We are making a breast cancer deduction system that first recognizes what kind of ακτινογραφια έχει μπροστά του και έπειτα μπορεί να ανιχνεύσει αν είναι benign ή malignant.

# ~~Εισαγωγή και περιγραφή του προβλήματος~~

~~Breast cancer is the most prevalent cancer among women worldwide and remains a leading cause of cancer-related deaths. Early and accurate detection is critical for improving survival rates, as it enables timely intervention and tailored treatment strategies. Medical imaging modalities such as mammography, ultrasound, and histopathology are central to the diagnostic process, but manual interpretation can be time-consuming and prone to variability among clinicians.[[1]](#footnote-1)~~

~~Machine learning (ML), particularly deep learning (DL), has emerged as a transformative tool in breast cancer image analysis. These technologies can process large volumes of imaging data, identify subtle patterns, and assist in differentiating between benign and malignant lesions with high accuracy.~~

# ~~Περιγραφή των συνόλων δεδομένων~~

~~(αρχικά μου δόθηκε αυτό το dataset:~~ ~~<https://huggingface.co/datasets/sanaa13/breastmnist2/blob/main/README.md> αλλά δεν φόρτωνε οπότε πήρα αυτό:~~ [~~https://zenodo.org/records/10519652~~](https://zenodo.org/records/10519652)~~, που το βρήκα από την πρωτότυπη σελίδα του MNIST:~~ [~~https://medmnist.com/~~](https://medmnist.com/) ~~)~~

~~Το συγκεκριμένο σύνολο του BreastMNIST είναι ένα προεπεξεργασμένο υποσύνολο του MedMNIST+. Οι δημιουργοί του πήραν το αρχικό BreastMNIST+ dataset, το οποίο παρέχει τις εικόνες του σε ανάλυση μέχρι και 224x224 σε αντίθεση με το 28x28 του αρχικού MedMNIST[[2]](#footnote-2), του αφαίρεσαν τις επαναλαμβανόμενες εικόνες, διόρθοσαν λάθη στο annotation, cropped ώστε να αφαιρέσουν άχρηστα ή ασήμαντα όρια από τις εικόνες και ελέχθηκαν από ειδικούς radiologists για τυχόν άλλα λάθη.[[3]](#footnote-3)~~

~~Το τελικό αποτέλεσμα ήταν~~ ~~780 εικόνες οι οποίες χωρίζονται σε 3 σετ (train, validation, test) των 546 (70.00%), 78 (10.00)% και 156 (20.00%) αντιστοίχως. Μεταξύ τους ήταν labeled με σταθερό αλλά μη ισορροπημένο τρόπο, 73.8% του κάθε σετ να αντιστοιχεί σε υγιείς (normal/benign) εικόνες και το άλλο 26.92% σε εικόνες καρκινοπαθών (malignant) σε κάθε υποσύνολο. Δεν έλειπαν τιμές. Επίσης διαπιστώθηκε ένας εύκολος τρόπος για διαχωρισμό των labels δίχως την χρήση κάποιου μοντέλου, κοιτώντας απλά το mean των pixel της εκάστοτε εικόνας, το οποίο είναι σταθερά υψηλότερο για τις κανονικές εικόνες [[4]](#footnote-4).~~

~~A blue and orange rectangles

AI-generated content may be incorrect.~~

~~Τα dataset ερευνήθηκαν για corrupted εικόνες με σύντομο τρόπο εξερευνόντας εικόνες με μέγιστες και ελάχιστες τιμές φωτεινότητας στα pixel και μετέπειτα τις εικόνες με μέγιστο και ελάχιστο μέσο όρο φωτεινότητας στα pixel, ανα σετ. Οι εικόνες που προέκυψαν δεν εμφάνισαν διαφθορά οπότε τις διατήρησα. Επίσης έκανα αναζήτηση για outlier εικόνες αλλά με τον παρακάτω τρόπο αναζήτησης βρεθηκε μόνο μια εικόνα, η οποία λογικά θεωρήθηκε outlier επειδή ήταν πολύ φωτεινή. Δεδομένου πως είναι μόνο μια εικόνα που δεν παρουσιάζει φθορά και το dataset δεν είναι τόσο μεγάλο κρίθηκε πως δεν είναι αναγκαίο να αφαιρεθεί.~~

