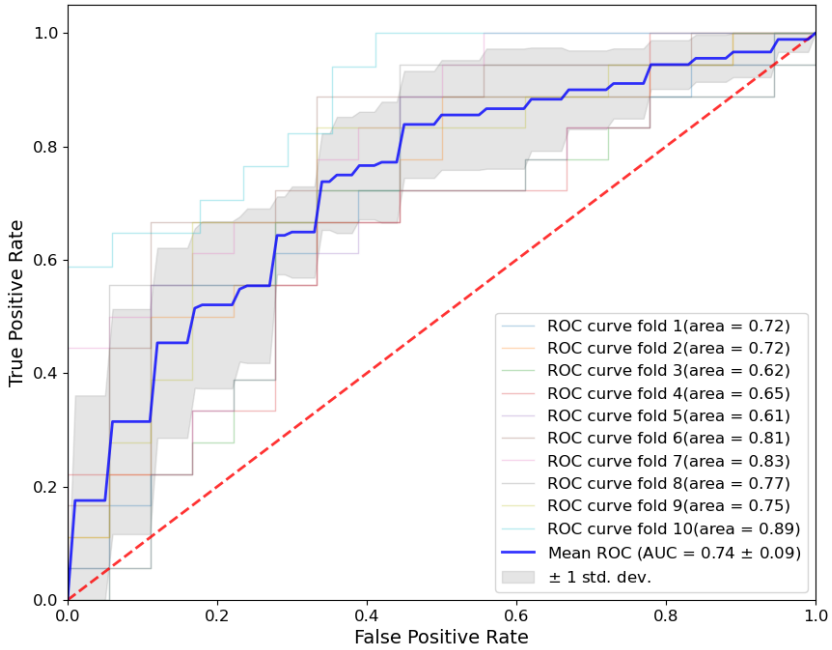


Figure C.9: The ROC for each fold of the 10-fold cross validation and the resulting mean ROC for the single-input CNN (small ROI). The mean AUC and corresponding standard deviation is 0.74 ± 0.09

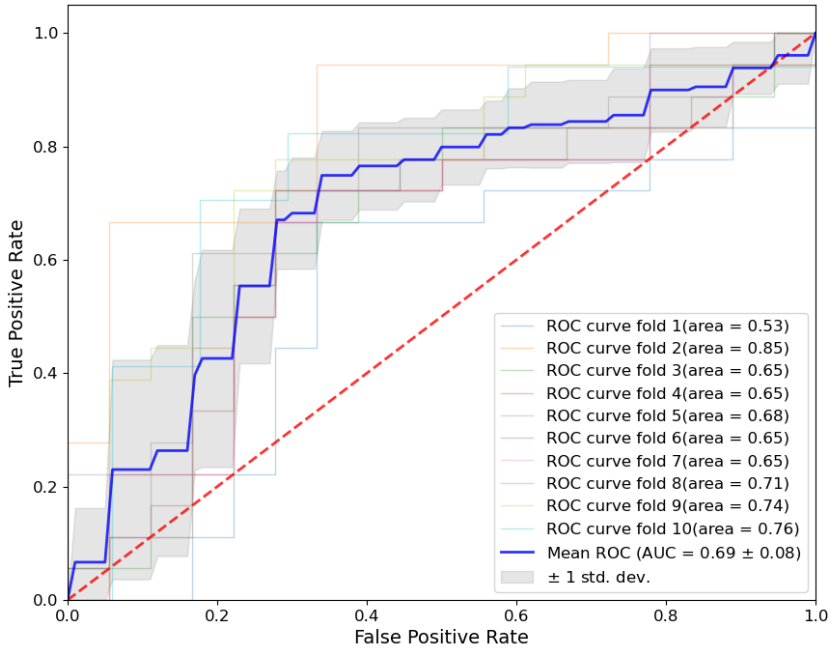


Figure C.10: The ROC for each fold of the 10-fold cross validation and the resulting mean ROC for the double input CNN (small ROI). The mean AUC and corresponding standard deviation is 0.69 ± 0.08

