# AN EXPLORATION OF ML METHODS FOR BREAST CANCER CLASSIFICATION

ALESSANDRO EMMANUEL PECORA

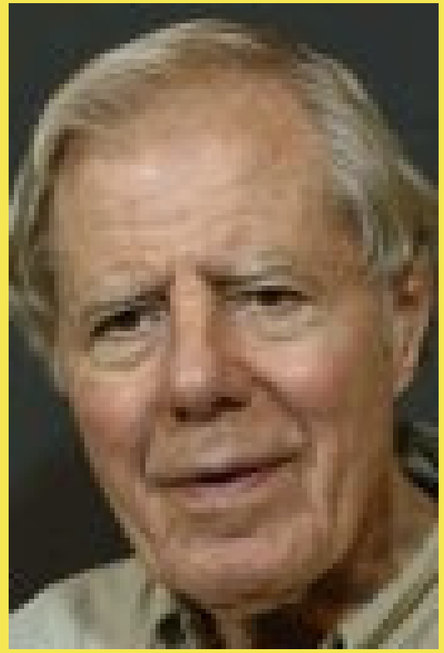
POLYTECHNIC UNIVERSITY OF TURIN - MATHEMATICS IN MACHINE LEARNING- 2021/2022

# INTRODUCTION

BREAST CANCER (BC) IS ONE OF THE MOST COMMON CANCERS AMONG WOMEN WORLDWIDE.

MACHINE LEARNING (ML) IS WIDELY RECOGNIZED AS THE METHODOLOGY OF CHOICE IN BC PATTERN CLASSIFICATION AND FORECAST MODELLING.

IN THIS RESEARCH, THE ANALYSIS IS BASED ON THE REAL DATASET ”BREAST CANCER WISCONSIN (ORIGINAL) DATA SET” [WOLBERG, 1992].



Dr. William H. Wolberg

# DATASET OVERVIEW

- The dataset containts **699** instances,.
  - Those arrive periodicall y as Dr. Wolberg reports his clinical cases.
  - The database therefore reflects this chronological grouping of the data.

- For each sample, we have:
  - **9** cytological characteristics of fine needle aspirations of the breast, classified from 1 to 10.
  - The sample id number, and the class of bresast, (benign or malign).

---

### MITOSES
The process in cell division.

### CLUMP THICKNESS
Measures the aggregations of epithelial cells.

### UNIFORMITY OF CELL SIZE
Indicating metastasis to lymph nodes.

### UNIFORMITY OF CELL SHAPE
Identifyng cancerous cell of varying size.

### MARGINAL ADHESION
Cohesion of the peripheral cells of the epithelial cell aggregates.

### SINGLE EPITHELIAL CELL SIZE
Large value are often related with malignant cells.

### BLAND CHROMATIN
Describes a uniform texture of the nucleus.

### BARE NUCLEI
Proportion of single epithelial nuclei devoid of surrounding cytoplasm.

### NORMAL NUCLEOLI
Usually very small in benign cells.

# DATASET EXPLORATION

## BALANCING

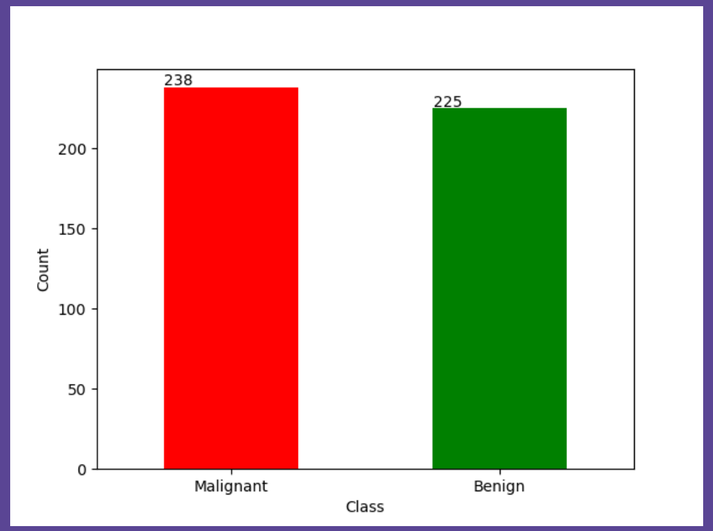
- To avoid biased classification:
  - Check the distributions of sampling over the labels.


- Dataset is unbalanced:
  - We have almost double benign instances.
  - Check if removing duplicate can balance the dataset.

# DATASET EXPLORATION

## DUPLICATES

- Checking for duplicates reveals two kind of duplicates:
  - Samples of same patient with same results: **54**
  - Samples of different patients with same results: **182**




- Dropping all 236 duplicates lead to a balanced **dataset with 463 samples.**

# DATASET EXPLORATION

## NULL VALUE HANDLING

- Present only for **bare nuclei** attribute:
  - 14 null instances founded.
  - All attributes are integer, replace null instances with median strategy.
  - Other possibility, knowing the means of bare nuclei, is to replace null values with 0.

## STANDARDIZATION

- Remind:
  - Some technique used require normalized data.
  - When a normalized dataset is needed we use standardization technique.
- Standardization consist in rescale the values to achieve 0 as mean and 1 as standard deviation.

# DATA ANALYSIS & FEATURES EXTRACTION

## CENTROIDS DISTANCES

- Inspired from "Supervised compact hypersphere" [Tingting Mu, 2008].
- Simple but efficient technique of features extraction:
  - At training time compute the centroids for both, malignant and benign cases.
  - Centroids are the two mean of the respectively benign and malign samples.
  - Centroids are points in a 9-dimensional space.

$$\mathbf{Centroid('m')} = \frac{1}{|m\_samples|} \sum_{x \in m\_sample} \mathbf{x}$$

$$\mathbf{Centroid('b')} = \frac{1}{|b\_samples|} \sum_{x \in b\_sample} \mathbf{x}$$

# DATA ANALYSIS & FEATURES EXTRACTION

## CENTROIDS DISTANCES

- When new sample arrives:
  - Compute its distances from malignant and benign centroid.
  - Use these distances as new features of the sample.

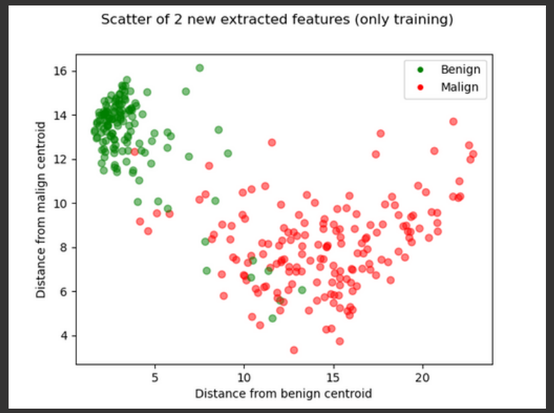
$$CD('b', \mathbf{x}) = \sqrt{\sum_{i}^{n\_features} (Centroid('b')_i - x_i)}$$

$$CD('m', \mathbf{x}) = \sqrt{\sum_{i}^{n\_features} (Centroid('m')_i - x_i)}$$

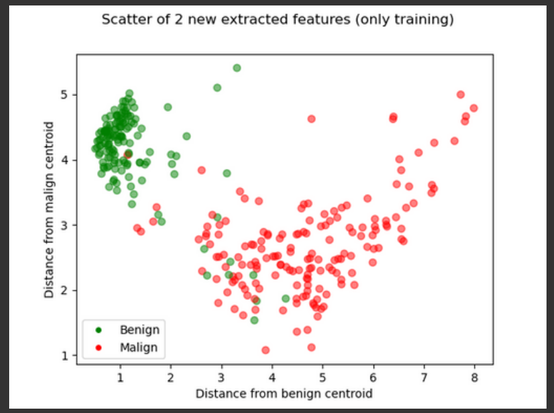
# DATA ANALYSIS & FEATURES EXTRACTION

## CENTROIDS DISTANCES

- Centroid distances are affected by non-normalized data:
  - Standardize each feature of the samples before extraction.




