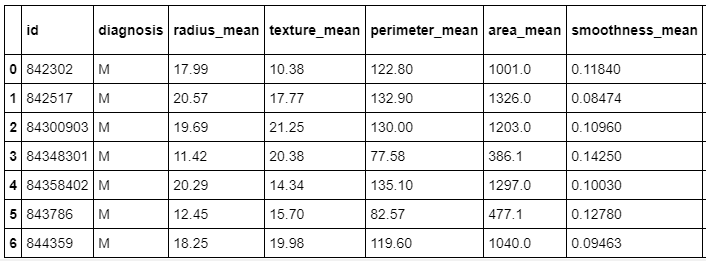
**1.1 Load the data,** and print the first 7 rows of data.

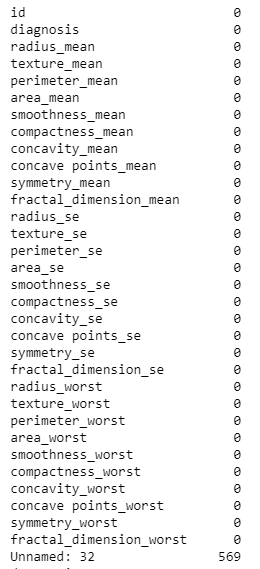
Each row of data represents a patient that may or may not have cancer.



Explore the data and **count the number of rows and columns** in the data set. Their are 569 rows of data which means their are 569 patients in this data set, and 33 columns which mean their are 33 features or data points for each patient.

https://miro.medium.com/max/91/1*WyHpp5Fyx-gMdP6Xx0hodw.png

Continue exploring the data and **get a count of all of the columns that contain empty (NaN, NAN, na) values.** Notice none of the columns contain any empty values except the column named ‘Unnamed: 32’ , which contains 569 empty values (the same number of rows in the data set, this tells me this column is completely useless)



**Remove the column ‘Unnamed: 32’** from the original data set since it adds no value.

**Get the new count** of the number of rows and columns.

https://miro.medium.com/max/87/1*nvQZezq-m34zgXtvStPQwg.png

**Get a count** of the number of patients with Malignant (M) cancerous and Benign (B) non-cancerous cells.

https://miro.medium.com/max/60/1*-tWG9_ZYro7SvQMWoRHHoQ.png?q=20

**# of Cancerous Cells:** 212 and **# of Non-Cancerous Cells:** 357

**Visualize the counts**, by creating a count plot.

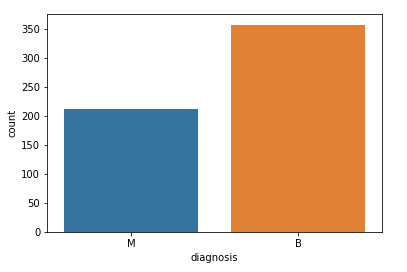
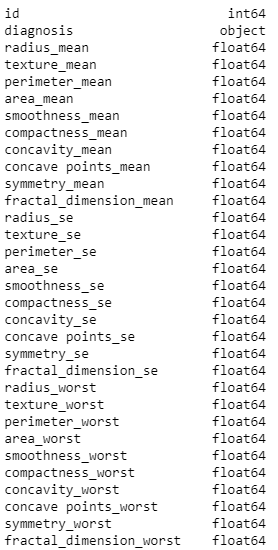


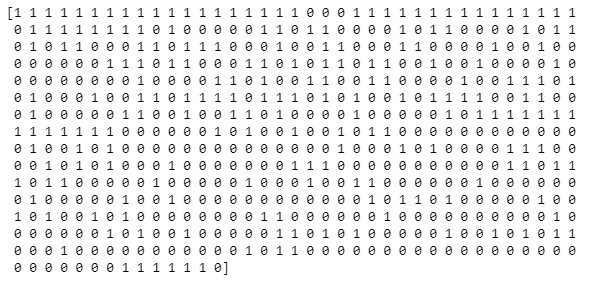
Chart displaying Malignant (cancerous) & Benign(non-cancerous) diagnosis

The data types to see which **columns need to be transformed / encoded**. We can see from the data types that all of the columns/features are numbers except for the column ‘diagnosis’, which is categorical data represented as an object in python.



