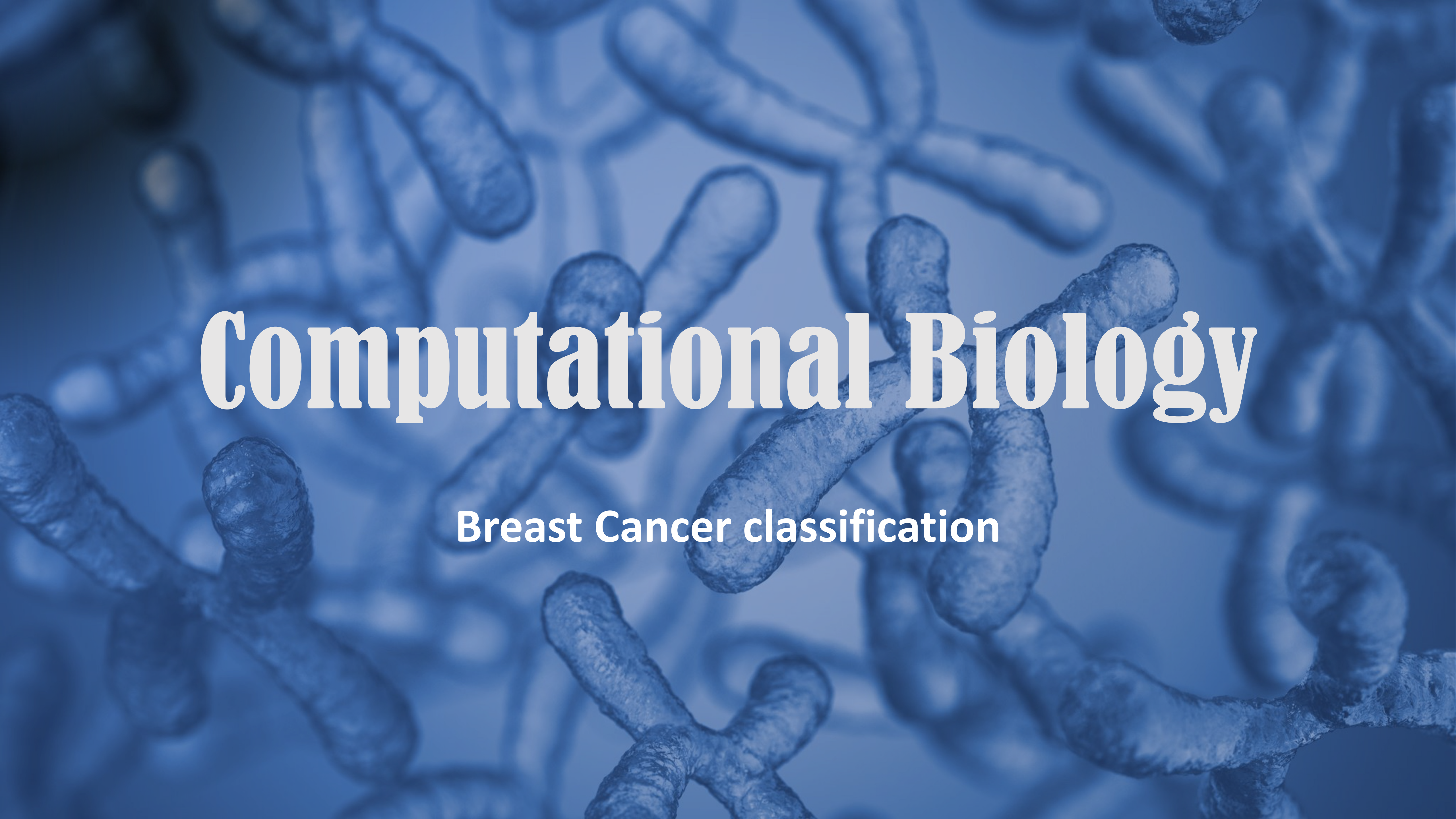


Computational Biology

Breast Cancer classification

Outline

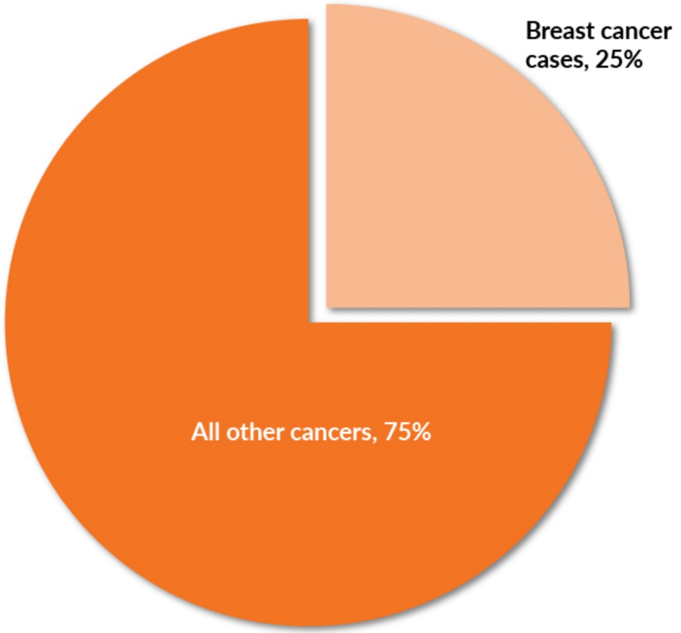
- INTRODUCTION
- DATASET DESCRIPTION
- PROJECT STRUCTURE
- IMPORTANT DATES

Breast Cancer

It is estimated that in 2022:

- 28,600 Canadian women will be diagnosed with breast cancer. This represents 25% of all new cancer cases in women in 2022.
- 5,500 Canadian women will die from breast cancer. This represents 14% of all cancer deaths in women in 2022.
- On average, 78 Canadian women will be diagnosed with breast cancer every day.
- On average, 15 Canadian women will die from breast cancer every day.
- 270 Canadian men will be diagnosed with breast cancer and 55 will die from breast cancer.