Extraction of *CD* features from **non standardized** dataset



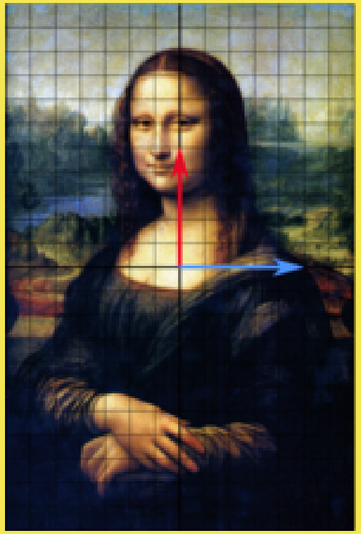

Extraction of *CD* features from **standardized** dataset

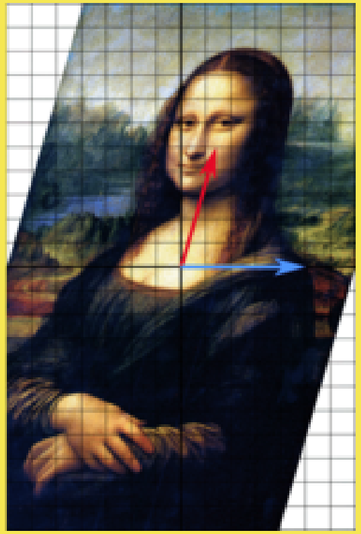
- In the right scatterplot, the features, better reflects the non lineariity of the extraction function.

# DATA ANALYSIS & FEATURES EXTRACTION

## PRINCIPAL COMPONENT ANALYSIS (PCA)

- PCA consists in finding the eigenvectors of the covariance matrix of the dataset.
- Eigenvectors are linear transformations such that:
  - No changes occurs in direction.
  - At most the scale is changed based on their associated eigenvalue.



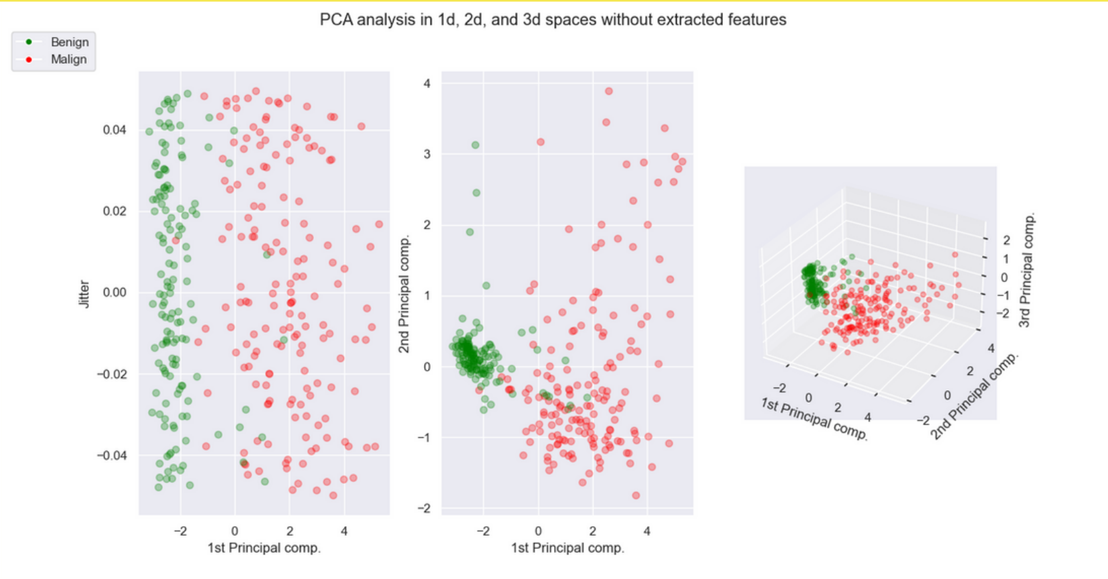


- The blue vector represents an eigenvector.
- The eigenvalue associated is 1.

# DATA ANALYSIS & FEATURES EXTRACTION

## PRINCIPAL COMPONENT ANALYSIS (PCA)

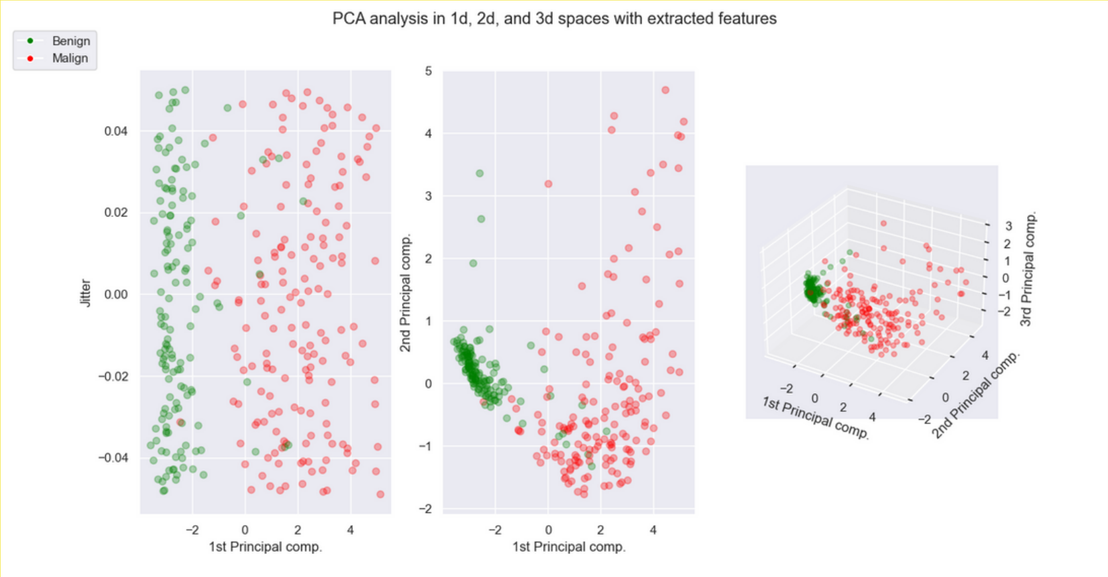
- PCA sorts the eigenvectors based on their eigenvalues.
- Then extracts the first **K** eigenvectors:
  - Principal components.
- Reducing is applyed projecting the data on the principal components.
  - In this way we do a projection that maximize the variance.
- We reduce the dimensionality losing the minimum amount of information.



# DATA ANALYSIS & FEATURES EXTRACTION

## PRINCIPAL COMPONENT ANALYSIS (PCA)

- Try PCA by reducing orginal features and the extracted ones:

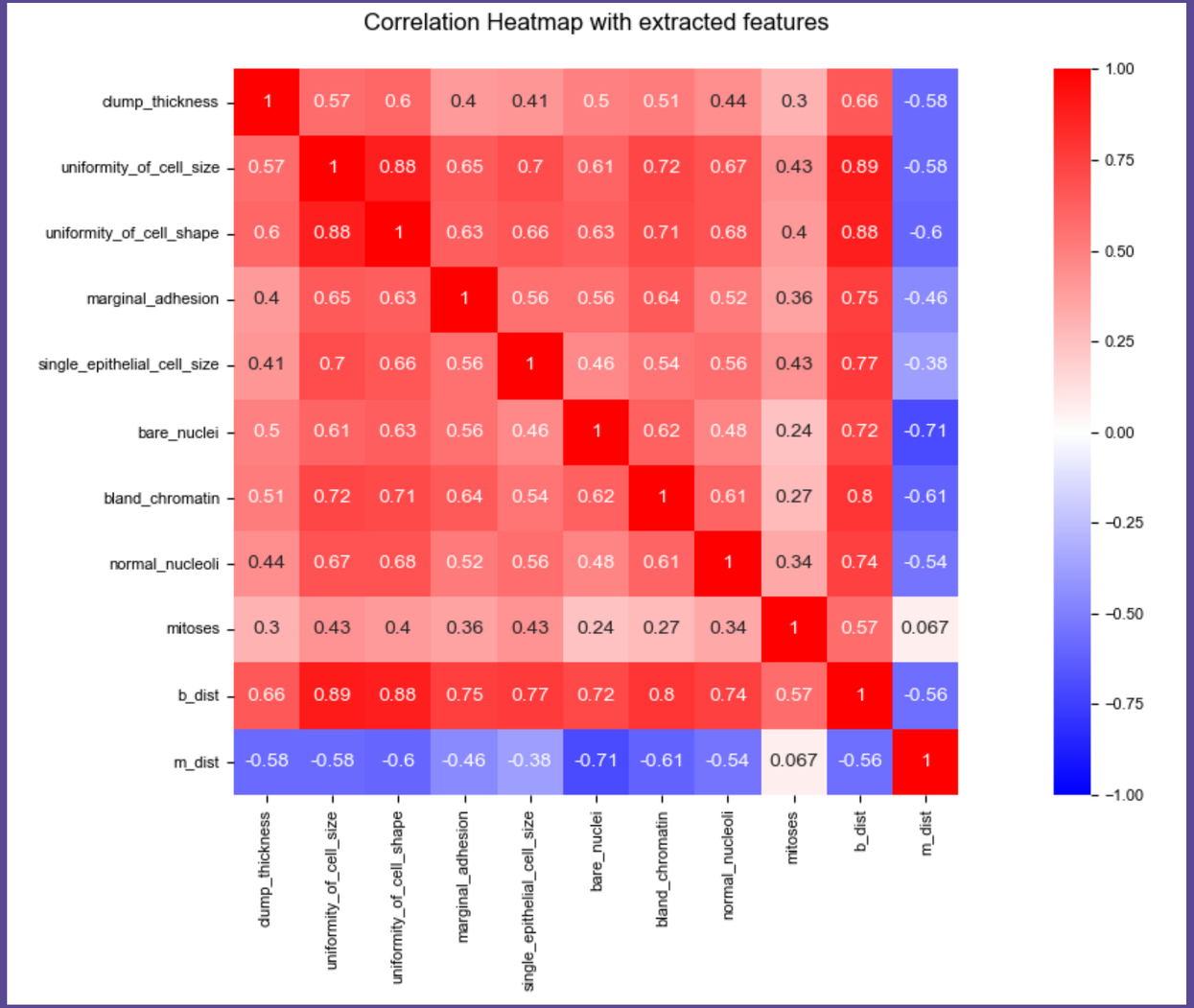



- We can compute the total **Explained Variance,** i.e. the amount of information that we don't lose by reduction:
  - K=3, reduction from only **original features: EV=76.68%**
  - K=3, reduction from **original and extracted features: EV=78.98%**

# DATA ANALYSIS & FEATURES EXTRACTION

## CORRELATION ANALYSIS

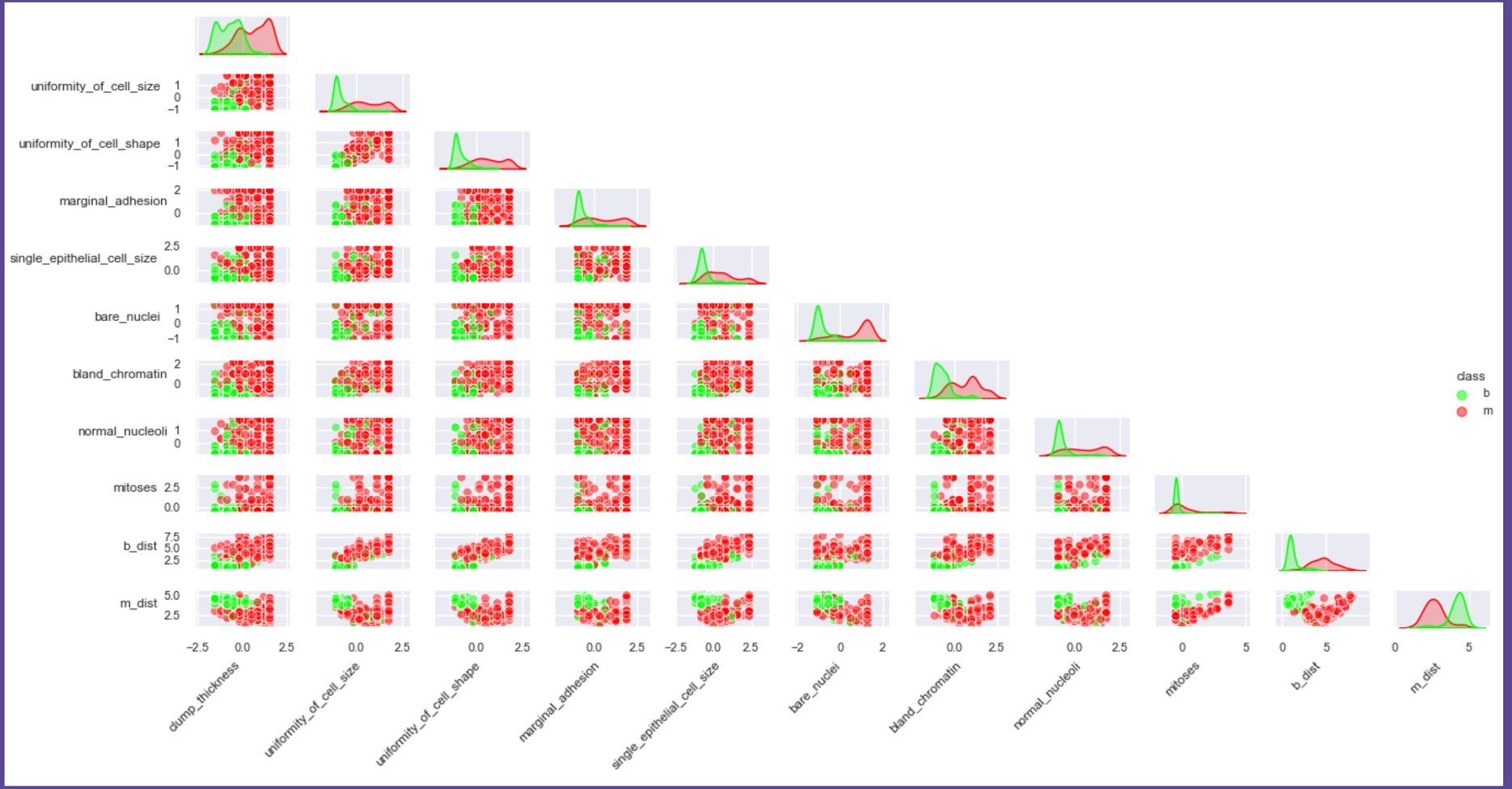
- Statistical method:
  - Measures the ”strength” of the linear relationship between two variables.
  - We use Pearson correlation coefficient:
    - 0 means that a linear correlation not exists.



# DATA ANALYSIS & FEATURES EXTRACTION

## PAIRS SCATTERPLOT

- Another way to visualize the correlation.
- Allow also to see (in the diagonal) the distribution of single variables across the labels.



# EXPERIMENT SETTINGS

## GRID SEARCH

- Used to find the optimal hyperparameters of a model.
- Builds a model for every combination of hyperparameters.
- Chooses the best model.

## CROSS VALIDATION

- Used to evaluate and compare learning algorithms.
- Divide the dataset in k fold:
  - Train on k - 1 folds.
  - Test on the last one.
  - Reiterate for all folds.
- Takes the average as final score.

## TOOLS & SETTINGS

- Experiments are made with a test set consisting in 33% of the dataset (k=5 for tuning on train-val).
- The language used is python, with scikit-learn library.

# MODELS & RESULTS

## INTRO

- Four models are tested:
  - K-nearest neighbors
  - Random forest
  - Logistic regression
  - Support vector machine

- For each model class, 3 models with:
  - Only original features.
  - Original and the extracted ones.
  - Only the extracted.

- We are in medical field:
  - Retrieve probability of the prediction as index of certainty.
  - Use single label scores:
    - False benign are more critical than false malign.