~~1. mean\_val = np.mean(image\_means)~~

~~2. std\_val = np.std(image\_means)~~

~~3. lower\_threshold = mean\_val - 3 \* std\_val~~

~~4. upper\_threshold = mean\_val + 3 \* std\_val~~

~~5.~~

Ερευνήθηκε και το ενδεχόμενο πολυωνυμικών χαρακτηρστικών αλλά απορίφθηκε πολύ εύκολα από νωρίς καθώς είχε κακά αποτελέσματα και δεν συνιθίζεται σε τασκ εικόνων ούτως ή άλλως

# Μεθοδολογία και προσέγγιση

Αρχικά, πραγματοποιήθηκε διερευνητική ανάλυση δεδομένων (EDA) για τον έλεγχο της ποιότητας, της ισορροπίας των κλάσεων και της κατανομής των χαρακτηριστικών στα σύνολα εκπαίδευσης, επικύρωσης και δοκιμής. Σχετικά με την οργάνωση των δεδομένων δημιουργήθηκαν Dataframes για τα σύνολα για την εκπαίδευση των μοντέλων αργότερα και δημιουργήθηκαν x,y υποσύνολα τα οποία αντιστοιχούν στις εικόνες και ετικέτες αντ΄σιτοιχα ώστε να είναι πιο καθολικεύσιμος ο κώδικας, παρότι η χ΄ρηση αυτής της σύμβασης έκανε τον κώδικα για την ανάλυση δεδομένων πλεοναστικό.

~~Ακολούθησε προεπεξεργασία των δεδομένων με τυποποίηση (standard scaling) και αφαίρεση χαρακτηριστικών χαμηλής διακύμανσης, ώστε να διασφαλιστεί η ομοιογένεια και η καταλληλότητα των δεδομένων για αλγορίθμους μηχανικής μάθησης. Επιπλέον,~~ ~~εφαρμόστηκαν τεχνικές εξαγωγής και μηχανικής χαρακτηριστικών (feature engineering), στατιστικά ανά εικόνα (μέση τιμή, τυπική απόκλιση, skewness, κ.λπ.), υφής (LBP) και σχήματος (π.χ. area, perimeter, compactness), για την ενίσχυση της διακριτικής ικανότητας των μοντέλων. Όλα τα παραπάνω συνοδεύονται από οπτικοποιήσεις (boxplots, pairplots, heatmaps) για καλύτερη κατανόηση της κατανομής και των συσχετίσεων των χαρακτηριστικών. Τέλος, διερευνάται ποια από τα παραπάνω βήματα και τεχνικές συμβάλλουν ουσιαστικά στη βελτίωση της ποιότητας των δεδομένων και της απόδοσης των μοντέλων, ώστε να επιλεγούν εκείνα που είναι πιο κατάλληλα για το task.~~

~~Εφαρμόστηκε PCA με 50 components και LDA στα δεδομένα για να γίνει καλύτερη εκπαίδευση μοντέλων. Ωστόσο ακόμα και έτσι, τα δεδομένα δεν ήταν διαχωρίσημα, όπως είναι εμφανές από το tsne plot παρακάτω.~~