A list of the columns & their data types

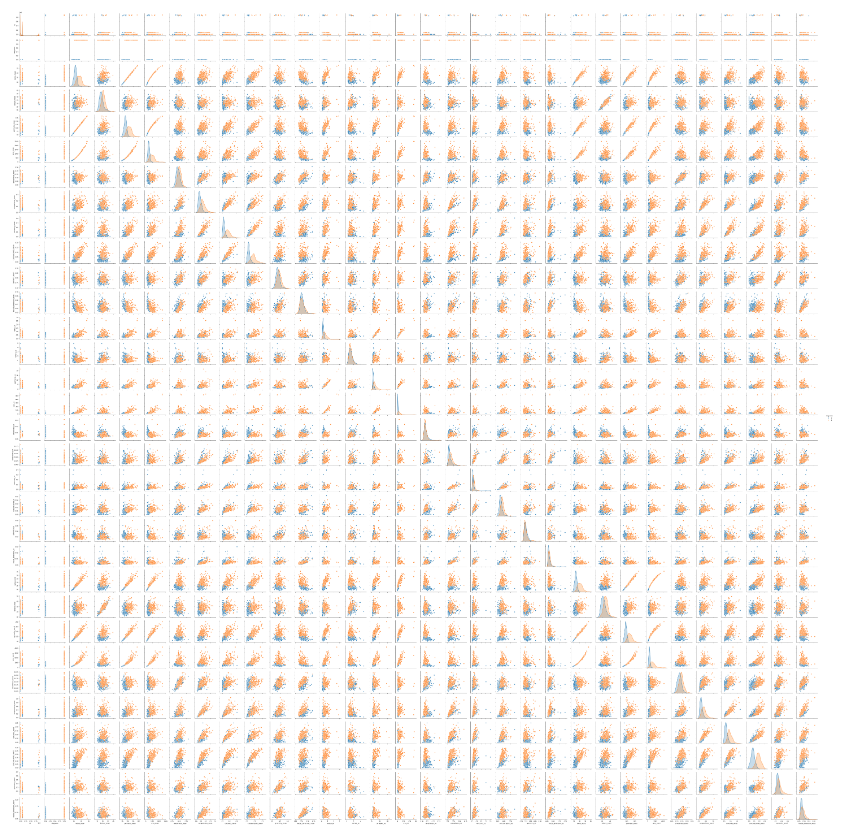
**Encode the categorical data**. Change the values in the column ‘diagnosis’ from **M and B** to **1 and 0** respectively, then print the results.



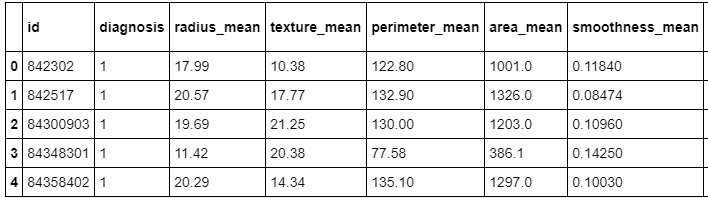
The encoded values of the feature/column diagnosis.

**1.2 CORRELATION OF DATASET**

**Create a pair plot**. A “pairs plot” is also known as a scatter plot, in which one variable in the same data row is matched with another variable’s value.