Percentage of All Estimated New Cancer Cases in Women in 2022



© Canadian Cancer Society

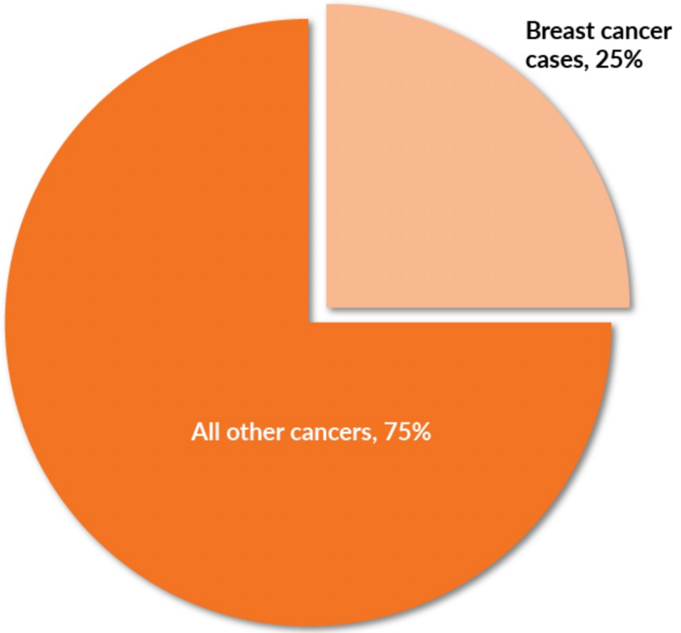
Breast Cancer

It is estimated that in 2022:

- 28,600 Canadian women will be diagnosed with breast cancer. This represents 25% of all new cancer cases in women in 2022.
- 5,500 Canadian women will die from breast cancer. This represents 14% of all cancer deaths in women in 2022.
- On average, 78 Canadian women will be diagnosed with breast cancer every day.
- On average, 15 Canadian women will die from breast cancer every day.
- 270 Canadian men will be diagnosed with breast cancer and 55 will die from breast cancer.

Motivation: the early diagnosis of breast cancer can improve the prognosis and chance of survival significantly, as it can promote timely clinical treatment to patients.

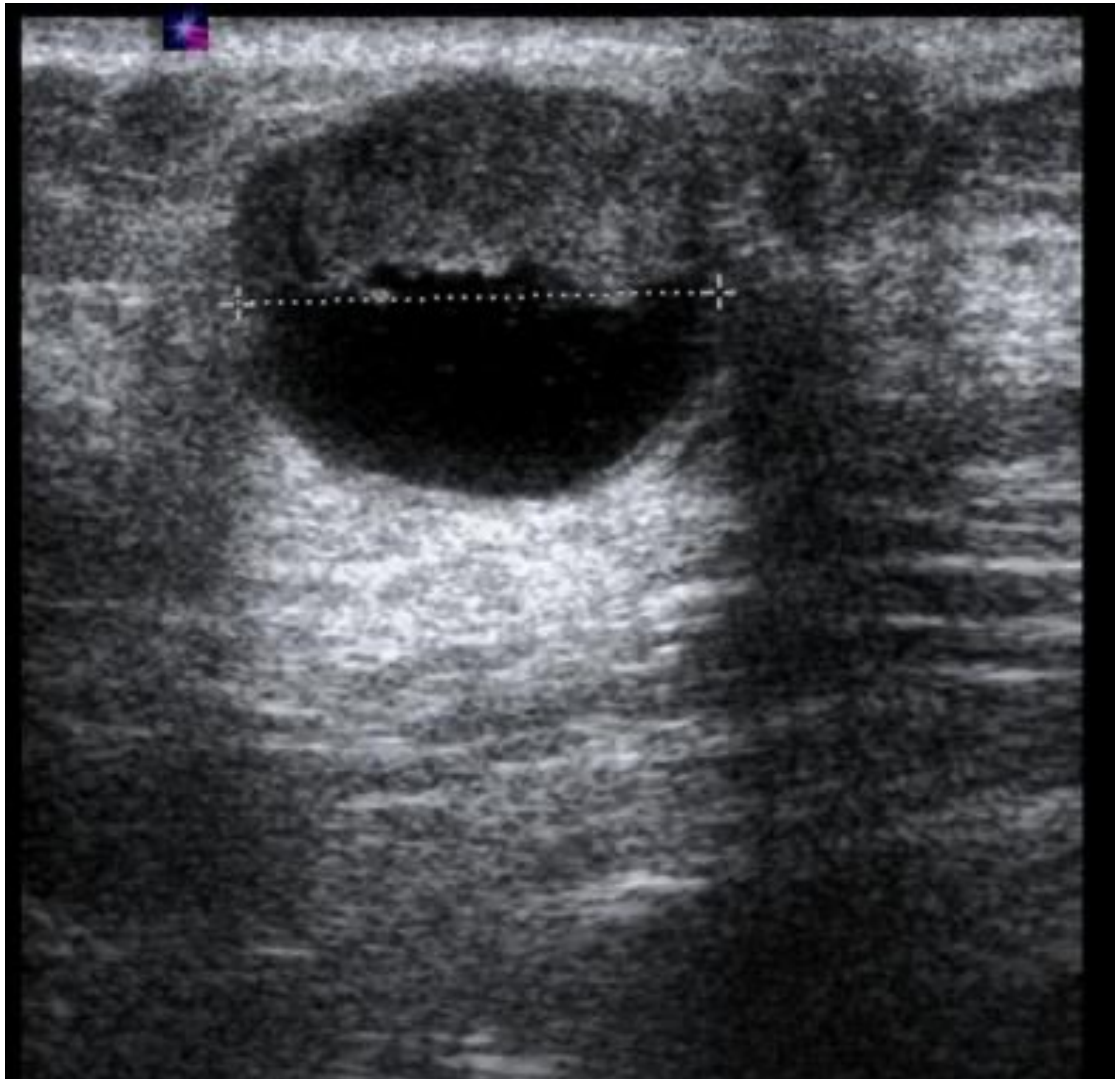
Percentage of All Estimated New Cancer Cases in Women in 2022