# MODELS & RESULTS

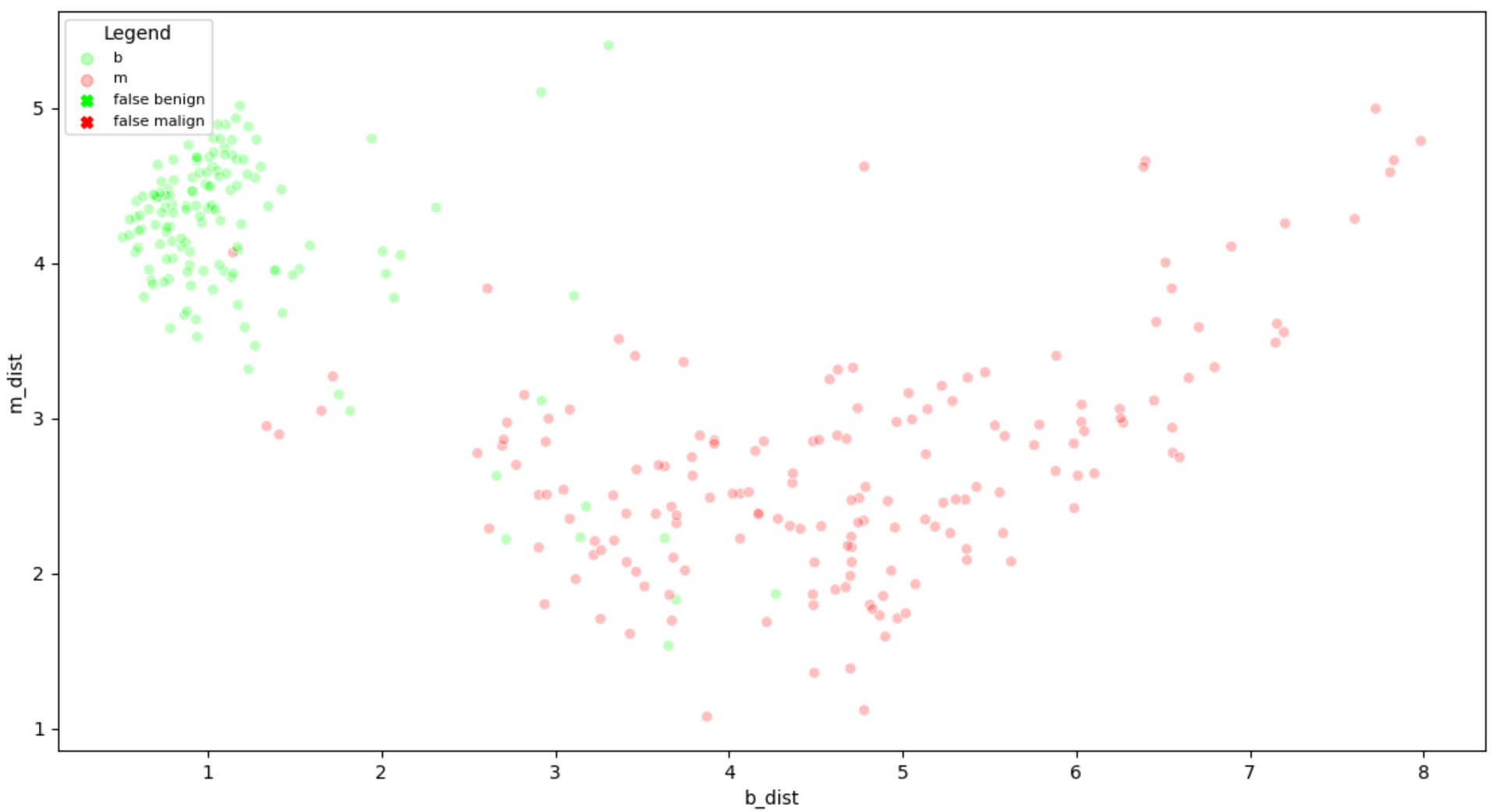
## K-NEAREST NEIGHBORS

- Assumes that similar things exist in close proximity:
  - Similar things are near to each other.

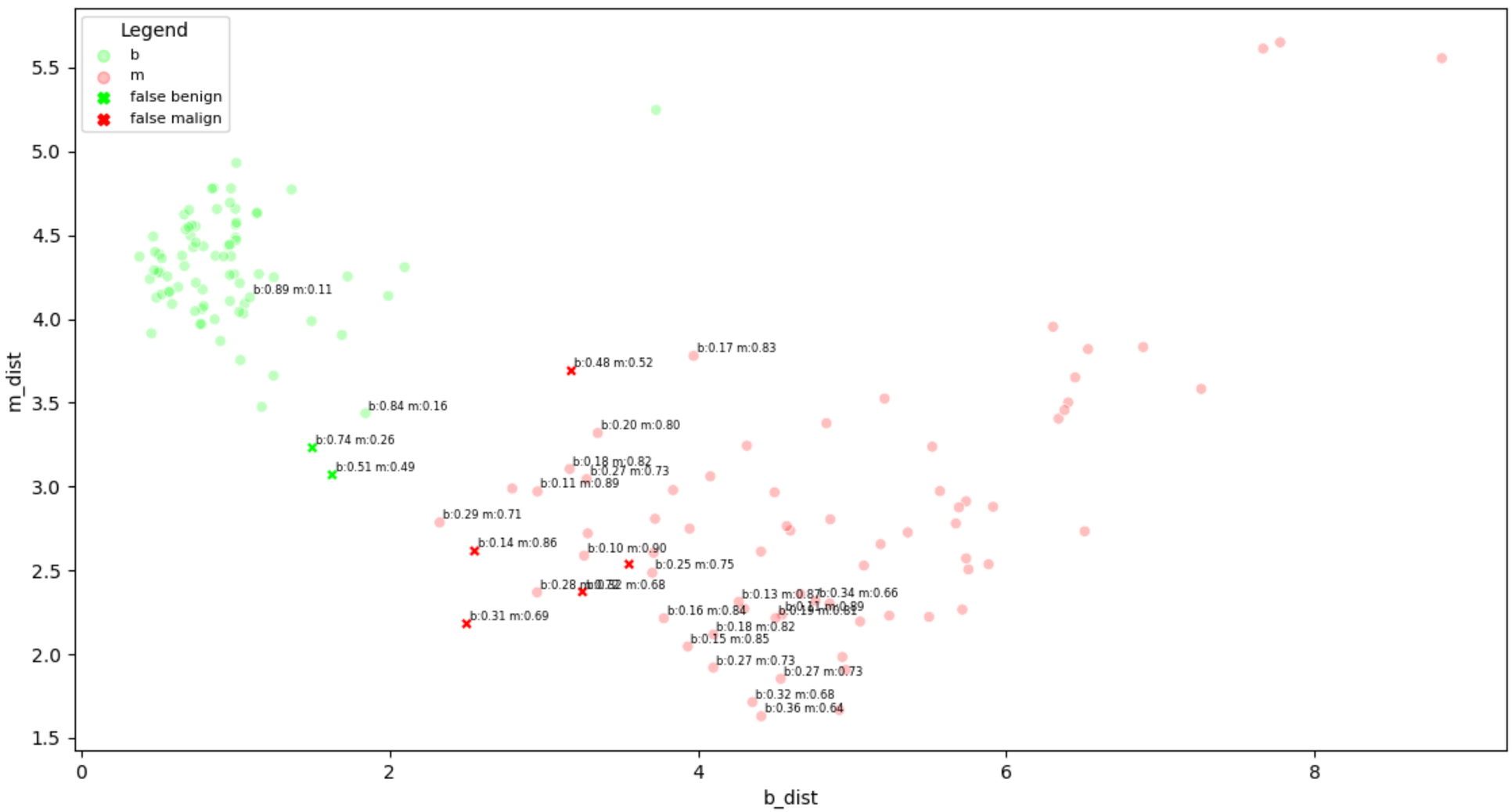
- Compute the distances of a new sample from all already classified samples:
  - Look at the K-nearest neighbors.
  - The prediction is the class of the mayority of the k-nearest neighbors.