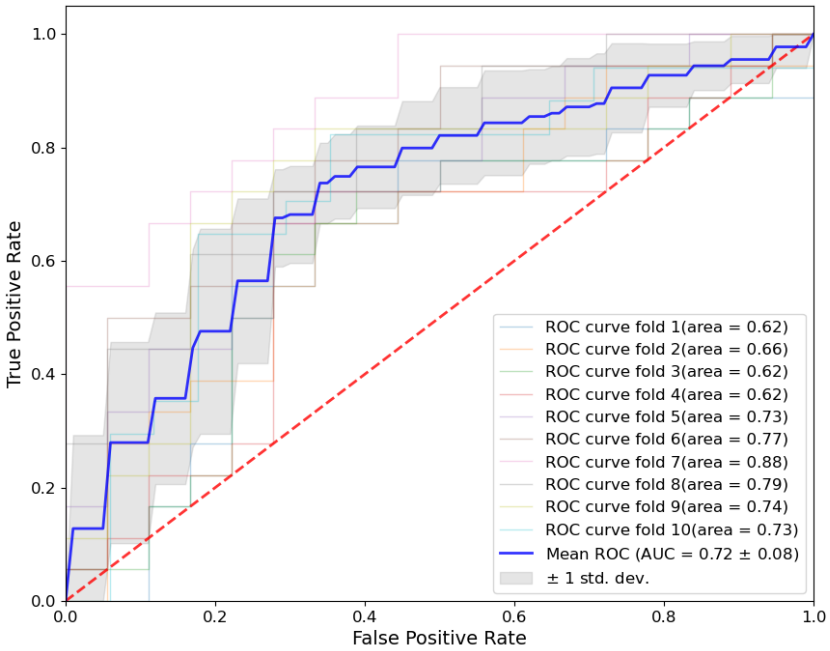


Figure C.11: The ROC for each fold of the 10-fold cross validation and the resulting mean ROC for the single-input CNN (small ROI) with the molecular subtype as extra input. The mean AUC and corresponding standard deviation is 0.72 ± 0.08

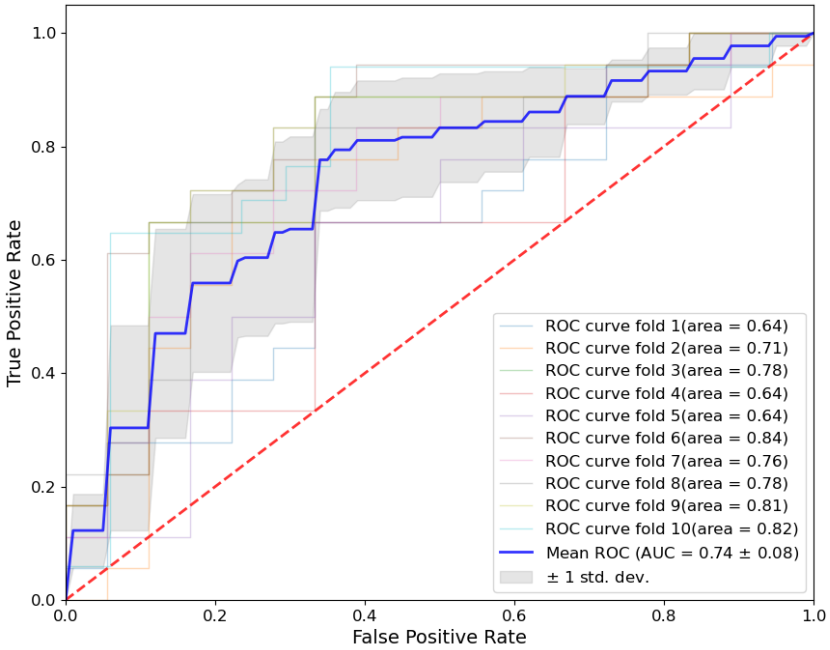


Figure C.12: The ROC for each fold of the 10-fold cross validation and the resulting mean ROC for the double input CNN (small ROI) with the molecular subtype as extra input. The mean AUC and corresponding standard deviation is 0.74 ± 0.08

Bibliography

Abadi, M., A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu and X. Zheng (2015), TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, software available from tensorflow.org.
https://www.tensorflow.org/

Alberts, B., D. Bray, K. Hopkin, A. Johnson, J. Lewis, M. Raff, K. Roberts and P. Walter (2014), Transport Across Cell Membranes, in *Essential Cell Biology*, Garland Science, Taylor Francis Group, chapter 12, p. 727, 4h edition, ISBN 978-0-8153-455-1.

Banerjee, I., S. Malladi, D. Lee, A. Depeursinge, M. Telli and J. Lipson (2017), Assessing treatment response in triple-negative breast cancer from quantitative image analysis in perfusion magnetic resonance imaging, **vol. 5**, no.01, p. 1, ISSN 2329-4302, doi:10.1117/1.jmi.5.1.011008.

Bian, T., Z. Wu, Q. Lin, H. Wang, Y. Ge, S. Duan, G. Fu, C. Cui and X. Su (2020), Radiomic signatures derived from multiparametric MRI for the pretreatment prediction of response to neoadjuvant chemotherapy in breast cancer, **vol. 93**, no.1115, p. 20200287, ISSN 1748880X, doi:10.1259/bjr.20200287.
https://pubmed.ncbi.nlm.nih.gov/32822542/

Braman, N., M. E. Adoui, M. Vulchi, P. Turk, M. Etesami, P. Fu, K. Bera, S. Drisis, V. Varadan, D. Plecha, M. Benjelloun, J. Abraham and A. Madabhushi (2020), Deep learning-based prediction of response to HER2-targeted neoadjuvant chemotherapy from pre-treatment dynamic breast MRI: A multi-institutional validation study, *arXiv*.
http://arxiv.org/abs/2001.08570