© Canadian Cancer Society

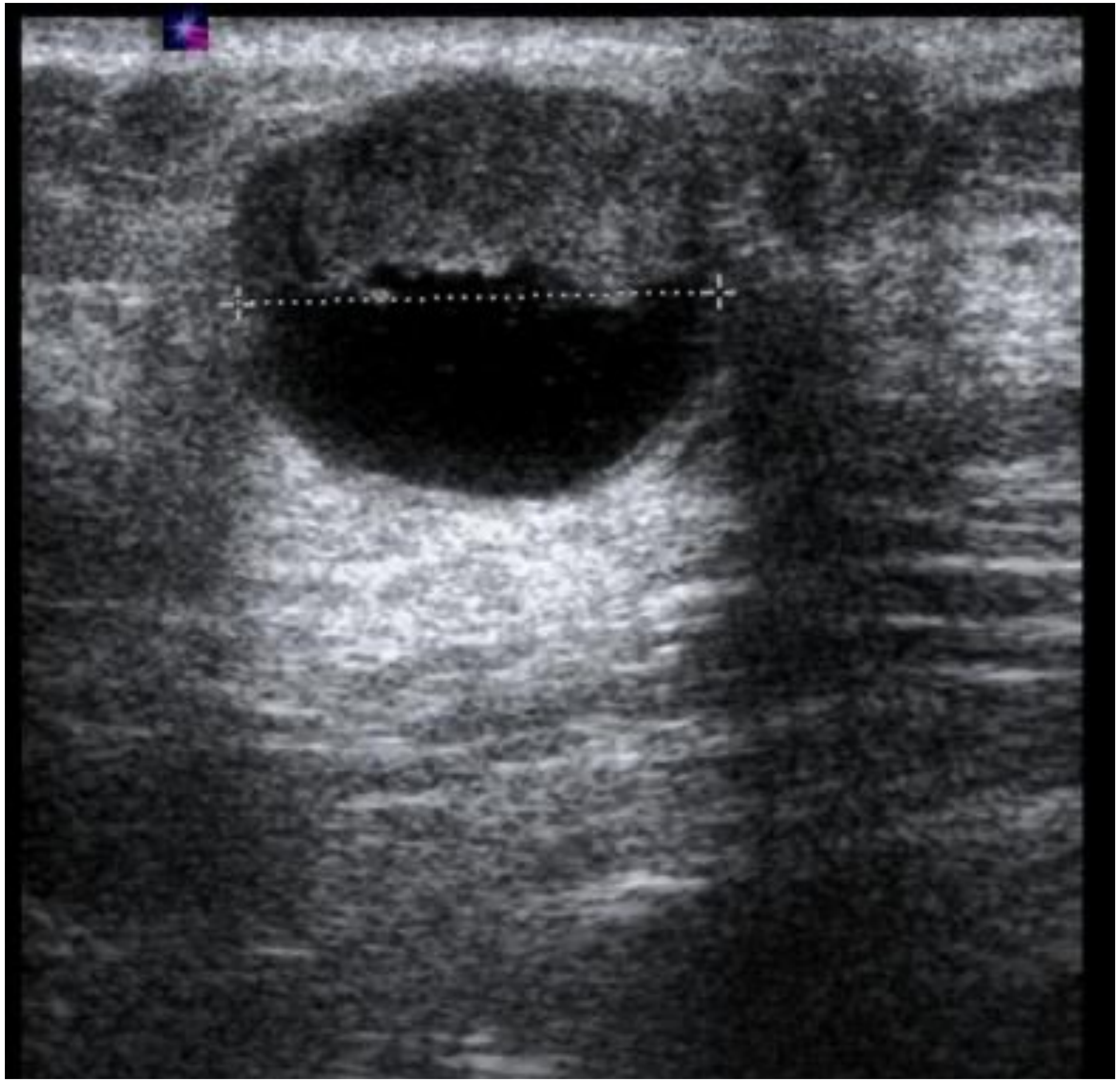
Breast Cancer Wisconsin (Diagnostic) Data Set

- The dataset is publicly available and was created by Dr. William H. Wolberg, physician at the University Of Wisconsin Hospital at Madison, Wisconsin, USA.
- To create the dataset Dr. Wolberg used fluid samples, taken from patients with solid breast masses and an easy-to-use graphical computer program called Xcyt



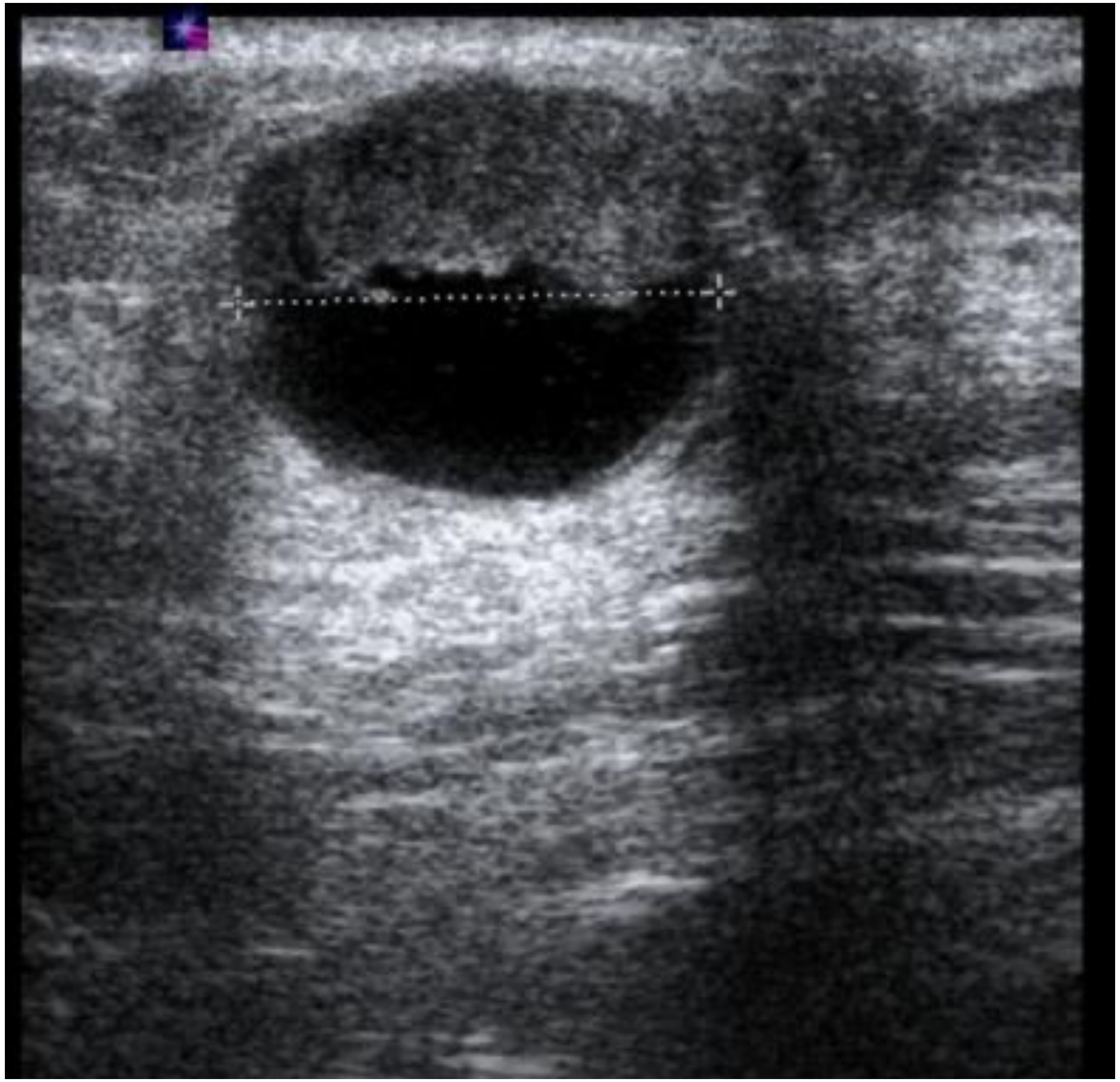
Breast Cancer Wisconsin (Diagnostic) Data Set

- ID number
- Diagnosis (M = malignant, B = benign)
- radius
- texture
- perimeter
- area
- smoothness (local variation in radius lengths)
- compactness (perimeter^2 / area — 1.0)
- concavity (severity of concave portions of the contour)
- concave points (number of concave portions of the contour)
- symmetry
- fractal dimension