~~A graph with blue and orange dots

AI-generated content may be incorrect.~~

~~Δοκιμάστηκαν διάφορες τιμές και δεν μπορούσε να βρεθεί αποτελεσματικός τρόπος να εμφανιστούν τα δεδομένα, εμφανιζόντοσυαν όλα σε μορφή συγκεχημένου νέφους.~~

classification

Στην συνέχεια έγινε μείωση διαστάσεων των δεδομένων με PCA και LDA, με τα δεδομένα που προέκυψαν να χρησιμοποιούνται ως είσοδος σε διάφορους αλγορίθμους μηχανικής μάθησης, όπως Logistic Regression, Support Vector Machines (SVM), k-Κοντινότεροι Γείτονες (k-NN) και Naive Bayes. Για κάθε αλγόριθμο πραγματοποιήθηκε αναζήτηση υπερπαραμέτρων (Grid Search) και αξιολόγηση της απόδοσης με μετρικές όπως η ακρίβεια, η precision, η recall, το F1-score και η καμπύλη ROC, αλλά δόθηκε παραπάνω σημασία στο recall των τιμών 0 καθώς είναι ιατρικό task. Η συστηματική αυτή διαδικασία επέτρεψε τη σύγκριση των μοντέλων και την επιλογή της βέλτιστης προσέγγισης για το συγκεκριμένο πρόβλημα.

Κατα την εξερεύνση των datasets που προέκυψαν από την προεπεξεργασία η επίλογη κατάλληλου συνόλου αποκαλύφθηκε να ειναι περίπλοκη. Πραγματοποιήθηκε εκτενής πειραματική αξιολόγηση διαφορετικών συνόλων χαρακτηριστικών, όπως στατιστικά ανά εικόνα, υφής (LBP), σχήματος και οι συνδυασμοί τους, με στόχο τη βελτιστοποίηση της απόδοσης των μοντέλων μηχανικής μάθησης. Για κάθε σύνολο χαρακτηριστικών εκπαιδεύτηκαν και αξιολογήθηκαν μοντέλα Logistic Regression τόσο στο validation όσο και στο test set, με και χωρίς τη χρήση PCA, ο οποίο χρησιμοποιήθηκε για μείωση της διαστασιμότητας. Η χρήση Logistic Regression μοντέλων έγινε για την ταχύτητα τους και ο έλεγχος γινόταν και στο test και στο validation set για τυχόν διαφοροποιήσεις που μπορεί να προκύψουν. Η υπόθεση ήταν πως τα μοντέλα με δεδομένα που είναι με περισς΄τοερες πληροφορίες μπορει να ευδοκιμούν καλύτερα αποτελέσματα, ωστόσο δεν ήταν προφανές στην αρχή ποιά features και ποιοί συνδιασμοί θα ήταν αυτοί με τα βέλτιστα αποτελέσματα.

Δυστυχως, κατά τη διάρκεια των δοκιμών παρατηρήθηκαν σημαντικές διακυμάνσεις στην απόδοση των μοντέλων ανάλογα με το σύνολο χαρακτηριστικών και το split των δεδομένων, γεγονός που δυσκόλεψε την επιλογή των «καλύτερων» χαρακτηριστικών. Αφενώς, οι προβλέψεις στα test και validation sets παρουσίαζαν σοβαρές διακυμάνσεις και μεταξύ τους και αφετέρου, χαρακτηριστικά που εμφάνιζαν υψηλή ακρίβεια σε ένα split, παρουσίαζαν χαμηλότερη απόδοση σε άλλο, υποδεικνύοντας ότι η σταθερότητα και η γενίκευση των αποτελεσμάτων δεν ήταν δεδομένες. Έτσι, η τελική επιλογή των χαρακτηριστικών θα χρειαζόταν να γίνει με βάση τη συνολική συμπεριφορά τους και όχι μόνο μεμονωμένες επιδόσεις, λαμβάνοντας υπόψη τη συνέπεια και τη γενικευσιμότητα των αποτελεσμάτων. Στην τελική επέλεξα να χρησιμοποιήσω το dataset συνδίαζε variance set και είχε επίσης τις στατιστικές από το mean και std, είχε τ ακαλύτερα αποτελέσματα και βρίσκει μια καλή ισορροπία μεταξύ λίγης και πολύς προεπεξεργασίας.