Braman, N., P. Prasanna, J. Whitney, S. Singh, N. Beig, M. Etesami, D. D. Bates, K. Gallagher, B. N. Bloch, M. Vulchi, P. Turk, K. Bera, J. Abraham, W. M. Sikov, G. Somlo, L. N. Harris, H. Gilmore, D. Plecha, V. Varadan and A. Madabhushi (2019), Association of Peritumoral Radiomics With Tumor Biology and Pathologic Response to Preoperative Targeted Therapy for HER2 (ERBB2)-Positive Breast Cancer, **vol. 2**, no.4, p. e192561, ISSN 25743805, doi:10.1001/jamanetworkopen.2019.2561.
https://pubmed.ncbi.nlm.nih.gov/31002322/

Braman, N. M., M. Etesami, P. Prasanna, C. Dubchuk, H. Gilmore, P. Tiwari, D. Pletcha and A. Madabhushi (2017), Intratumoral and peritumoral radiomics for the pretreatment prediction of pathological complete response to neoadjuvant chemotherapy based on breast DCE-MRI, **vol. 19**, no.1, p. 57, ISSN 1465542X, doi:10.1186/s13058-017-0846-1.
https://breast-cancer-research.biomedcentral.com/articles/10.1186/s13058-017-0846-1

Breast Cancer Foundation NZ (2021), DCIS (Ductal Carcinoma in Situ) (accessed: 06-08-2021).
https://www.breastcancerfoundation.org.nz/breast-cancer/types-of-breast-cancer/pre-invasive

Cain, E. H., A. Saha, M. R. Harowicz, J. R. Marks, P. K. Marcom and M. A. Mazurowski (2019), Multivariate machine learning models for prediction of pathologic response to neoadjuvant therapy in breast cancer using MRI features: a study using an independent validation set, **vol. 173**, no.2, pp. 455–463, ISSN 15737217, doi:10.1007/s10549-018-4990-9.
https://doi.org/10.1007/s10549-018-4990-9

Cawley, G. C. and N. L. Talbot (2010), On over-fitting in model selection and subsequent selection bias in performance evaluation, *Journal of Machine Learning Research*, **vol. 11**, pp. 2079–2107, ISSN 15324435.

Chamming's, F., Y. Ueno, R. Ferré, E. Kao, A. S. Jannot, J. Chong, A. Omeroglu, B. Mesurolle, C. Reinhold and B. Gallix (2018), Features from computerized texture analysis of breast cancers at pretreatment MR imaging are associated with response to neoadjuvant chemotherapy, **vol. 286**, no.2, pp. 412–420, ISSN 15271315, doi:10.1148/radiol.2017170143.

Charniak, E. (2018), *Introduction to deep learning*, Mit Press Ltd., ISBN 978-0-262-03951-2.

Chen, X., X. Chen, J. Yang, Y. Li, W. Fan and Z. Yang (2020), Combining Dynamic Contrast-Enhanced Magnetic Resonance Imaging and Apparent Diffusion Coefficient Maps for a Radiomics Nomogram to Predict Pathological Complete Response to Neoadjuvant Chemotherapy in Breast Cancer Patients, **vol. 44**, no.2, pp. 275–283, ISSN 15323145, doi:10.1097/RCT.0000000000000978.
https://pubmed.ncbi.nlm.nih.gov/32004189/

Choi, M., Y. H. Park, J. S. Ahn, Y. H. Im, S. J. Nam, S. Y. Cho and E. Y. Cho (2017), Evaluation of Pathologic Complete Response in Breast Cancer Patients Treated with Neoadjuvant Chemotherapy: Experience in a Single Institution over a 10-Year Period, **vol. 51**, no.1, pp. 69–78, ISSN 2383-7837, doi:10.4132/JPTM.2016.10.05.
https://pubmed.ncbi.nlm.nih.gov/28013533/

Chollet, F. (2015), keras, https://github.com/fchollet/keras.

Cortazar, P., L. Zhang, M. Untch, K. Mehta, J. P. Costantino, N. Wolmark, H. Bonnefoi, D. Cameron, L. Gianni, P. Valagussa, S. M. Swain, T. Prowell, S. Loibl, D. L. Wickerham, J. Bogaerts, J. Baselga, C. Perou, G. Blumenthal, J. Blohmer, E. P. Mamounas, J. Bergh, V. Semiglazov, R. Justice, H. Eidtmann, S. Paik, M. Piccart, R. Sridhara, P. A. Fasching, L. Slaets, S. Tang, B. Gerber, C. E. Geyer, R. Pazdur, N. Ditsch, P. Rastogi, W. Eiermann and G. Von Minckwitz (2014), Pathological complete response and long-term clinical benefit in breast cancer: The CTNeoBC pooled analysis, **vol. 384**, no.9938, pp. 164–172, ISSN 1474547X, doi:10.1016/S0140-6736(13)62422-8.
https://pubmed.ncbi.nlm.nih.gov/24529560/

Drukker, K., A. Edwards, C. Doyle, J. Papaioannou, K. Kulkarni and M. L. Giger (2019), Breast MRI radiomics for the pretreatment prediction of response to neoadjuvant chemotherapy in node-positive breast cancer patients, **vol. 6**, no.03, p. 1, ISSN 2329-4302, doi:10.1117/1.jmi.6.3.034502.