Breast Cancer Wisconsin (Diagnostic) Data Set

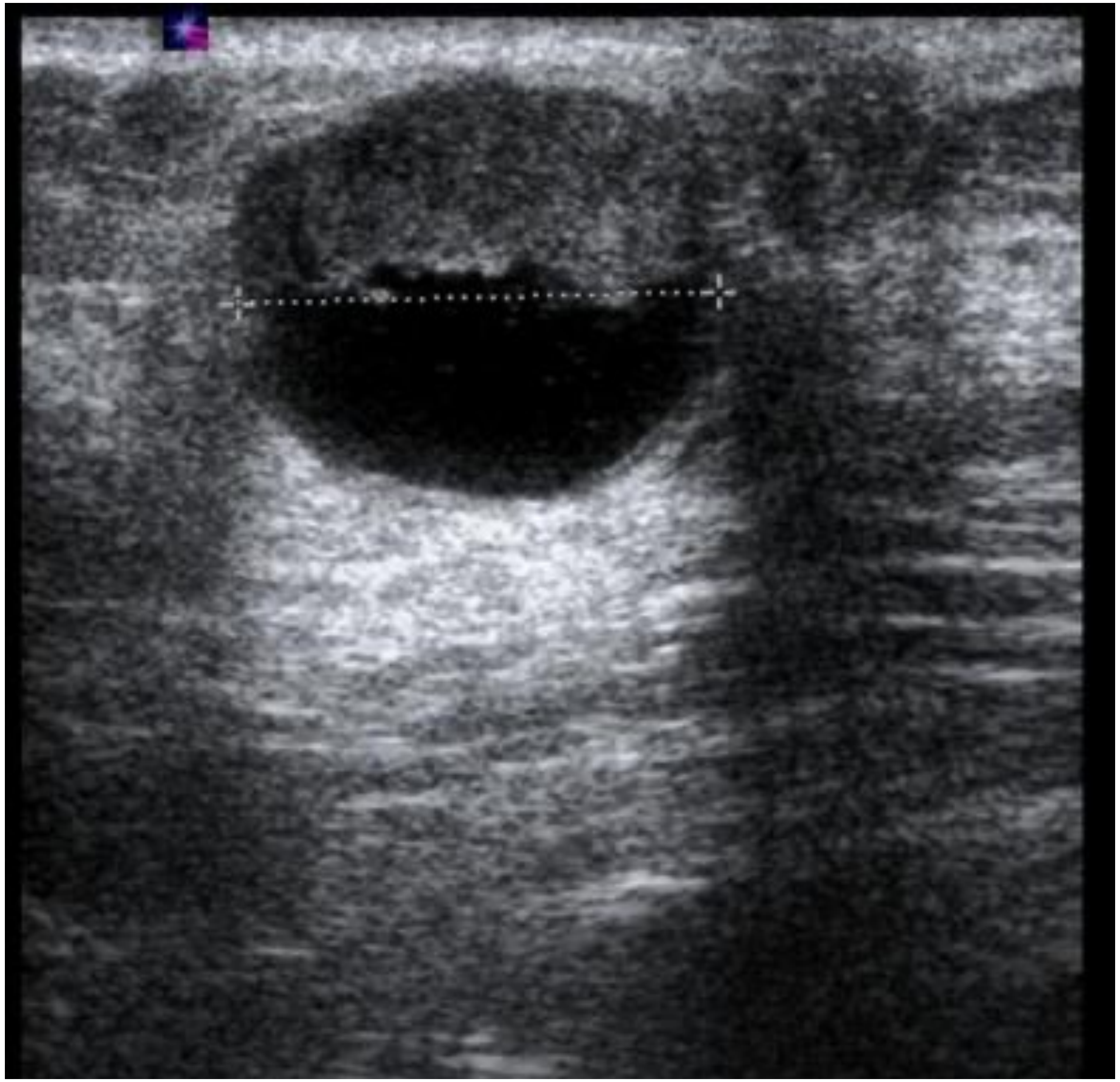
- 33 columns
- 569 rows (357 benign, 212 malignant)
- Numerical (int, float), categorical (b, m)



Breast Cancer Wisconsin (Diagnostic) Data Set

- 33 columns
- 569 rows (357 benign, 212 malignant)
- Numerical (int, float), categorical (b, m)

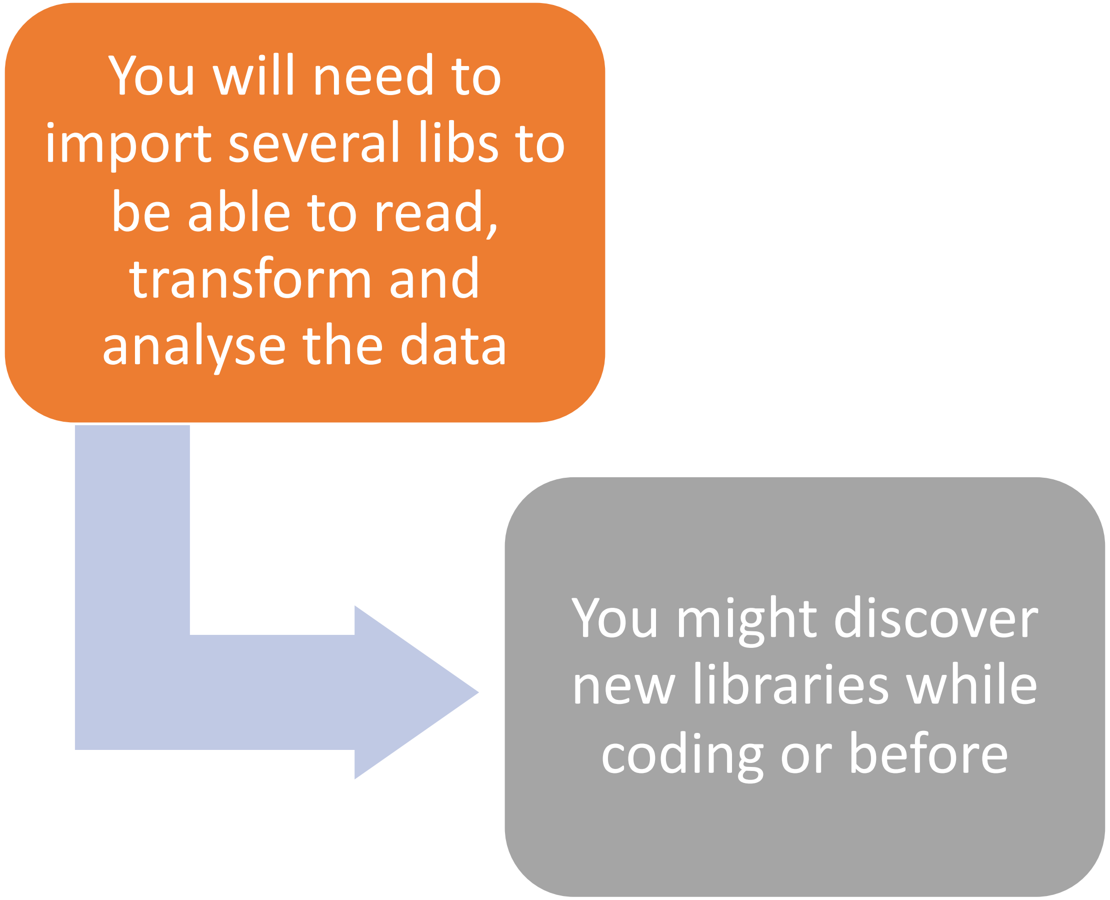
Aim: we want to be able to classify the diagnosis as benign or malignant