Επίσης άλλες δοκιμές που έκανα ήταν μεταξύ των σετ έτσι όπως τα είχα από το npz αρχείο και με έναν νέο διαχωρισμό των δεδομένω. Πρόσεξα να γίνει ισορροπημένα ώστε να υπάρχουν αντίστοιχα ποσοστά μεταξύ των υποσυνόλων και να ανακατευθούν ώστε να εγγυηθεί διαφοροποίηση από τον αρχικό χωρισμό. Το αποτέλεσμα αυτού ήταν τραγικά χειρότερα αποτελέσματα από ότι πριν. Ϋστερα από διερεύνση των δεδομεών δεν μπροούσα να καταλήξω σε κάποιο συμπέρασμα για το γιατί συνέβαινε αυτό. Ενδεχομένως να ε΄χιε χωριστεί με ένα συγκεκριμένο σκεπτικό με αυτόν τον τρόπο τον οποίο δεν διευκρίνισαν πουθενά ή να ήταν εντελώς τυχαίο, σε κάθε περίπτωση έκρινα πως θα είναι καλύτερο να διατηρήσω τα δεδομένα χωρισμένα όπως πρότειναν. Επίσης επειδή θα εφαρμόσουμε k-folding στους αλγόριθμους που ακολουθούν επέλεξα να συνδιάσω το validation set (10% του συνόλου) στο training set (70% του συνόλου) ώστε να έχω το Test set (20% του συνόλου) για δοκιμές που είναι μεγαλύτερο και άρα θα μειώνει το variance, δίνοντας έτσι πιο έγκυρες προβλέψεις.

Θέλω να **ελαχιστοποιήσω τις περιπτώσεις ψευδώς αρνητικών**, δηλαδή τις περιπτώσεις όπου υπάρχει κακοήθης όγκος (κλάση 0) αλλά το μοντέλο προβλέπει "φυσιολογικό" (κλάση 1). Συνεπώς, πρέπει να **μεγιστοποιήσω την ανάκληση (recall) για την τιμή 0**. Το βασικό μοντέλο είχε πολύ χαμηλή ανάκληση για την κλάση 0 (περίπου 34%), πιθανότατα επειδή το σύνολο δεδομένων δεν είναι ισορροπημένο.

Για να το αντιμετωπίσω, **ρύθμισα την παράμετρο class\_weight του logistic regression σε 'balanced'** και **αφαίρεσα την επιλογή multi\_class='multinomial'**, καθώς το πρόβλημά μου αφορά μόνο δύο κατηγορίες, σε αντίθεση με των υπολοίπων στην ομάδα μου. Εστίασα την GridSearchCV στη χρήση του **αλγορίθμου lbfgs**, καθώς το σύνολο δεδομένων μου δεν είναι αρκετά μεγάλο ώστε να αξιοποιήσω τον saga, και έψαξα για υπερπαραμέτρους που θα **μεγιστοποιούσαν το συνολικό recall**.

Ωστόσο, αυτό οδήγησε σε recall 0.96 για την κλάση 1 αλλά μόλις **0.40 για την κλάση 0**. Χρησιμοποιώντας έναν **προσαρμοσμένο μετρητή (custom scorer)** που επικεντρώνεται στην ανάκληση της κλάσης 0, το μοντέλο έφτασε σε **ανάκληση 0.67 για την κλάση 0**, που παρότι αποτελεί βελτίωση, **παραμένει μη ικανοποιητική**. Τουλάχιστον, η συνολική ακρίβεια στο test set **δεν μειώθηκε σημαντικά**, πηγαίνοντας από 0.81 σε 0.79.

Τέλος, εξερεύνησα διαφορετικές επιλογές για **PCA components**, αλλά **χωρίς επιτυχία**.