Drukker, K., H. Li, N. Antropova, A. Edwards, J. Papaioannou and M. L. Giger (2018), Most-enhancing tumor volume by MRI radiomics predicts recurrence-free survival "early on" in neoadjuvant treatment of breast cancer, **vol. 18**, no.1, ISSN 14707330, doi:10.1186/s40644-018-0145-9.

Eisenhauer, E. A., P. Therasse, J. Bogaerts, L. H. Schwartz, D. Sargent, R. Ford, J. Dancey, S. Arbuck, S. Gwyther, M. Mooney, L. Rubinstein, L. Shankar, L. Dodd, R. Kaplan, D. Lacombe and J. Verweij (2009), New response evaluation criteria in solid tumours: revised RECIST guideline (version 1.1), **vol. 45**, no.2, pp. 228–247, ISSN 1879-0852, doi:10.1016/J.EJCA.2008.10.026.
https://pubmed.ncbi.nlm.nih.gov/19097774/

El Adoui, M., S. Drisis and M. Benjelloun (2020), Multi-input deep learning architecture for predicting breast tumor response to chemotherapy using quantitative MR images, **vol. 15**, no.9, pp. 1491–1500, ISSN 18616429, doi:10.1007/s11548-020-02209-9.
https://pubmed.ncbi.nlm.nih.gov/32556920/

Eun, N. L., D. Kang, E. J. Son, J. S. Park, J. H. Youk, J. A. Kim and H. M. Gweon (2020), Texture analysis with 3.0-T MRI for association of response to neoadjuvant chemotherapy in breast

cancer, **vol. 294**, no.1, pp. 31–41, ISSN 15271315, doi:10.1148/radiol.2019182718. https://pubmed.ncbi.nlm.nih.gov/31769740/

Fan, M., G. Wu, H. Cheng, J. Zhang, G. Shao and L. Li (2017), Radiomic analysis of DCE-MRI for prediction of response to neoadjuvant chemotherapy in breast cancer patients, *European Journal of Radiology*, **vol. 94**, pp. 140–147, ISSN 18727727, doi:10.1016/j.ejrad.2017.06.019.

Gabriel, A. and G. P. Maxwell (2020), *Anatomy of the Breast*, Springer International Publishing, Cham, pp. 1–10, ISBN 978-3-030-48226-8, doi:10.1007/978-3-030-48226-8_1. https://doi.org/10.1007/978-3-030-48226-8_1

Gampenrieder, S. P., A. Peer, C. Weismann, M. Meissnitzer, G. Rinnerthaler, J. Webhofer, T. Westphal, M. Riedmann, T. Meissnitzer, H. Egger, F. K. Federspiel, R. Reitsamer, C. Hauser-Kronberger, K. Stering, K. Hergan, B. Mlineritsch and R. Greil (2019), Radiologic complete response (rCR) in contrast-enhanced magnetic resonance imaging (CE-MRI) after neoadjuvant chemotherapy for early breast cancer predicts recurrence-free survival but not pathologic complete response (pCR), **vol. 21**, no.1, pp. 1–11, ISSN 1465-542X, doi:10.1186/S13058-018-1091-Y. https://breast-cancer-research.biomedcentral.com/articles/10.1186/s13058-018-1091-y

Giannini, V., S. Mazzetti, A. Marmo, F. Montemurro, D. Regge and L. Martincich (2017), A computer-aided diagnosis (CAD) scheme for pretreatment prediction of pathological response to neoadjuvant therapy using dynamic contrast-enhanced MRI texture features, **vol. 90**, no.1077, p. 20170269, ISSN 0007-1285, doi:10.1259/bjr.20170269. http://www.birpublications.org/doi/10.1259/bjr.20170269

Gomes Do Nascimento, R. and K. M. Otoni (2020), Histological and molecular classification of breast cancer: what do we know?, *Mastology*, **vol. 30**, p. 20200024, doi:10.29289/25945394202020200024.

Goodfellow, I., Y. Bengio and A. Courville (2016), *Deep Learning*, MIT Press, http://www.deeplearningbook.org.

Ha, R., C. Chin, J. Karcich, M. Z. Liu, P. Chang, S. Mutasa, E. Pascual Van Sant, R. T. Wynn, E. Connolly and S. Jambawalikar (2019), Prior to Initiation of Chemotherapy, Can We Predict Breast Tumor Response? Deep Learning Convolutional Neural Networks Approach Using a Breast MRI Tumor Dataset, **vol. 32**, no.5, pp. 693–701, ISSN 1618727X, doi:10.1007/s10278-018-0144-1. https://pubmed.ncbi.nlm.nih.gov/30361936/

Haenlein, M. and A. Kaplan (2019), A Brief History of Artificial Intelligence: On the Past, Present, and Future of Artificial Intelligence, **vol. 61**, no.4, pp. 5–14, ISSN 0008-1256, doi:10.1177/0008125619864925. http://journals.sagepub.com/doi/10.1177/0008125619864925