Project Structure

- Import libraries & dataset
- Preprocess data
- Analyze data
- Choose features
- Create and evaluate models
- Choose best model

Import necessary libraries



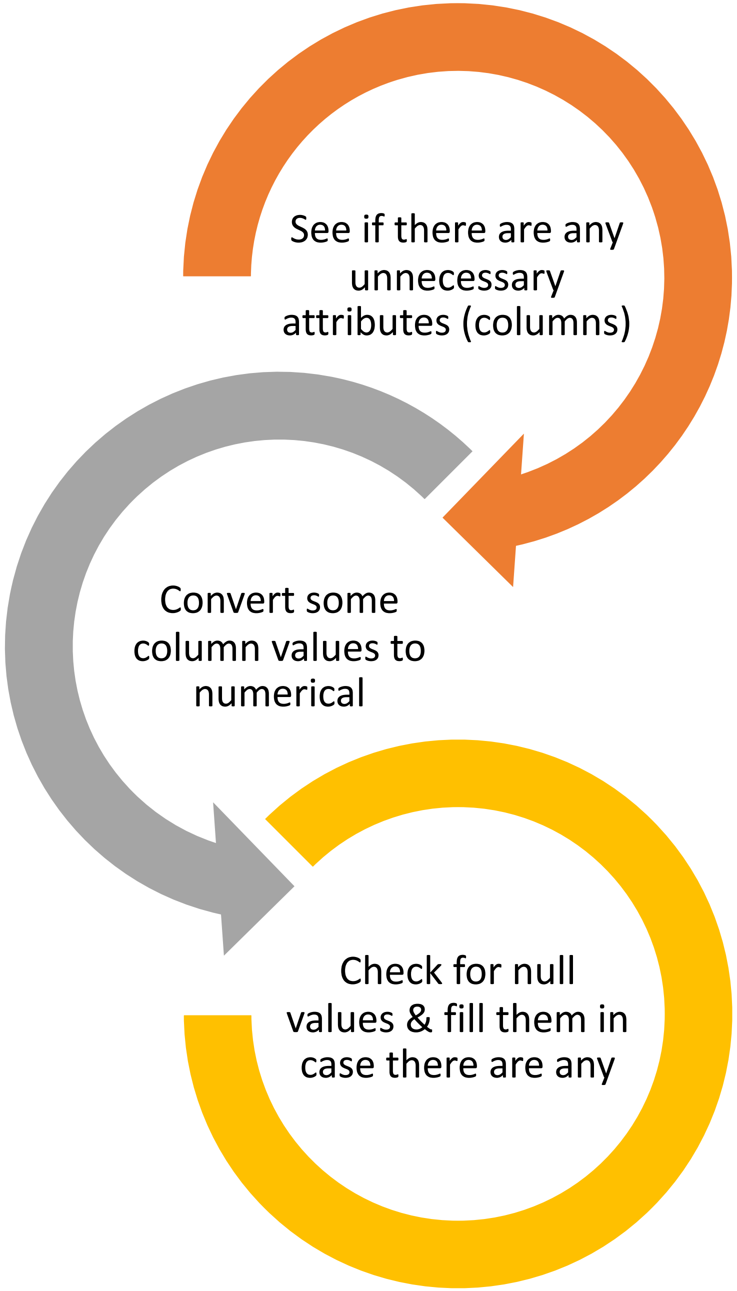
You will need to import several libs to be able to read, transform and analyse the data

You might discover new libraries while coding or before

Here are some libraries you might want to consider:

- pandas
- numpy
- seaborn
- matplotlib.pyplot
- missingno
- sklearn

Preprocess data



See if there are any unnecessary attributes (columns)