**Με παρόμοιο τρόπο όπως στο Logistic Regression**, εφάρμοσα grid search για την εύρεση των βέλτιστων υπερπαραμέτρων και στους υπόλοιπους αλγορίθμους. Ακολουθούν τα αποτελέσματα:

|  |  |  |  |
| --- | --- | --- | --- |
|  | PCA Components | Hyperparameters | Results |
| Logistic Regression | 50 | 'C': np.float64(0.0001),  'class\_weight': 'balanced',  'max\_iter': 100,  'penalty': 'l2',  'solver': 'lbfgs' | Recall\_0: 0.67  Test Accuracy: 0.7948717948717948 |
| SVM | 50 | 'C': 1,  'class\_weight': 'balanced',  'gamma': 'scale',  'kernel': 'rbf' | Recall\_0: 0.67  Test Accuracy: 0.7948717948717948 |
| K-NN | 50 | 'metric': 'manhattan', 'n\_neighbors': np.int64(2), 'weights': 'uniform' | Recall\_0: 0.76  Test Accuracy: 0.7051282051282052 |
| Naive Bayes | 50 | ‘var\_smoothing’: np.float64(2.782559402207126e-06 | Recall\_0: 0.76  Test Accuracy: 0.6602564102564102 |

Η ανάκληση των KNN και Naïve Bayes έφτασε στο υψηλότερο επίπεδο των 0.76, αλλά η ακρίβεια στο test set ήταν 0.705, αρκετά χαμηλότερη από την καλύτερη δυνατή τιμή που επιτεύχθηκε με το Logistic Regression και το SVM, 0.794.

LDA

|  |  |  |  |
| --- | --- | --- | --- |
|  | PCA Components | Hyperparameters | Results |
| Logistic Regression | 50 | 'C': np.float64(0.0001), 'class\_weight': 'balanced', 'max\_iter': 100,  'penalty': None,  'solver': 'lbfgs' | Recall\_0: 0.67  Test Accuracy: 0.8141025641025641 |
| SVM | 50 | 'C': 1,  'class\_weight': 'balanced',  'gamma': 0.0001,  'kernel': 'rbf' | Recall\_0: 0.93  Test Accuracy: 0.4807692307692308 |
| K-NN | 50 | 'metric': 'euclidean', 'n\_neighbors': np.int64(2), 'weights': 'uniform' | Recall\_0: 0.76  Test Accuracy: 0.7435897435897436 |
| Naive Bayes | 50 | 'var\_smoothing': np.float64(2.782559402207126e-06) | Recall\_0: 0.60  Test Accuracy: 0.8076923076923077 |

ΝΕΥΡΩΝΙΚΑ

Τέλος, η προσέγγιση επικεντρώθηκε στην αξιοποίηση βαθιών νευρωνικών δικτύων για την ταξινόμηση των ιατρικών εικόνων. Αρχικά, υλοποιήθηκε ένα Πολυεπίπεδο Αντιληπτικό Δίκτυο (MLP), όπου πραγματοποιήθηκε πειραματισμός με διαφορετικές αρχιτεκτονικές, συναρτήσεις ενεργοποίησης, ρυθμούς μάθησης και τεχνικές regularization (dropout και L2). Στη συνέχεια, σχεδιάστηκε και εκπαιδεύτηκε ένα Συνελικτικό Νευρωνικό Δίκτυο (CNN), με στόχο την αυτόματη εξαγωγή χωρικών χαρακτηριστικών από τις εικόνες. Για το CNN δοκιμάστηκαν διάφορες παραλλαγές ως προς τη δομή, τις συναρτήσεις ενεργοποίησης και τις υπερπαραμέτρους, ενώ εφαρμόστηκε regularization (dropout, weight decay) για την αποφυγή υπερπροσαρμογής. Η αξιολόγηση των μοντέλων έγινε με χρήση μετρικών όπως η ακρίβεια, το classification report, οι καμπύλες ROC και οι confusion matrices σε train, validation και test set. Μέσω αυτής της διαδικασίας, συγκρίθηκαν οι επιδόσεις των δύο προσεγγίσεων και αναδείχθηκε η αποτελεσματικότητα των βαθιών νευρωνικών δικτύων στην ανάλυση ιατρικών εικόνων