Heil, J., S. Kü Mmel, B. Schaefgen, S. Paepke, C. Thomssen, G. Rauch, B. Ataseven, R. Groe, V. Dreesmann, T. Kü Hn, S. Loibl, J.-U. Blohmer and G. Von Minckwitz (2015), Diagnosis of pathological complete response to neoadjuvant chemotherapy in breast cancer by minimal invasive biopsy techniques, doi:10.1038/bjc.2015.381. www.bjcancer.com

Heil, J., B. Schaefgen, P. Sinn, H. Richter, A. Harcos, C. Gomez, A. Stieber, A. Hennigs, G. Rauch, F. Schuetz, C. Sohn, A. Schneeweiss and M. Golatta (2016), Can a pathological complete response of breast cancer after neoadjuvant chemotherapy be diagnosed by minimal invasive biopsy?, *European Journal of Cancer*, **vol. 69**, pp. 142–150, ISSN 18790852, doi:10.1016/j.ejca.2016.09.034.

Henderson, S., C. Purdie, C. Michie, A. Evans, R. Lerski, M. Johnston, S. Vinnicombe and A. M. Thompson (2017), Interim heterogeneity changes measured using entropy texture features

on T2-weighted MRI at 3.0 T are associated with pathological response to neoadjuvant chemotherapy in primary breast cancer, **vol. 27**, no.11, pp. 4602–4611, ISSN 14321084, doi:10.1007/s00330-017-4850-8.
https://link.springer.com/article/10.1007/s00330-017-4850-8

Huynh, B. Q., N. Antropova and M. L. Giger (2017), Comparison of breast DCE-MRI contrast time points for predicting response to neoadjuvant chemotherapy using deep convolutional neural network features with transfer learning, in *Medical Imaging 2017: Computer-Aided Diagnosis*, volume 10134, Eds. S. G. Armato and N. A. Petrick, SPIE, p. 101340U, doi:10.1117/12.2255316.
http://proceedings.spiedigitallibrary.org/proceeding.aspx?doi=10.1117/12.2255316

IBM Corp. (2019), IBM SPSS Statistics for Windows, Version 26.0, Armonk, NY: IBM Corp.

Irish Cancer Society (2021), Lobular carcinoma in situ (LCIS) (accessed: 06-08-2021).
https://www.cancer.ie/cancer-information-and-support/cancer-types/breast-cancer/types-of-breast-cancer/lobular-carcinoma-in-situ-lcis

Kaufmann, M., G. N. Hortobagyi, A. Goldhirsch, S. Scholl, A. Makris, P. Valagussa, J. U. Blohmer, W. Eiermann, R. Jackesz, W. Jonat, A. Lebeau, S. Loibl, W. Miller, S. Seeber, V. Semiglazov, R. Smith, R. Souchon, V. Stearns, M. Untch and G. Von Minckwitz (2006), Recommendations from an international expert panel on the use of neoadjuvant (primary) systemic treatment of operable breast cancer: An update, doi:10.1200/JCO.2005.02.6187.
https://pubmed.ncbi.nlm.nih.gov/16622270/

Kong, X., M. S. Moran, N. Zhang, B. Haffty and Q. Yang (2011), Meta-analysis confirms achieving pathological complete response after neoadjuvant chemotherapy predicts favourable prognosis for breast cancer patients, **vol. 47**, no.14, pp. 2084–2090, ISSN 09598049, doi:10.1016/j.ejca.2011.06.014.
https://pubmed.ncbi.nlm.nih.gov/21737257/

Kumar, P. and M. Clark (2012), Malignant disease, in *Kumar Clark's Clinical Medicine*, Eds. C. Callagher, T. Lister and M. Smith, Elsevier Ltd, chapter 9, pp. 473–476, eighth edition, ISBN 978-0-7020-4499-1.

LeCun, Y., Y. Bengio and G. Hinton (2015), Deep learning, **vol. 521**, no.7553, pp. 436–444, ISSN 1476-4687, doi:10.1038/NATURE14539.
https://pubmed.ncbi.nlm.nih.gov/26017442/

Li, L. T., G. Jiang, Q. Chen and J. N. Zheng (2015), Predic Ki67 is a promising molecular target in the diagnosis of cancer (Review), **vol. 11**, no.3, pp. 1566–1572, ISSN 17913004, doi:10.3892/mmr.2014.2914.

Liu, S. V., L. Melstrom, K. Yao, C. A. Russell and S. F. Sener (2010), Neoadjuvant therapy for breast cancer, **vol. 101**, no.4, pp. 283–291, ISSN 00224790, doi:10.1002/JSO.21446/FORMAT/PDF.