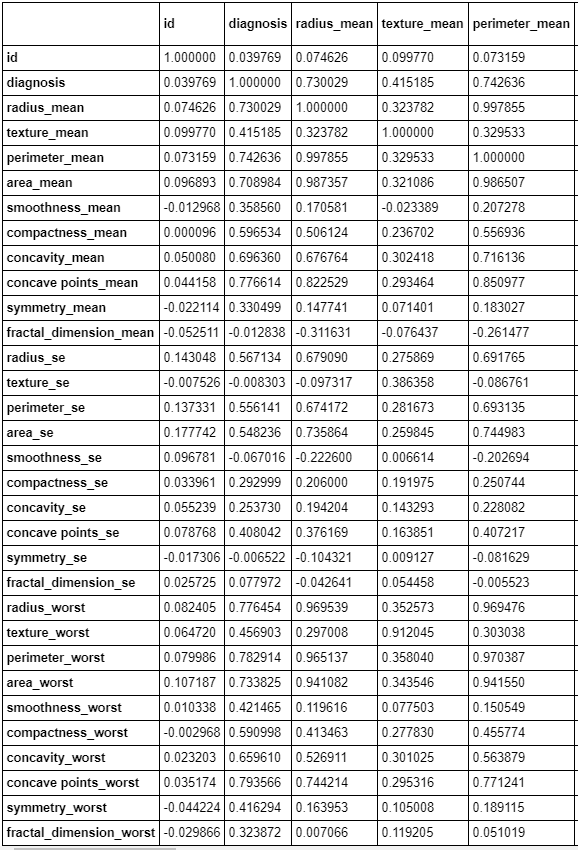


**Print the new data set** which now has only 32 columns. Print only the first 5 rows.



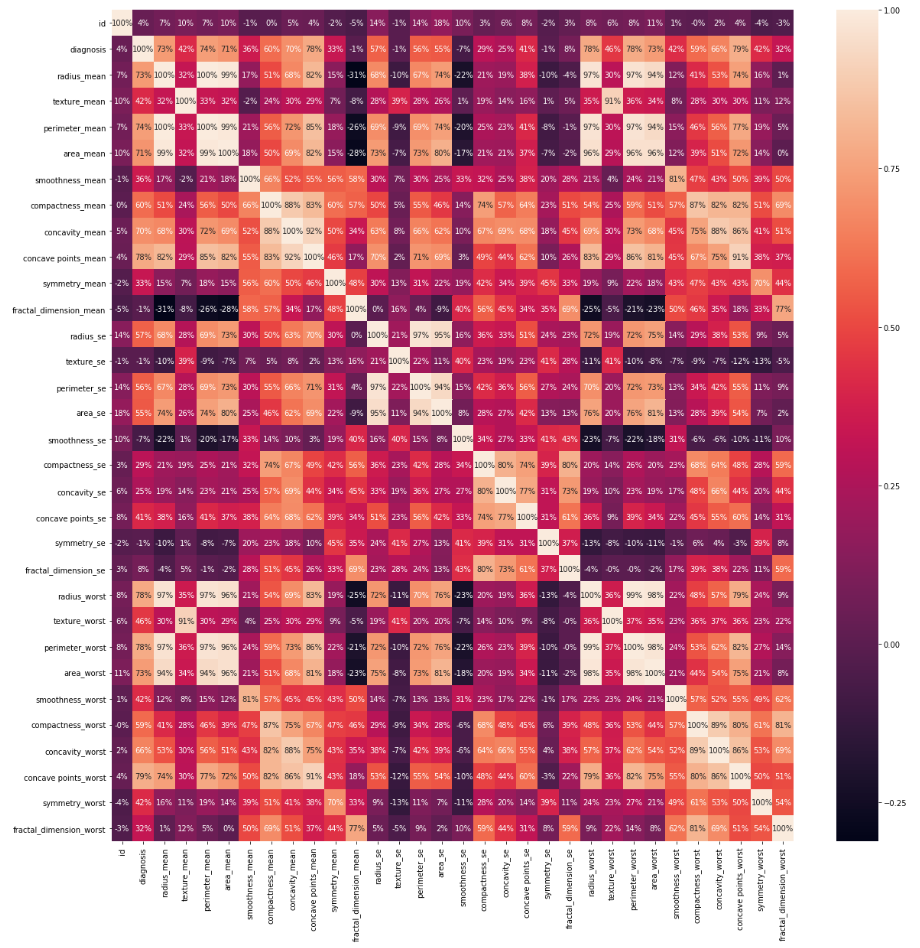
5 rows of the new data set

Get the correlation of the columns.



Column correlation

**Visualize the correlation** by creating a heat map.



Heat map of correlations

Set up the data for the model by first **splitting the data set** into a feature data set also known as the independent data set (X), and a target data set also known as the dependent data set (Y).