| | Best params | Accuracy | Precision | | Recall | | F1 | | Support | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | Benign | Malign | Benign | Malign | Benign | Malign | Benign | Malign |
| KNN (original feats.) | K:3; weights: uniform | 0,92 | 0,91 | 0,94 | 0,95 | 0,89 | 0,93 | 0,91 | 82 | 71 |
| KNN (CD+Original feats.) | K:3; weights: distance | 0,93 | 0,92 | 0,94 | 0,95 | 0,90 | 0,93 | 0,92 | 82 | 71 |
| KNN (CD feats.) | K:10; weights: distance | 0,95 | 0,97 | 0,93 | 0,94 | 0,97 | 0,96 | 0,95 | 82 | 71 |

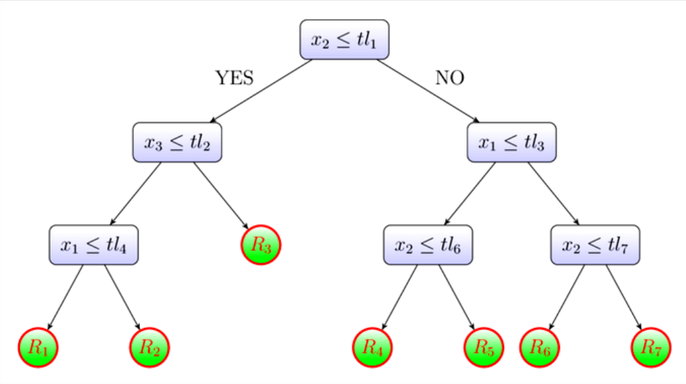
# K-NEAREST NEIGHBORS (CD FEATURES ONLY) TRAIN



# K-NEAREST NEIGHBORS (CD FEATURES ONLY) TEST



# MODELS & RESULTS

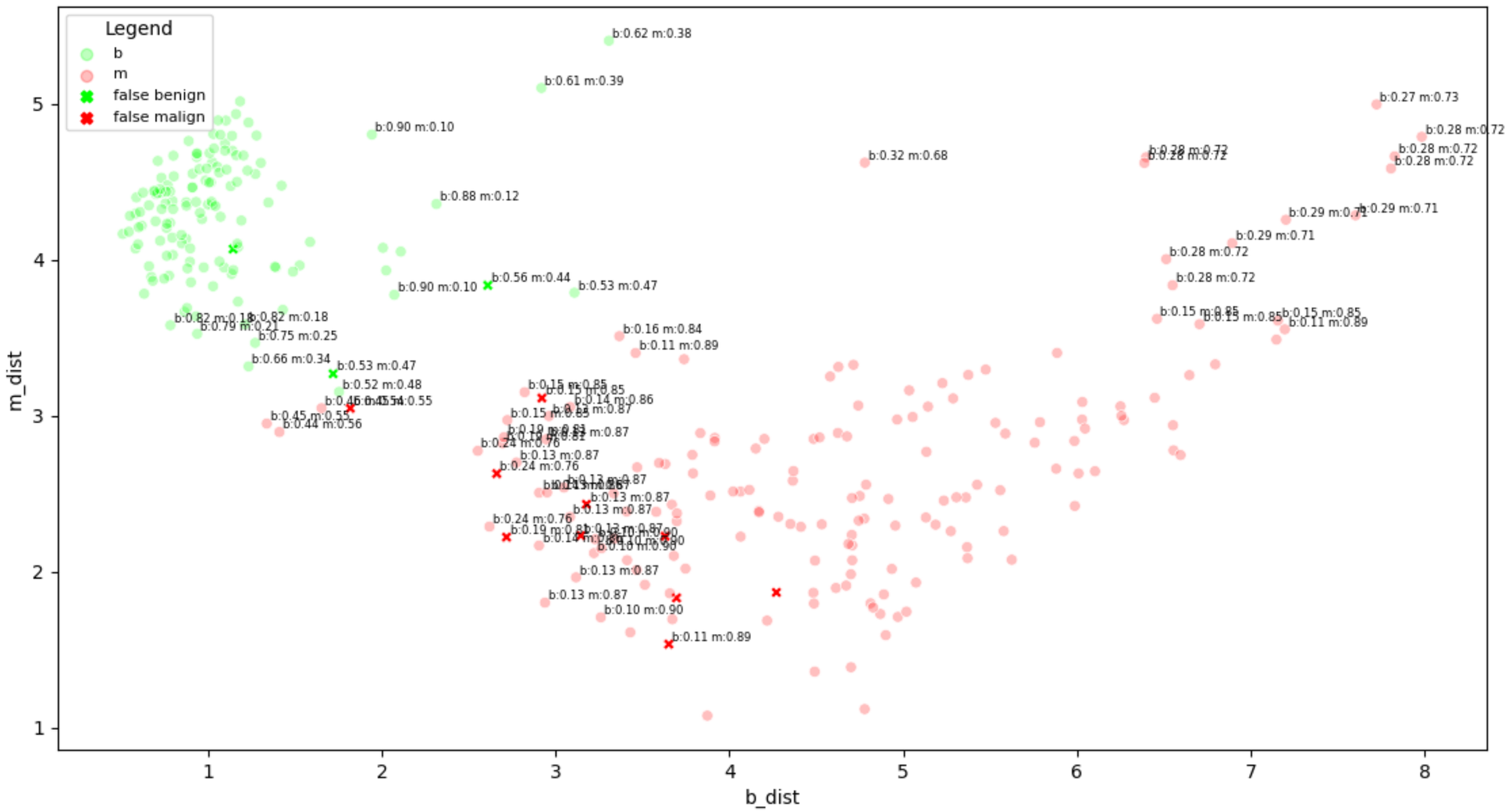



## RANDOM FOREST

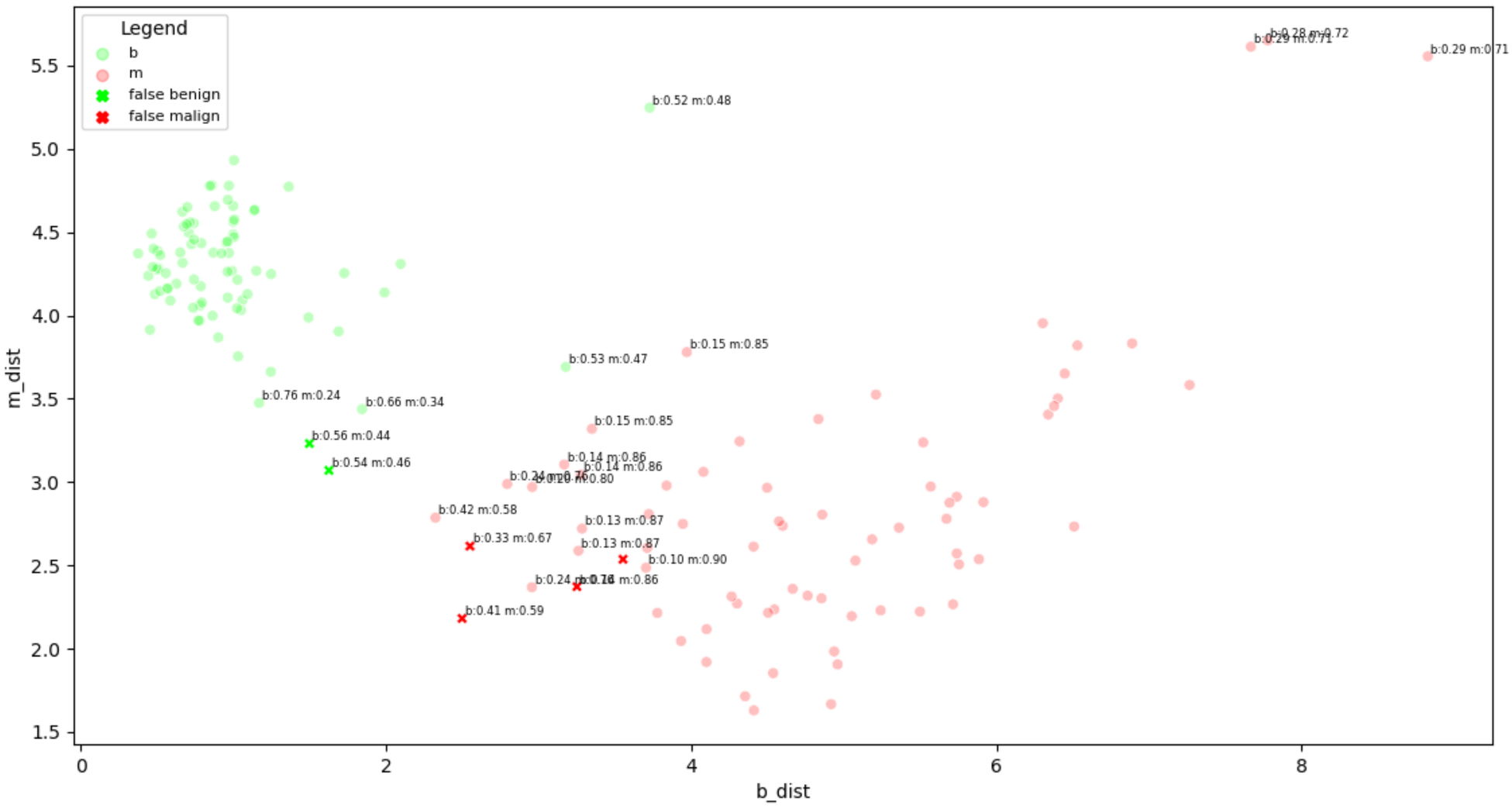
- Ensembling algorithm:
  - Use a set of decision trees.
- Each tree trained on a different subset of training data:
  - Bootstrap aggregation:
    - The subset are sampled with replacement.
- The final predicition is the label of the majority of trees in the forest.