Liu, Z., Z. Li, J. Qu, R. Zhang, X. Zhou, L. Li, K. Sun, Z. Tang, H. Jiang, H. Li, Q. Xiong, Y. Ding, X. Zhao, K. Wang, Z. Liu and J. Tian (2019), Radiomics of multiparametric MRI for pretreatment prediction of pathologic complete response to neoadjuvant chemotherapy in breast cancer: A multicenter study, **vol. 25**, no.12, pp. 3538–3547, ISSN 15573265, doi:10.1158/1078-0432.CCR-18-3190.
https://pubmed.ncbi.nlm.nih.gov/30842125/

Machireddy, A., G. Thibault, A. Tudorica, A. Afzal, M. Mishal, K. Kemmer, A. Naik, M. Troxell, E. Goranson, K. Oh, N. Roy, N. Jafarian, M. Holtorf, W. Huang and X. Song (2019), Early Prediction of Breast Cancer Therapy Response using Multiresolution Fractal Analysis of DCE-MRI Parametric Maps, **vol. 5**, no.1, pp. 90–98, ISSN 2379139X,

doi:10.18383/j.tom.2018.00046.
https://pubmed.ncbi.nlm.nih.gov/30854446/

Mann, R. M., C. K. Kuhl, K. Kinkel and C. Boetes (2008), Breast MRI: Guidelines from the European Society of Breast Imaging, **vol. 18**, no.7, pp. 1307–1318, ISSN 09387994, doi:10.1007/s00330-008-0863-7.
/pmc/articles/PMC2441490//pmc/articles/PMC2441490/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC2441490/

McCorduck, P. (2004), *Machines Who Think*, ISBN 1568812051, doi:10.1126/science.254.5036.1291-a.

McReynolds, T. and D. Blythe (2005), Image Processing Techniques, in *Advanced Graphics Programming Using OpenGL*, Morgan Kaufmann, chapter 12, pp. 211–245, ISBN 978-1-55860-659-3, doi:10.1016/B978-155860659-3.50014-7.

Meretoja, T. J., M. H. Leidenius, T. Tasmuth, R. Sipilä and E. Kalso (2014), Pain at 12 months after surgery for breast cancer, doi:10.1001/jama.2013.278795.

Meyer-Baese, A. and V. Schmid (2014), Computer-Aided Diagnosis for Diagnostically Challenging Breast Lesions in DCE-MRI, in *Pattern Recognition and Signal Analysis in Medical Imaging*, chapter 13, pp. 391–420, ISBN 9780124095458, doi:10.1016/b978-0-12-409545-8.00013-3.

Miller, K. D., L. Nogueira, A. B. Mariotto, J. H. Rowland, K. R. Yabroff, C. M. Alfano, A. Jemal, J. L. Kramer and R. L. Siegel (2019), Cancer treatment and survivorship statistics, 2019, Wiley, pp. 363–385, ISSN 0007-9235, doi:10.3322/caac.21565.
https://onlinelibrary.wiley.com/doi/abs/10.3322/caac.21565

Montemurro, F., I. Nuzzolese and R. Ponzone (2020), Neoadjuvant or adjuvant chemotherapy in early breast cancer?, doi:10.1080/14656566.2020.1746273.
https://pubmed.ncbi.nlm.nih.gov/32237920/

Newitt, D. and N. Hylton (2016), Single site breast DCE-MRI data and segmentations from patients undergoing neoadjuvant chemotherapy, doi:10.7937/K9/TCIA.2016.QHsyhJKy.
https://wiki.cancerimagingarchive.net/display/Public/Breast-MRI-NACT-Pilot#22513764fb2b62ab82ef47da8b39106de8b728bd

Nielsen, M. A. (2015), Using neural nets to recognize handwritte digits, in *Neural Networks and Deep Learning*, Determination Press, chapter 1.
http://neuralnetworksanddeeplearning.com

Parham, P. (2015), *The Immune System*, Garland Science, Taylor Francis Group, chapter 17, pp. 510–511, 4th edition, ISBN 978-0-8153-4466-7.

Prevos, R., M. L. Smidt, V. C. Tjan-Heijnen, M. Van Goethem, R. G. Beets-Tan, J. E. Wildberger and M. B. Lobbes (2012), Pre-treatment differences and early response monitoring of neoadjuvant chemotherapy in breast cancer patients using magnetic resonance imaging: A systematic review, **vol. 22**, no.12, pp. 2607–2616, ISSN 09387994, doi:10.1007/s00330-012-2653-5.
https://link.springer.com/article/10.1007/s00330-012-2653-5

Qu, Y. H., H. T. Zhu, K. Cao, X. T. Li, M. Ye and Y. S. Sun (2020), Prediction of pathological complete response to neoadjuvant chemotherapy in breast cancer using a deep learning (DL) method, **vol. 11**, no.3, pp. 651–658, ISSN 17597714, doi:10.1111/1759-7714.13309.
https://pubmed.ncbi.nlm.nih.gov/31944571/

Rapoport, B. L., G. S. Demetriou, S. D. Moodley and C. A. Benn (2014), When and how do i use neoadjuvant chemotherapy for breast cancer?, **vol. 15**, no.1, pp. 86–98, ISSN 15346277, doi:10.1007/s11864-013-0266-0.
https://pubmed.ncbi.nlm.nih.gov/24306808/

Ravichandran, K., N. Braman, A. Janowczyk and A. Madabhushi (2018), A deep learning classifier for prediction of pathological complete response to neoadjuvant chemotherapy from baseline breast DCE-MRI, SPIE-Intl Soc Optical Eng, p. 11, ISBN 9781510616394, ISSN 16057422, doi:10.1117/12.2294056.