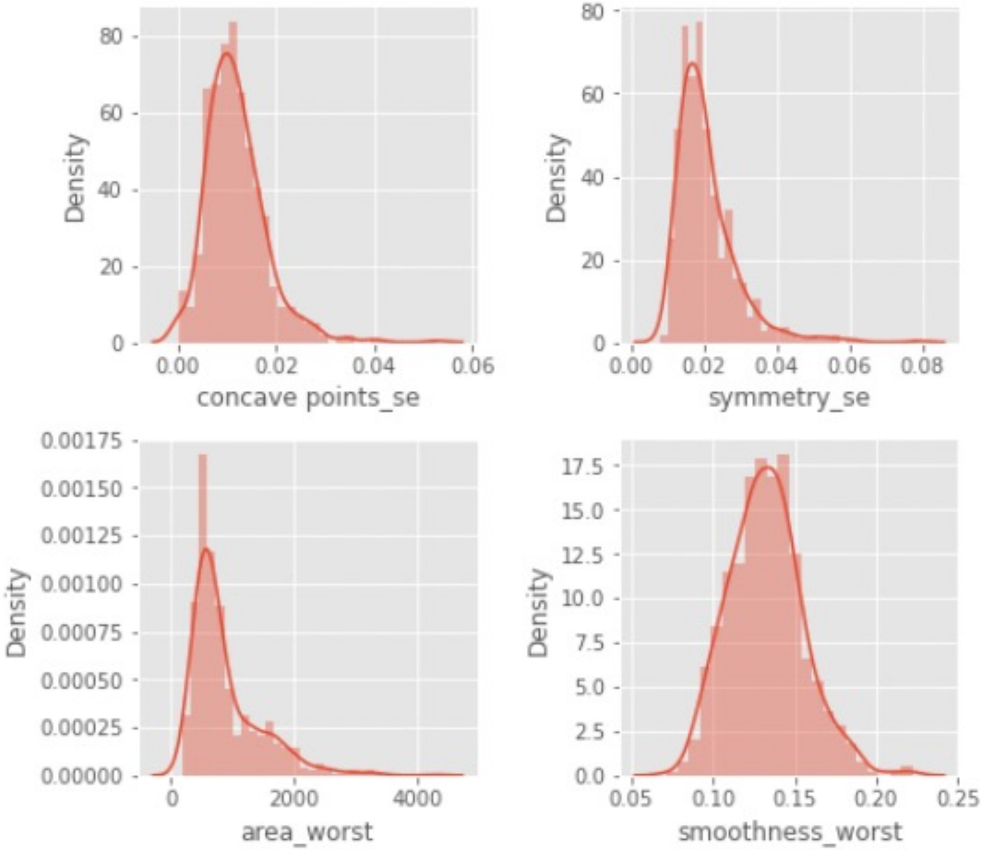
Convert some column values to numerical

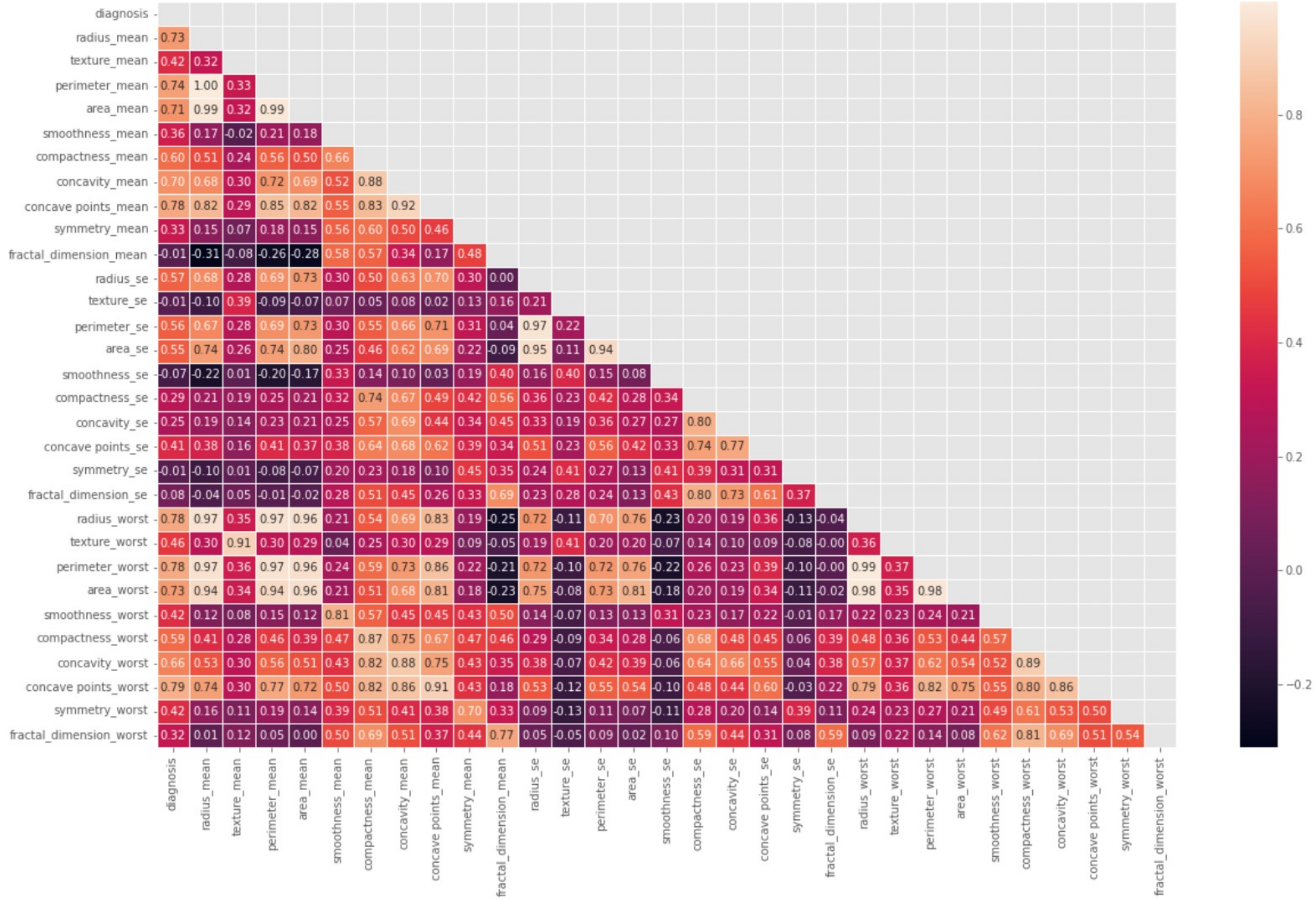
Check for null values & fill them in case there are any

Here are some pandas and missingno tools you might find useful:

- df.drop([‘Columns’])
- df[‘column'].apply(lambda val: 1 *condition* else 0)
- df.describe()
- df.info()
- df.isna().sum()
- msno.bar(df)