Για τα νευρωνικά δίκτυα αξιοποίησα τις εικόνες δίχως επεξεργασία, ωστόσο άλλαξα την αναλογία τους σε 70/20/10 για train, validation και test σετ αντίστοιχα ώστε να έχουν περισσότερα δεδομένα για να γίνεται η επικύρωση.

~~Στο mlp του είπα να κάνει 70 % recal και 30 acuracy επειδή ήδη έφτιαξα εύκολα μοντέλο που βελτιστοποιεί recall 0. Δοκιμάστηκαν διάφορες μέθοδοι για να βρω ισορρπία μεταξύ υψηλού recall και ccuracy, ωσόσο τα αποτελέσματα ήταν πολύ κακά ή μεγιστοποιούσε το recall του 1 για ανεξήγητους λόγους, οπότε εκεί έκανα συμβιβασμό και το άφησα με το 70/30 recall/ accuracy split.~~

Στο cnn το μοντέλο έβγαζε σχεδόν 0 στον recall 0, το οποίο είναι το αντίθετο από αυτό που θέλουμε.

…

Πινακάκι αποτελεσμάτων, σε vs code

logreg\_params = dict(max\_iter=10000, random\_state=56, solver='saga', n\_jobs=-1)

Feature Set Val Acc Test Acc

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

Variance-thresholded 0.8077 0.7500

Variance + mean/std 0.8590 0.8269

Stats + LBP 0.7436 0.7308

Stats + LBP + Shape 0.7308 0.7244

All Shape Features 0.7308 0.7308

PCA 50

logreg\_params = dict(max\_iter=10000, random\_state=56, solver='saga', n\_jobs=2)

Feature Set with PCA Val Acc Test Acc

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

Standardized only (PCA) 0.8462 0.7885

Variance-thresholded (PCA) 0.8462 0.7885

Variance + mean/std (PCA) 0.8462 0.7885

Stats + LBP (PCA) 0.7436 0.7244

Stats + LBP + Shape (PCA) 0.5513 0.6090

All Shape Features (PCA) 0.5513 0.5577

PCA 95% threshhold

logreg\_params = dict(max\_iter=6000, random\_state=56, solver='liblinear')

Feature Set for 95% variance for high-dimensional features Val Acc Test Acc

Standardized only (PCA 50) 0.8333 0.7885

Variance-thresholded (PCA 250) 0.8077 0.7115

Variance+mean/std (PCA 249) 0.7821 0.7244

Stats + LBP (no PCA) 0.7436 0.7244

Stats + LBP + Shape (no PCA) 0.7308 0.7308

All Shape Features (no PCA) 0.7308 0.7244

max\_iter=15000, random\_state=56, solver='saga', n\_jobs=-1

Feature Set (New Split) Val Acc Test Acc

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

Standardized only (new) 0.7436 0.6859

Variance-thresholded (new) 0.7436 0.6859

Variance + mean/std (new) 0.8077 0.7821

Stats + LBP (new) 0.7308 0.7372

Stats + LBP + Shape (new) 0.7308 0.7308

Pca =50

logreg\_params = dict(max\_iter=10000, random\_state=56, solver='saga', n\_jobs=-1)

Feature Set (New Split, PCA) Val Acc Test Acc

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

Standardized only (PCA) 0.7308 0.7949

Variance-thresholded (PCA) 0.7308 0.7949

Variance + mean/std (PCA) 0.7308 0.7949

Stats + LBP (PCA) 0.7308 0.7500

Stats + LBP + Shape (PCA) 0.5897 0.5641

logreg\_params = dict(max\_iter=6000, random\_state=56, solver='liblinear')

n\_components = 50

Feature Set for 95% variance for high-dimensional features (New Split) Val Acc Test Acc