| | Best param | Accuracy | Precision | | Recall | | F1 | | Support | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | Benign | Malign | Benign | Malign | Benign | Malign | Benign | Malign |
| Random Forest (Original feats.) | crit.: gini; max_dep.: 5; max_feat.: sqrt; min_samp_split: 2; n_est.: 500 | 0,95 | 0,96 | 0,94 | 0,95 | 0,96 | 0,96 | 0,95 | 82 | 71 |
| Random Forest (CD+original feats.) | crit.: gini; max_dep.: None; max_feat.: sqrt; min_samp_split: 5; n_est.: 50 | 0,95 | 0,96 | 0,94 | 0,95 | 0,96 | 0,96 | 0,95 | 82 | 71 |
| Random Forest (CD feats.) | crit.: gini; max_dep.: 2; max_feat.: sqrt; min_samp_split: 2; n_est.: 100 | 0,96 | 0,97 | 0,95 | 0,95 | 0,97 | 0,96 | 0,96 | 82 | 71 |

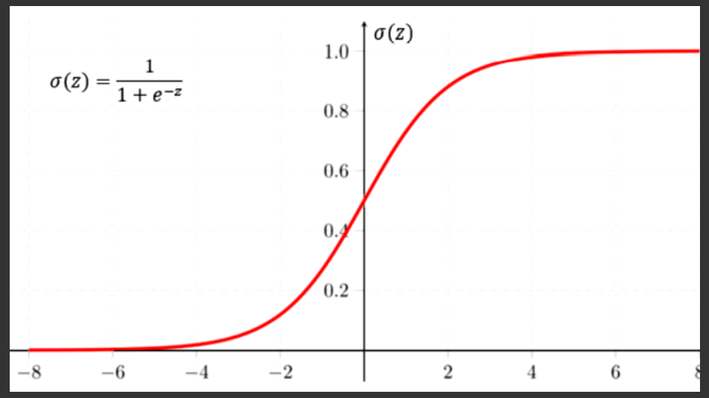
# RANDOM FOREST (CD FEATURES ONLY) TRAIN