Analyze data



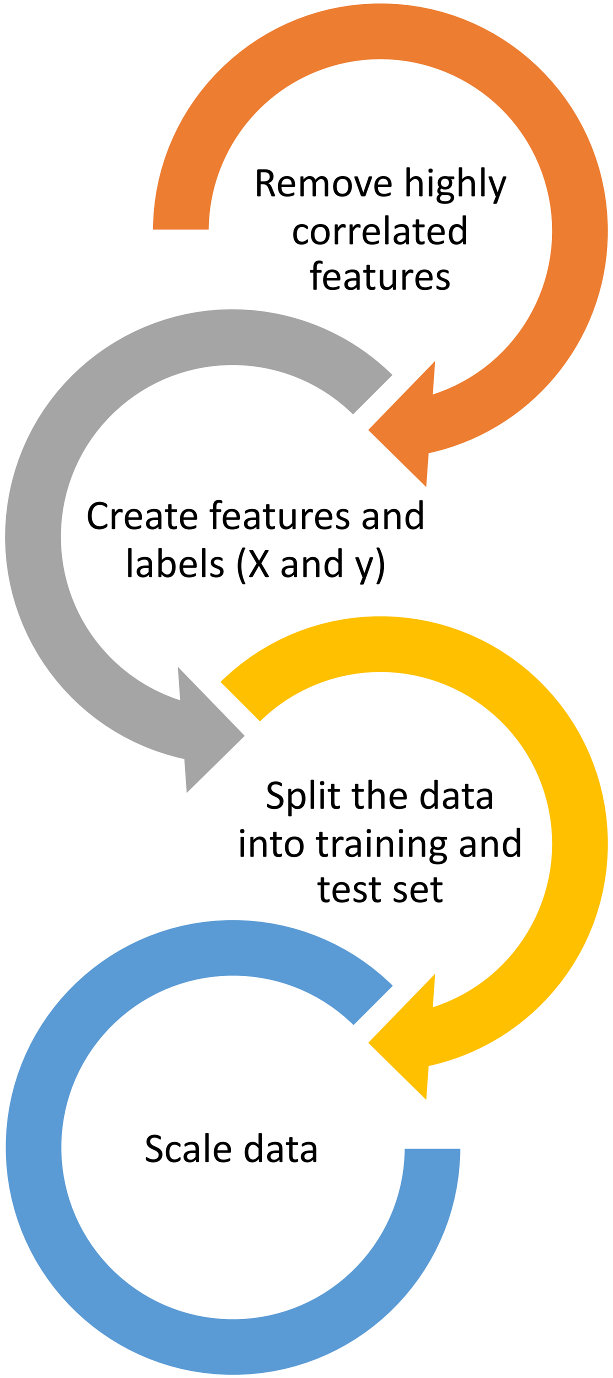


	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	symmetry_mean	fractal_dimension_mean	radius_se	texture_se	perimeter_se	area_se	smoothness_se	compactness_se	concavity_se	concave points_se	symmetry_se	fractal_dimension_se	radius_worst	texture_worst	perimeter_worst	area_worst	smoothness_worst	compactness_worst	concavity_worst	concave points_worst	symmetry_worst	fractal_dimension_worst
diagnosis																															
radius_mean	0.73																														
texture_mean	0.42	0.32																													
perimeter_mean	0.74	1.00	0.33																												
area_mean	0.71	0.99	0.32	0.99																											
smoothness_mean	0.36	0.17	-0.02	0.21	0.18																										
compactness_mean	0.60	0.51	0.24	0.56	0.50	0.66																									
concavity_mean	0.70	0.68	0.30	0.72	0.69	0.52	0.88																								
concave points_mean	0.78	0.82	0.29	0.85	0.82	0.55	0.83	0.92																							
symmetry_mean	0.33	0.15	0.07	0.18	0.15	0.56	0.60	0.50	0.46																						
fractal_dimension_mean	-0.01	-0.31	-0.08	-0.26	-0.28	0.58	0.57	0.34	0.17	0.48																					
radius_se	0.57	0.68	0.28	0.69	0.73	0.30	0.50	0.63	0.70	0.30	0.00																				
texture_se	-0.01	-0.10	0.39	-0.09	-0.07	0.07	0.05	0.08	0.02	0.13	0.16	0.21																			
perimeter_se	0.56	0.67	0.28	0.69	0.73	0.30	0.55	0.66	0.71	0.31	0.04	0.97	0.22																		
area_se	0.55	0.74	0.26	0.74	0.80	0.25	0.46	0.62	0.69	0.22	-0.09	0.95	0.11	0.94																	
smoothness_se	-0.07	-0.22	0.01	-0.20	-0.17	0.33	0.14	0.10	0.03	0.19	0.40	0.16	0.40	0.15	0.08																
compactness_se	0.29	0.21	0.19	0.25	0.21	0.32	0.74	0.67	0.49	0.42	0.56	0.36	0.23	0.42	0.28	0.34															
concavity_se	0.25	0.19	0.14	0.23	0.21	0.25	0.57	0.69	0.44	0.34	0.45	0.33	0.19	0.36	0.27	0.27	0.80														
concave points_se	0.41	0.38	0.16	0.41	0.37	0.38	0.64	0.68	0.62	0.39	0.34	0.51	0.23	0.56	0.42	0.33	0.74	0.77													
symmetry_se	-0.01	-0.10	0.01	-0.08	-0.07	0.20	0.23	0.18	0.10	0.45	0.35	0.24	0.41	0.27	0.13	0.41	0.39	0.31	0.31												
fractal_dimension_se	0.08	-0.04	0.05	-0.01	-0.02	0.28	0.51	0.45	0.26	0.33	0.69	0.23	0.28	0.24	0.13	0.43	0.80	0.73	0.61	0.37											
radius_worst	0.78	0.97	0.35	0.97	0.96	0.21	0.54	0.69	0.83	0.19	-0.25	0.72	-0.11	0.70	0.76	-0.23	0.20	0.19	0.36	-0.13	-0.04										
texture_worst	0.46	0.30	0.91	0.30	0.29	0.04	0.25	0.30	0.29	0.09	-0.05	0.19	0.41	0.20	0.20	-0.07	0.14	0.10	0.09	-0.08	-0.00	0.36									
perimeter_worst	0.78	0.97	0.36	0.97	0.96	0.24	0.59	0.73	0.86	0.22	-0.21	0.72	-0.10	0.72	0.76	-0.22	0.26	0.23	0.39	-0.10	-0.00	0.99	0.37								
area_worst	0.73	0.94	0.34	0.94	0.96	0.21	0.51	0.68	0.81	0.18	-0.23	0.75	-0.08	0.73	0.81	-0.18	0.20	0.19	0.34	-0.11	-0.02	0.98	0.35	0.98							
smoothness_worst	0.42	0.12	0.08	0.15	0.12	0.81	0.57	0.45	0.45	0.43	0.50	0.14	-0.07	0.13	0.13	0.31	0.23	0.17	0.22	-0.01	0.17	0.22	0.23	0.24	0.21						
compactness_worst	0.59	0.41	0.28	0.46	0.39	0.47	0.87	0.75	0.67	0.47	0.46	0.29	-0.09	0.34	0.28	-0.06	0.68	0.48	0.45	0.06	0.39	0.48	0.36	0.53	0.44	0.57					
concavity_worst	0.66	0.53	0.30	0.56	0.51	0.43	0.82	0.88	0.75	0.43	0.35	0.38	-0.07	0.42	0.39	-0.06	0.64	0.66	0.55	0.04	0.38	0.57	0.37	0.62	0.54	0.52	0.89				
concave points_worst	0.79	0.74	0.30	0.77	0.72	0.50	0.82	0.86	0.91	0.43	0.18	0.53	-0.12	0.55	0.54	-0.10	0.48	0.44	0.60	-0.03	0.22	0.79	0.36	0.82	0.75	0.55	0.80	0.86			
symmetry_worst	0.42	0.16	0.11	0.19	0.14	0.39	0.51	0.41	0.38	0.70	0.33	0.09	-0.13	0.11	0.07	-0.11	0.28	0.20	0.14	0.39	0.11	0.24	0.23	0.27	0.21	0.49	0.61	0.53	0.50		
fractal_dimension_worst	0.32	0.01	0.12	0.05	0.00	0.50	0.69	0.51	0.37	0.44	0.77	0.05	-0.05	0.09	0.02	0.10	0.59	0.44	0.31	0.08	0.59	0.09	0.22	0.14	0.08	0.62	0.81	0.69	0.51	0.54	

Plot the distribution of each column

Plot the correlation matrix to see highly correlated features

Choose features & transform data



Remove highly correlated features

Create features and labels (X and y)

Split the data into training and test set

Scale data

You might find these useful:

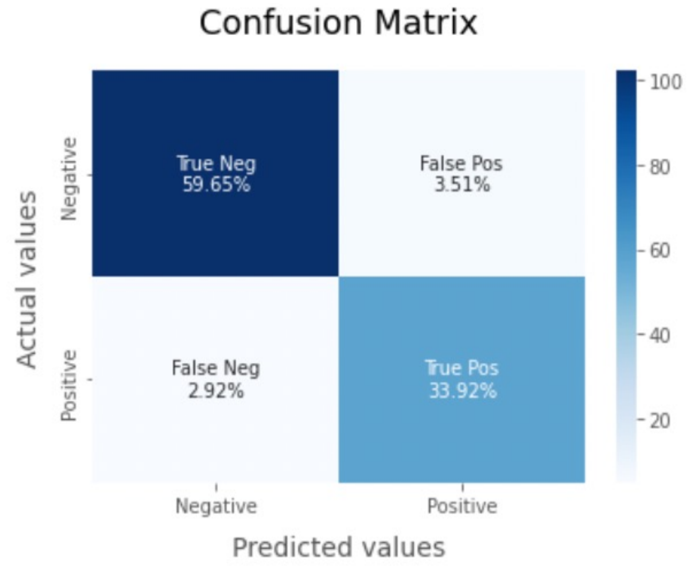
- from sklearn.model_selection import train_test_split
- X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.30)