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

Standardized only (PCA 50) 0.7308 0.8013

Variance-thresholded (PCA 245) 0.7179 0.6923

Variance+mean/std (PCA 243) 0.6923 0.6987

Stats + LBP (no PCA) 0.7308 0.7308

Stats + LBP + Shape (no PCA) 0.7179 0.7372

|  |  |
| --- | --- |
|  | CNN |
|  |  |
| Best Recall 0 | Test Loss: 0.2699,  Test Accuracy: 92.31%  Recall 0: 0.83 |
| Best custom metric | Test Loss: 0.3670,  Accuracy: 91.03%,  Recall 0: 0.7619 |

|  |  |
| --- | --- |
|  | LDA |
| Logistic Regression | Recall\_0: 0.67  Test Accuracy: 0.8141025641025641 |
| SVM | Recall\_0: 0.93  Test Accuracy: 0.4807692307692308 |
| K-NN | Recall\_0: 0.76  Test Accuracy: 0.7435897435897436 |
| Naive Bayes | Recall\_0: 0.60  Test Accuracy: 0.8076923076923077 |

|  |  |
| --- | --- |
|  | PCA |
| Logistic Regression | Recall\_0: 0.67  Test Accuracy: 0.7948717948717948 |
| SVM | Recall\_0: 0.67  Test Accuracy: 0.7948717948717948 |
| K-NN | Recall\_0: 0.76  Test Accuracy: 0.7051282051282052 |
| Naive Bayes | Recall\_0: 0.76  Test Accuracy: 0.6602564102564102 |

# Συμπεράσματα και προτάσεις για μελλοντική έρευνα

Θα μπορούσε να έχιε ετικέτα για benign ώστε να μπορεί να κατηγοριοποιήσει πιο εύστοχα γιατί τώρα μπορεί να τα μπερδεύει

Περαιτέρω εξερεύνση των autoencoder

More techniques for imbalanced data

Clean up the code more

Could make a tsne plot that made sense

More time with the neural networks

* [ ] Precision-Recall Curve + F1 Score for imbalanced sets (from log reg)
* [ ] Loss / accuracy plot (όπως στην τν2)

# Βιβλιογραφικές αναφορές

Jalloul, R., Chethan, H. K., & Alkhatib, R. (2023). A Review of Machine Learning Techniques for the Classification and Detection of Breast Cancer from Medical Images. *Diagnostics*, *13*(14), 2460. <https://doi.org/10.3390/diagnostics13142460>

Walid Al-Dhabyani, Mohammed Gomaa, Hussien Khaled, Aly Fahmy,Dataset of breast ultrasound images,Data in Brief, Volume 28, 2020, 104863, ISSN 2352-3409, <https://doi.org/10.1016/j.dib.2019.104863>.

1. ~~Jalloul, R., Chethan, H. K., & Alkhatib, R. (2023). A Review of Machine Learning Techniques for the Classification and Detection of Breast Cancer from Medical Images.~~*~~Diagnostics~~*~~,~~*~~13~~*~~(14), 2460.~~ [~~https://doi.org/10.3390/diagnostics13142460~~](https://doi.org/10.3390/diagnostics13142460) [↑](#footnote-ref-1)
2. [~~https://bionichaos.com/BreastMNIST/~~](https://bionichaos.com/BreastMNIST/) [↑](#footnote-ref-2)
3. [~~https://www.sciencedirect.com/science/article/pii/S2352340919312181?via%3Dihub~~](https://www.sciencedirect.com/science/article/pii/S2352340919312181?via%3Dihub) [↑](#footnote-ref-3)
4. <https://www.youtube.com/watch?v=imdS1LIlISY> [↑](#footnote-ref-4)