# RANDOM FOREST (CD FEATURES ONLY) TEST



# MODELS & RESULTS

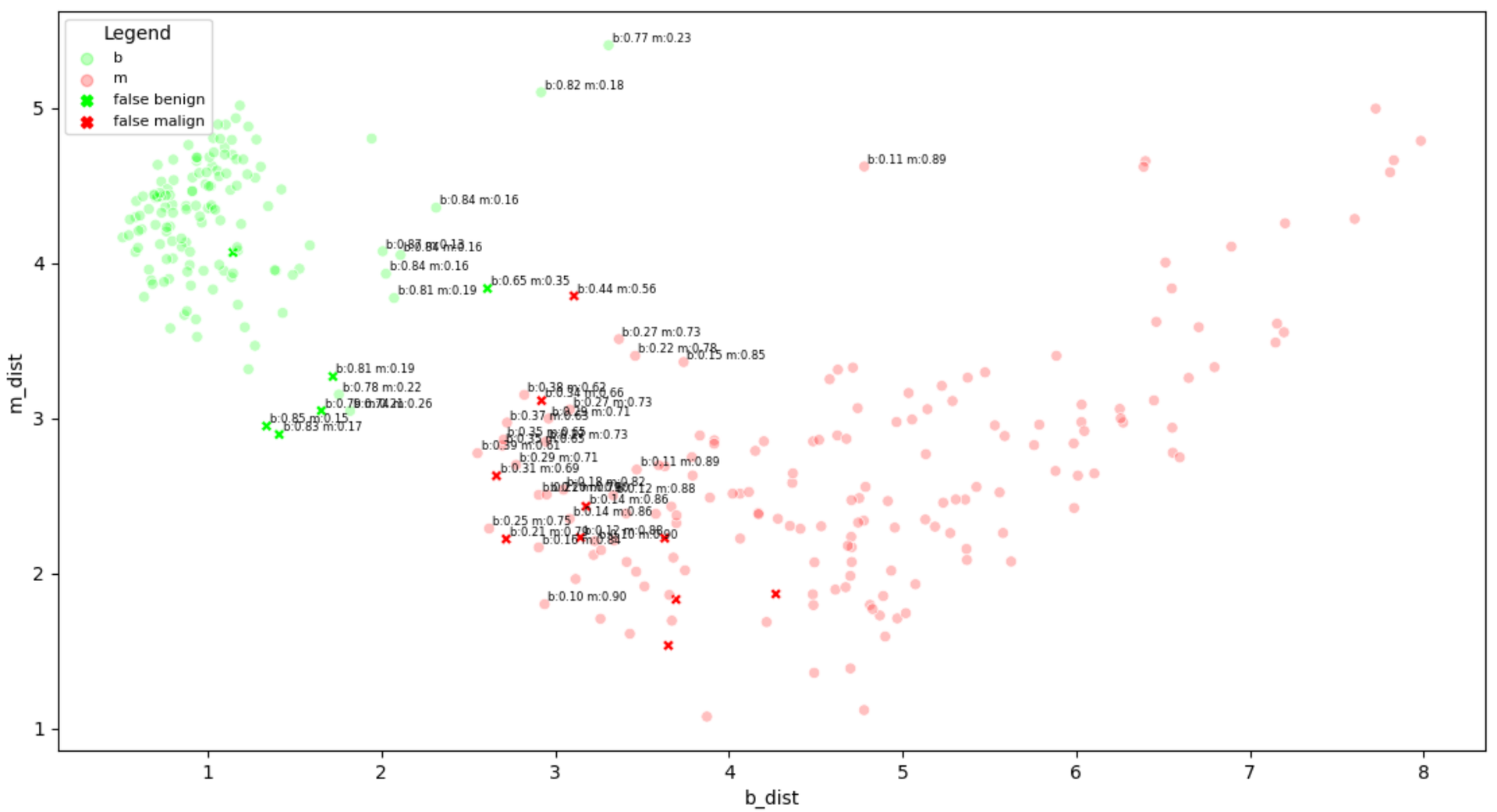



## LOGISTIC REGRESSION

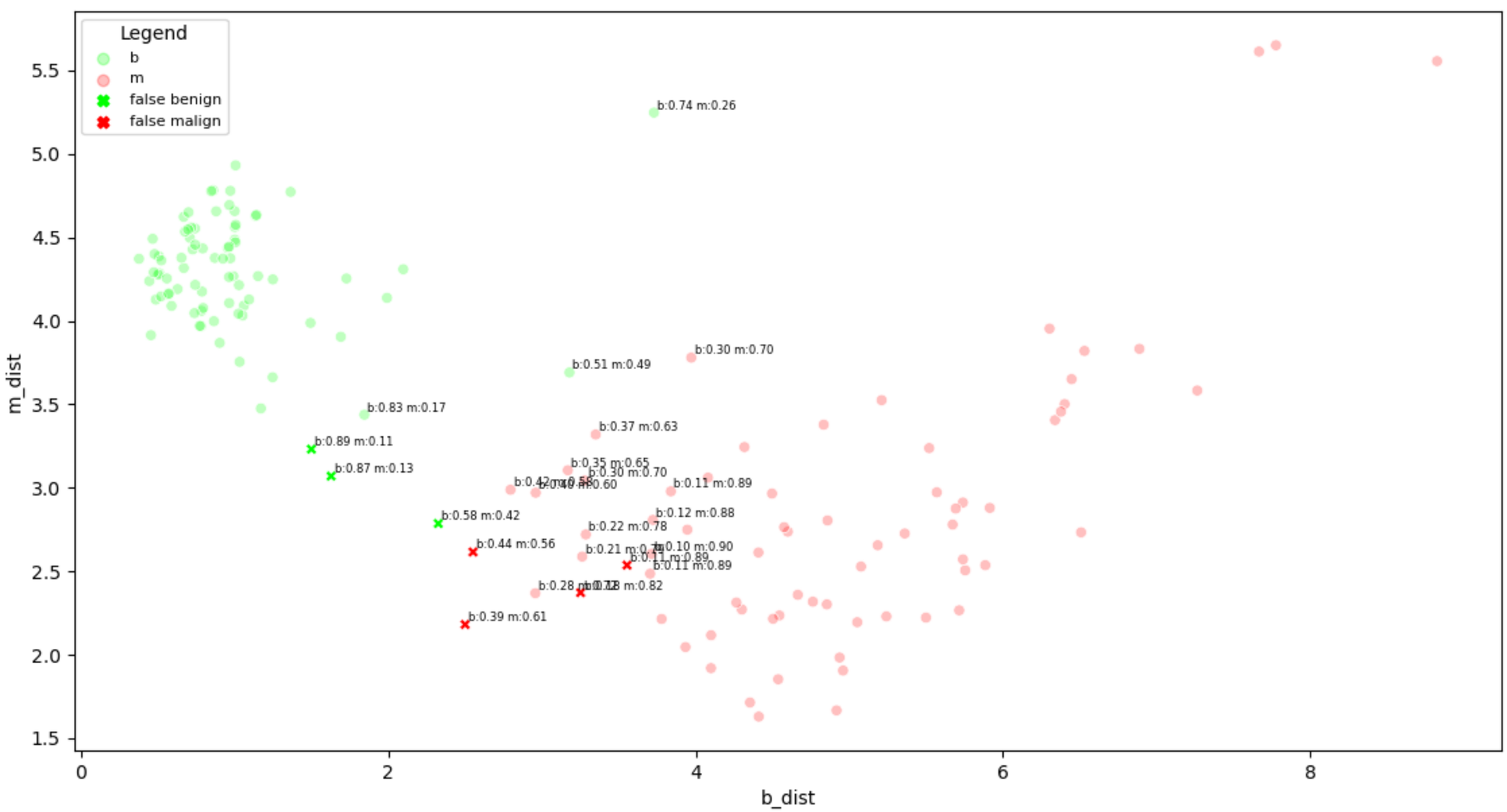
- Shape a linear combination *z* of the features.
  - Map *z* to a probability with a logistic function *σ(z)*.
    - Often the sigmoid function.
- Train the model with the "Maximum Likelihood Estimation".
  - Use likelihood function modeled with a Bernoulli distributions.

| | Best param | Accuracy | Precision | | Recall | | F1 | | Support | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | Benign | Malign | Benign | Malign | Benign | Malign | Benign | Malign |
| Log Regression (Original feat.) | C: 1; dual: False; max_iter: 500; penalty: none | 0,95 | 0,95 | 0,94 | 0,95 | 0,94 | 0,95 | 0,94 | 82 | 71 |
| Log Regression (CD+Original feat.) | C: 10; dual: False; max_iter: 500; penalty: l2 | 0,94 | 0,94 | 0,94 | 0,95 | 0,93 | 0,95 | 0,94 | 82 | 71 |
| Log Regression (CD feat.) | C: 1; dual: False; max_iter: 500; penalty: none | 0,95 | 0,96 | 0,94 | 0,95 | 0,96 | 0,96 | 0,95 | 82 | 71 |