Reinisch, M., G. von Minckwitz, N. Harbeck, W. Janni, S. Kümmel, M. Kaufmann, D. Elling, V. Nekljudova and S. Loibl (2013), Side Effects of Standard Adjuvant and Neoadjuvant Chemotherapy Regimens According to Age Groups in Primary Breast Cancer, **vol. 8**, no.1, pp. 60–66, ISSN 1661-3805, doi:10.1159/000346834.
https://www.karger.com/Article/FullText/346834

Rouzier, R., C. M. Perou, W. F. Symmans, N. Ibrahim, M. Cristofanilli, K. Anderson, K. R. Hess, J. Stec, M. Ayers, P. Wagner, P. Morandi, C. Fan, I. Rabiul, J. S. Ross, G. N. Hortobagyi and L. Pusztai (2005), Breast cancer molecular subtypes respond differently to preoperative chemotherapy, **vol. 11**, no.16, pp. 5678–5685, ISSN 10780432, doi:10.1158/1078-0432.CCR-04-2421.
https://pubmed.ncbi.nlm.nih.gov/16115903/

Rubin, E. and H. M. Reisner (2014), The Breast, in *Essentials of Rubin's Pathology*, Eds. A. M. Mulligan and F. P. O'Malley, Wolters Kluwer, chapter 19, pp. 531–541, 6th edition, ISBN 978-1-4511-1023-4.

Selvaraju, R. R., M. Cogswell, A. Das, R. Vedantam, D. Parikh and D. Batra (2019), Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization, **vol. 128**, no.2, p. 336–359, ISSN 1573-1405, doi:10.1007/s11263-019-01228-7.
http://dx.doi.org/10.1007/s11263-019-01228-7

Subbiah, S., G. Gopu, P. Senthilkumar and P. Muniasamy (2017), Molecular subtypes as a predictor of response to neoadjuvant chemotherapy in breast cancer patients, **vol. 54**, no.4, pp. 652–657, ISSN 19984774, doi:10.4103/ijc.IJC_238_17.
https://pubmed.ncbi.nlm.nih.gov/30082552/

Sung, H., J. Ferlay, R. L. Siegel, M. Laversanne, I. Soerjomataram, A. Jemal and F. Bray (2021), Global Cancer Statistics 2020: GLOBOCAN Estimates of Incidence and Mortality Worldwide for 36 Cancers in 185 Countries, **vol. 71**, no.3, pp. 209–249, ISSN 1542-4863, doi:10.3322/CAAC.21660.
https://onlinelibrary.wiley.com/doi/full/10.3322/caac.21660https://onlinelibrary.wiley.com/doi/abs/10.3322/caac.21660https://acsjournals.onlinelibrary.wiley.com/doi/10.3322/caac.21660

Sutton, E. J., N. Onishi, D. A. Fehr, B. Z. Dashevsky, M. Sadinski, K. Pinker, D. F. Martinez, E. Brogi, L. Braunstein, P. Razavi, M. El-Tamer, V. Sacchini, J. O. Deasy, E. A. Morris and H. Veeraraghavan (2020), A machine learning model that classifies breast cancer pathologic complete response on MRI post-neoadjuvant chemotherapy, **vol. 22**, no.1, p. 57, ISSN 1465542X, doi:10.1186/s13058-020-01291-w.
https://breast-cancer-research.biomedcentral.com/articles/10.1186/s13058-020-01291-w

Thibault, G., A. Tudorica, A. Afzal, S. Y. Chui, A. Naik, M. L. Troxell, K. A. Kemmer, K. Y. Oh, N. Roy, N. Jafarian, M. L. Holtorf, W. Huang and X. Song (2017), DCE-MRI Texture Features for Early Prediction of Breast Cancer Therapy Response, **vol. 3**, no.1, pp. 23–32, ISSN 23791381, doi:10.18383/j.tom.2016.00241.

Thompson, A. M. and S. L. Moulder-Thompson (2012), Neoadjuvant treatment of breast cancer, **vol. 23**, no.SUPPL. 10, ISSN 09237534, doi:10.1093/annonc/mds324.
https://pubmed.ncbi.nlm.nih.gov/22987968/

Tian, J., D. Dong, Z. Liu and J. Wei (2021), Introduction, in *Radiomics and Its Clinical Application*, Elsevier, chapter 1, pp. 1–18, doi:10.1016/B978-0-12-818101-0.00004-5. https://linkinghub.elsevier.com/retrieve/pii/B9780128181010000045

Triwijoyo, B. K. and A. Adil (2021), Analysis of Medical Image Resizing Using Bicubic Interpolation Algorithm, **vol. 14**, no.1, p. 20, doi:10.24843/JIK.2021.V14.I01.P03.