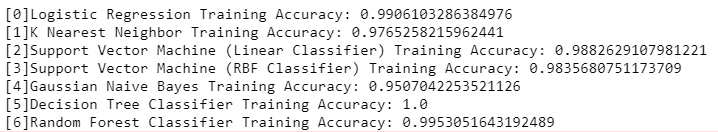
**Split the data** again, but this time into 75% training and 25% testing data sets.

**Scale the data** to bring all features to the same level of magnitude, which means the feature / independent data will be within a specific range for example 0–100 or 0–1.

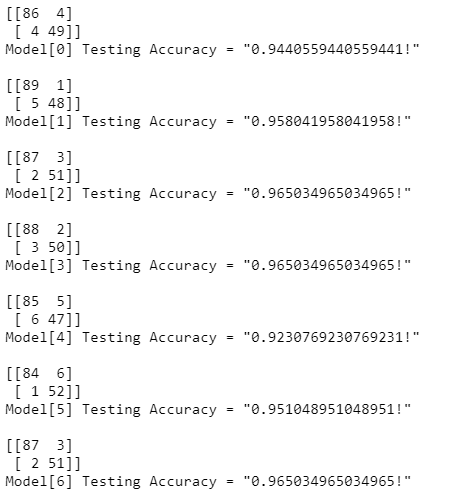
**1.3 CLASSIFICATION METHODS, METRICS AND ACCURACY**

**Create a function** to hold many different models (e.g. Logistic Regression, Decision Tree Classifier, Random Forest Classifier) to make the classification. These are the models that will detect if a patient has cancer or not. Within this function I will also print the accuracy of each model on the training data.

**Create the model** that contains all of the models, and look at the accuracy score on the training data for each model to classify if a patient has cancer or not.

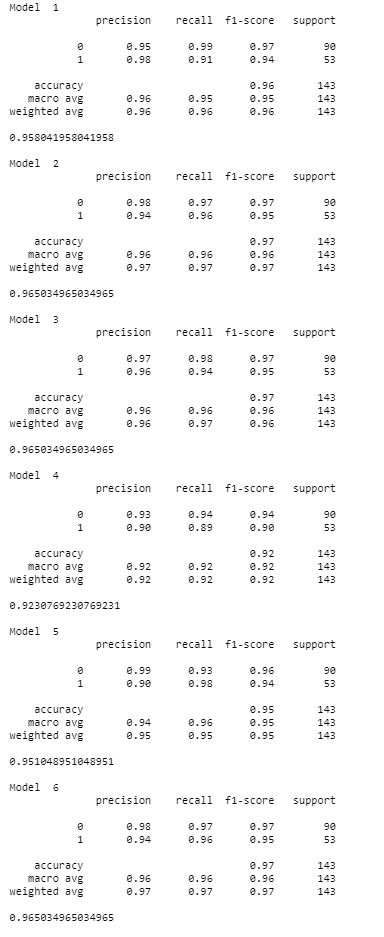


The accuracy of each model on the training data



The models confusion matrix and accuracy on test data

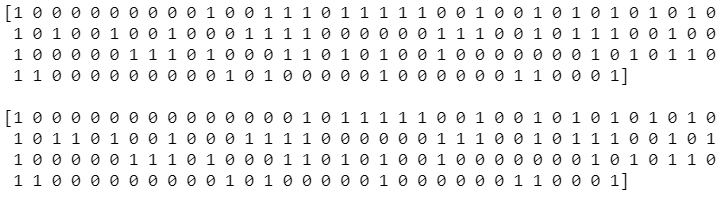
Other ways to get metrics on the model to see how well each one performed.



Sample of the models from 1–6 performance metrics on test data

**1.4 PREDICTED AND ACTUAL VALUES OF PATIENTS**

From the accuracy and metrics above, the model that performed the best on the test data was the Random Forest Classifier with an accuracy score of about **96.5%.** So I will choose that model to detect cancer cells in patients. Make the prediction/classification on the test data and show both the Random Forest Classifier model classification/prediction and the actual values of the patient that shows rather or not they have cancer.



**Top:** Decision Tree Classifier prediction, **Bottom:** The actual classification of the patients