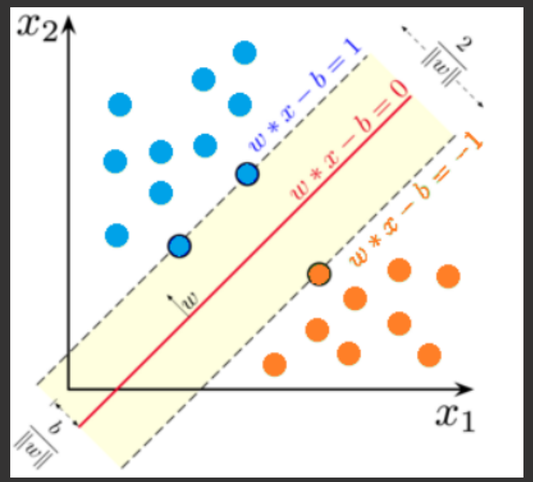
# LOGISTIC REGRESSION (CD FEATURES ONLY) TRAIN



# LOGISTIC REGRESSION (CD FEATURES ONLY) TEST



# MODELS & RESULTS

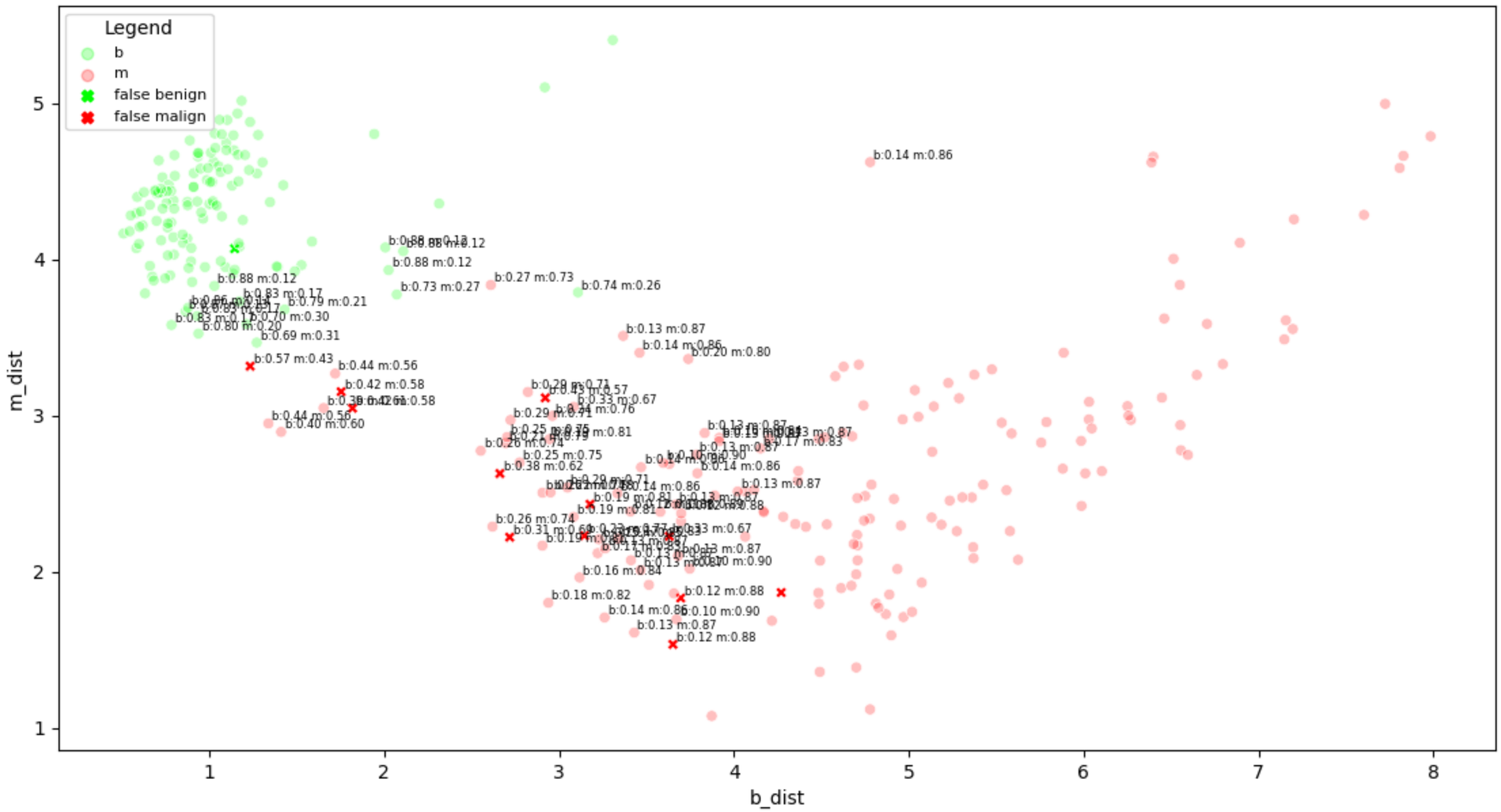



## SUPPORT VECTOR MACHINE

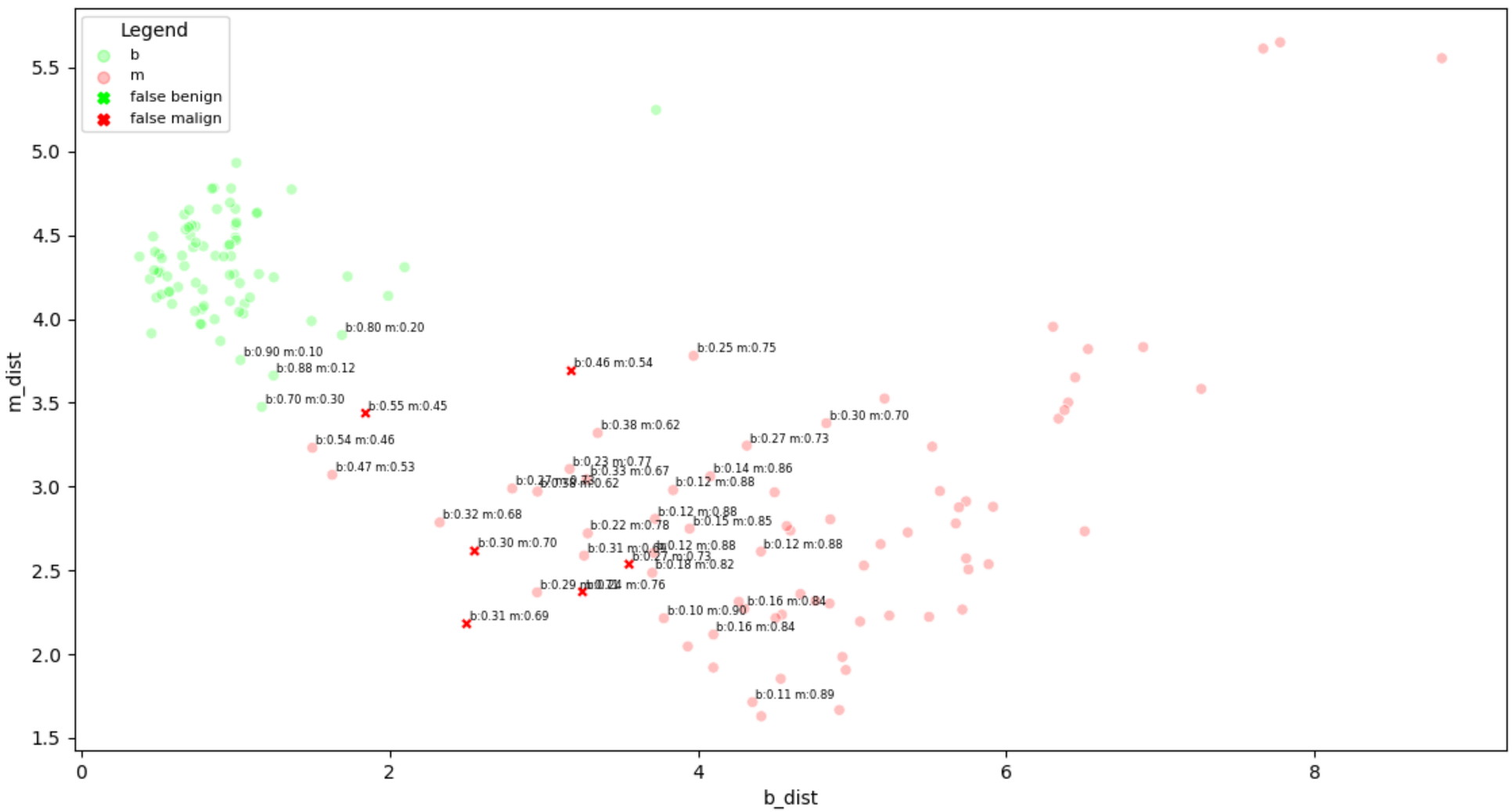
- Maximize the margin, to separate two different classes.
  - Margins are the (perpendicular) distances between a separating hyperplane and those dots (samples) closest to the hyperplane.
- A separation hyperplane may not exist:
  - Allow the maximization problem to make errors.
- Extention:
  - Use kernel methods to remap data in a higher (potentially unlimited) dimensionally space.
- The prediction is the sign of the projection of the sample on the hyperplane.

| | Best param | Accuracy | Precision | | Recall | | F1 | | Support | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | Benign | Malign | Benign | Malign | Benign | Malign | Benign | Malign |
| SVC (Original feats.) | C: 1; gamma: scale; kernel: poly | 0,96 | 1 | 0,92 | 0,93 | 1 | 0,96 | 0,96 | 82 | 71 |
| SVC (CD+Original feats.) | C: 1; gamma: scale; kernel: poly | 0,96 | 1 | 0,92 | 0,93 | 1 | 0,96 | 0,96 | 82 | 71 |
| SVC (CD feats.) | C: 0.1; gamma: scale; kernel: rbf | 0,96 | 0,97 | 0,95 | 0,95 | 0,97 | 0,96 | 0,96 | 82 | 71 |

# SVC (CD + ORIGINAL FEATURES) TRAIN



# SVC (CD + ORIGINAL FEATURES) TEST



# CONCLUSION

- CENTROIDS DISTANCES COULD BE A GOOD TECHNIQUE FOR DATA VISUALIZATIO AND TO IMPROVE THE RESULTS. BUT WE NEED TO PAY ATTENTION AT OVERFITTING.

- SVC ACHIEVES THE BEST SCORE, IN PARTICULAR ACHIEVES A PERFECT SCORE IN TERMS OF FALSE BENIGN.

- WE HAVE SEEN DIFFERENT TECHNIQUE FOR EACH MODEL TO RETRIEVE AN INDEX OF CERTAINTY, CRUCIAL IN MEDICAL APPLICATIONS.

- FURTHER WORK CAN FOCUS ON EXPLOITING OTHER KIND OF CENTROIDS: "THE CENTER OF THE SMALLEST ENCLOSING BALL" OR THE CENTER OF THE "SUPERVISED COMPACT HYPERSPHERE"

- THE OVERFITTING PHENOMENON CAN BE TESTED BETTER, BY PLOTTING THE TRAINING SCORE WHEN PARAMETERS CHANGES.

- OTHER KIND OF HYPERPARAMETER TUNING CAN BE EXPLOITED TO TEST THE MODELS WITH A MORE WIDE PARAMETERS SPECTRUM.


T. HANKS