Tudorica, A., K. Y. Oh, S. Y. Chui, N. Roy, M. L. Troxell, A. Naik, K. A. Kemmer, Y. Chen, M. L. Holtorf, A. Afzal, C. S. Springer, X. Li and W. Huang (2016), Early prediction and evaluation of breast cancer response to neoadjuvant chemotherapy using quantitative DCE-MRI, **vol. 9**, no.1, pp. 8–17, ISSN 19365233, doi:10.1016/j.tranon.2015.11.016. http://dx.doi.org/10.1016/j.tranon.2015.11.016

Van Der Waal, D., A. L. Verbeek, G. J. Den Heeten, T. M. Ripping, V. C. Tjan-Heijnen and M. J. Broeders (2015), Breast cancer diagnosis and death in the Netherlands: A changing burden, **vol. 25**, no.2, pp. 320–324, ISSN 1464360X, doi:10.1093/eurpub/cku088. https://pubmed.ncbi.nlm.nih.gov/24972595/

Van Rossum, G. and F. L. Drake (2009), *Python 3 Reference Manual*, CreateSpace, Scotts Valley, CA, ISBN 1441412697.

Waks, A. G. and E. P. Winer (2019), Breast Cancer Treatment A Review, **vol. 321**, no.3, pp. 288–300, doi:10.1001/jama.2018.19323. https://jamanetwork.com/

Wang, L., J. C. Cohen, N. Devasenapathy, B. Y. Hong, S. Kheyson, D. Lu, Y. Oparin, S. A. Kennedy, B. Romerosa, N. Arora, H. Y. Kwon, K. Jackson, M. Prasad, D. Jayasekera, A. Li, G. Guarna, S. Natalwalla, R. J. Couban, S. Reid, J. S. Khan, M. McGillion and J. W. Busse (2020), Prevalence and intensity of persistent post-surgical pain following breast cancer surgery: a systematic review and meta-analysis of observational studies, **vol. 125**, no.3, pp. 346–357, ISSN 14716771, doi:10.1016/j.bja.2020.04.088. https://doi.org/10.1016/j.bja.2020.04.088

Wasser, K., H. Sinn, C. Fink, S. Klein, H. Junkermann, H. Lüdemann, I. Zuna and S. Delorme (2003), Accuracy of tumor size measurement in breast cancer using MRI is influenced by histological regression induced by neoadjuvant chemotherapy, **vol. 13**, no.6, pp. 1213–1223, ISSN 1432-1084, doi:10.1007/S00330-002-1730-6. https://link.springer.com/article/10.1007/s00330-002-1730-6

Whiteside, T. (2008), The tumor microenvironment and its role in promoting tumor growth, **vol. 27**, no.45, p. 5904, doi:10.1038/ONC.2008.271. /pmc/articles/PMC3689267//pmc/articles/PMC3689267/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC3689267/

Woo, J., J. M. Ryu, S. M. Jung, H. J. Choi, K. Lee, J. Yu, E. Lee, S. W. Kim, J. Nam and J. Chae (2020), Breast radiologic complete response is associated with favorable survival outcomes after neoadjuvant chemotherapy in breast cancer, doi:10.1016/j.ejso.2020.08.023. https://doi.org/10.1016/j.ejso.2020.08.023

Wu, J., G. Gong, Y. Cui and R. Li (2016), Intratumor partitioning and texture analysis of dynamic contrast-enhanced (DCE)-MRI identifies relevant tumor subregions to predict pathological response of breast cancer to neoadjuvant chemotherapy, **vol. 44**, no.5, pp. 1107–1115, ISSN 10531807, doi:10.1002/jmri.25279. http://doi.wiley.com/10.1002/jmri.25279

Xiong, Q., X. Zhou, Z. Liu, C. Lei, C. Yang, M. Yang, L. Zhang, T. Zhu, X. Zhuang, C. Liang, Z. Liu, J. Tian and K. Wang (2020), Multiparametric MRI-based radiomics analysis for prediction of breast cancers insensitive to neoadjuvant chemotherapy, **vol. 22**, no.1, pp. 50–59, ISSN 16993055, doi:10.1007/s12094-019-02109-8.

https://doi.org/10.1007/s12094-019-02109-8

Yoon, H. J., Y. Kim, J. Chung and B. S. Kim (2019), Predicting neo-adjuvant chemotherapy response and progression-free survival of locally advanced breast cancer using textural features of intratumoral heterogeneity on F-18 FDG PET/CT and diffusion-weighted MR imaging, **vol. 25**, no.3, pp. 373–380, ISSN 15244741, doi:10.1111/tbj.13032.

Youden, W. J. (1950), Index for rating diagnostic tests, doi:10.1002/1097-0142(1950)3:1<32::AID-CNCR2820030106>3.0.CO;2-3.

Zhou, J., J. Lu, C. Gao, J. Zeng, C. Zhou, X. Lai, W. Cai and M. Xu (2020), Predicting the response to neoadjuvant chemotherapy for breast cancer: Wavelet transforming radiomics in MRI, **vol. 20**, no.1, ISSN 14712407, doi:10.1186/s12885-020-6523-2. https://pubmed.ncbi.nlm.nih.gov/32024483/