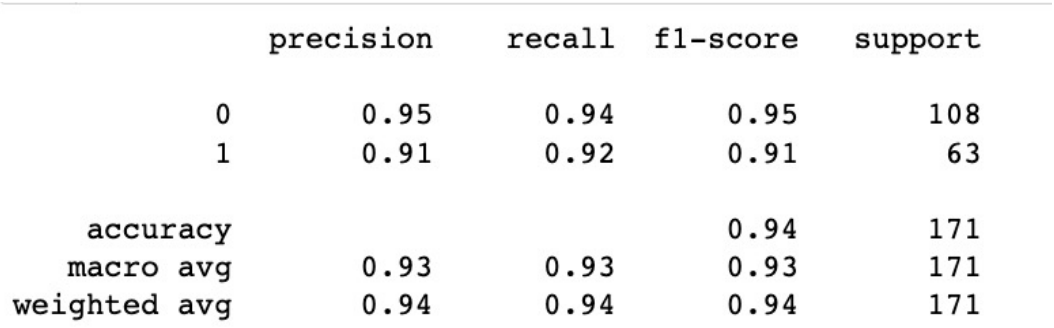
- from sklearn.preprocessing import StandardScaler
- scaler = StandardScaler()
- X_train = scaler.fit_transform(X_train)
- X_test = scaler.transform(X_test)

Create and evaluate models

```
def model_Evaluate(model):

   # Predict values for Test dataset
   # Print the evaluation metrics for the dataset.
   # Compute and plot the Confusion matrix
```

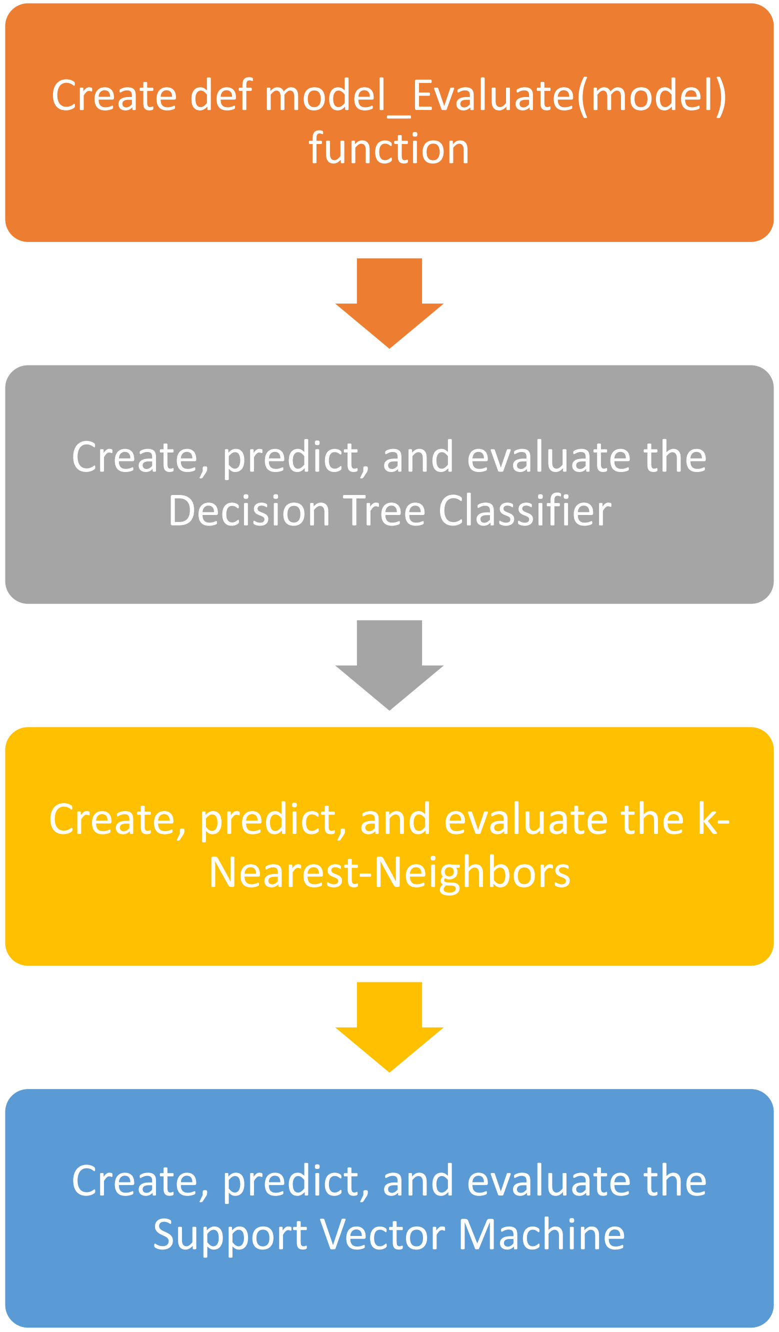




```
              precision    recall  f1-score   support

           0       0.95      0.94      0.95       108
           1       0.91      0.92      0.91        63

    accuracy                           0.94       171
   macro avg       0.93      0.93      0.93       171
weighted avg       0.94      0.94      0.94       171
```



Create def model_Evaluate(model) function

Create, predict, and evaluate the Decision Tree Classifier

Create, predict, and evaluate the k-Nearest-Neighbors

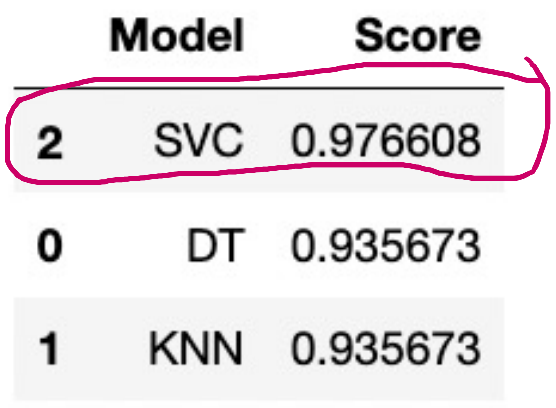
Create, predict, and evaluate the Support Vector Machine

Choose best model

Sort out the accuracies of each model

Compare them

Choose the best model



	Model	Score
2	SVC	0.976608
0	DT	0.935673
1	KNN	0.935673

Important dates

Task	Week number	Deadline
Data preprocesssing	Week 1	7/13/2022
Analyzing data	Week 1	7/15/2022
Choosing features and transforming data	Week 2	7/21/2022
Model creation and evaluation	Week 2	7/22/2022
Choosing the best model	Week 3	7/28/2022
Project presentation	Week 3	7/29/2022

Reference

- https://radiopaedia.org/articles/complex-cystic-and-solid-breast-mass
- https://cancer.ca/en/cancer-information/cancer-types/breast/statistics