Linear Models (gradient descend classifier) and  
Support Vector Machine

1. Take the ‘geyser.csv’ for the classification task. Split it on train and test parts.

**Option 1 (100/100)**

Train the model [SGDClassifier](https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.SGDClassifier.html) on it. To do it, you should convert labels to numbers, where a positive class (P) will be encoded by ‘+1’, and a negative (N) by ‘-1’. Print the resulting equation of the separating line.

*f(x) = w\_1 \* x\_ 1 + w\_2 \* x\_2 .....*

Consider ‘chips.csv’. Also split it on train and test parts. Do the same things.

Evaluate classifiers with F1 score . Compare the resluts (F1 scores)

F1 score - <https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html#sklearn.metrics.f1_score>

**Option 2 (150/100)**

Do exactly the same in option 1 *without using scikit library* for Stachastic Gradient Descent algorithm.

**2)** For both datasets (‘chips.csv’ and ‘geyser.csv’), do the train and test split and apply the [SVM model](https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html) on them. Try different kernels. For each kernel, find the best parameter **C** for it and draw how the SVM model classify whole space with it. You can find an example [here](https://colab.research.google.com/drive/1G9pHzipwykMJZbQP8NgJF5bWaSTPlOmJ) (notebook in the attachment).

Compare the SVM results (by F1 score).

Here you can use the scikit library.