# 3. Models validation

When we create a classification or regression model, we need to measure our model performance.

## 3.1 Training and testing sets

We divide the data into training and testing sets. The **training** set is used to fit (train) the model and the **testing** set for evaluating it.

| fixed acidity | volatile acid | citric acid | residual sug | chlorides | free sulfur d | total sulfur c | density | pH | sulphates | alcohol | quality |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 7.4 | 0.7 | 0 | 1.9 | 0.076 | 11 | 34 | 0.9978 | 3.51 | 0.56 | 9.4 | 5 |
| 7.8 | 0.88 | 0 | 2.6 | 0.098 | 25 | 67 | 0.9968 | 3.2 | 0.68 | 9.8 | 5 |
| 7.8 | 0.76 | 0.04 | 2.3 | 0.092 | 15 | 54 | 0.997 | 3.26 | 0.65 | 9.8 | 5 |
| 11.2 | 0.28 | 0.56 | 1.9 | 0.075 | 17 | 60 | 0.998 | 3.16 | 0.58 | 9.8 | 6 |
| 7.4 | 0.7 | 0 | 1.9 | 0.076 | 11 | 34 | 0.9978 | 3.51 | 0.56 | 9.4 | 5 |
| 7.4 | 0.66 | 0 | 1.8 | 0.075 | 13 | 40 | 0.9978 | 3.51 | 0.56 | 9.4 | 5 |
| 7.9 | 0.6 | 0.06 | 1.6 | 0.069 | 15 | 59 | 0.9964 | 3.3 | 0.46 | 9.4 | 5 |
| 7.3 | 0.65 | 0 | 1.2 | 0.065 | 15 | 21 | 0.9946 | 3.39 | 0.47 | 10 | 7 |
| 7.8 | 0.58 | 0.02 | 2 | 0.073 | 9 | 18 | 0.9968 | 3.36 | 0.57 | 9.5 | 7 |
| 7.5 | 0.5 | 0.36 | 6.1 | 0.071 | 17 | 102 | 0.9978 | 3.35 | 0.8 | 10.5 | 5 |
| 6.7 | 0.58 | 0.08 | 1.8 | 0.097 | 15 | 65 | 0.9959 | 3.28 | 0.54 | 9.2 | 5 |
| 7.5 | 0.5 | 0.36 | 6.1 | 0.071 | 17 | 102 | 0.9978 | 3.35 | 0.8 | 10.5 | 5 |
| 5.6 | 0.615 | 0 | 1.6 | 0.089 | 16 | 59 | 0.9943 | 3.58 | 0.52 | 9.9 | 5 |
| 7.8 | 0.61 | 0.29 | 1.6 | 0.114 | 9 | 29 | 0.9974 | 3.26 | 1.56 | 9.1 | 5 |
| 8.9 | 0.62 | 0.18 | 3.8 | 0.176 | 52 | 145 | 0.9986 | 3.16 | 0.88 | 9.2 | 5 |
| 8.9 | 0.62 | 0.19 | 3.9 | 0.17 | 51 | 148 | 0.9986 | 3.17 | 0.93 | 9.2 | 5 |

Input variables

Output variable

To divide the dataset, we randomly shuffle the samples and divide them into training and test sets. Traditionally, it could be 70% for training and 30% for testing.

| fixed acidity | volatile acid | citric acid | residual sug | chlorides | free sulfur d | total sulfur c | density | pH | sulphates | alcohol | quality |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 7.4 | 0.7 | 0 | 1.9 | 0.076 | 11 | 34 | 0.9978 | 3.51 | 0.56 | 9.4 | 5 |
| 7.8 | 0.88 | 0 | 2.6 | 0.098 | 25 | 67 | 0.9968 | 3.2 | 0.68 | 9.8 | 5 |
| 7.8 | 0.76 | 0.04 | 2.3 | 0.092 | 15 | 54 | 0.997 | 3.26 | 0.65 | 9.8 | 5 |
| 11.2 | 0.28 | 0.56 | 1.9 | 0.075 | 17 | 60 | 0.998 | 3.16 | 0.58 | 9.8 | 6 |
| 7.4 | 0.7 | 0 | 1.9 | 0.076 | 11 | 34 | 0.9978 | 3.51 | 0.56 | 9.4 | 5 |
| 7.4 | 0.66 | 0 | 1.8 | 0.075 | 13 | 40 | 0.9978 | 3.51 | 0.56 | 9.4 | 5 |
| 7.9 | 0.6 | 0.06 | 1.6 | 0.069 | 15 | 59 | 0.9964 | 3.3 | 0.46 | 9.4 | 5 |
| 7.3 | 0.65 | 0 | 1.2 | 0.065 | 15 | 21 | 0.9946 | 3.39 | 0.47 | 10 | 7 |
| 7.8 | 0.58 | 0.02 | 2 | 0.073 | 9 | 18 | 0.9968 | 3.36 | 0.57 | 9.5 | 7 |
| 7.5 | 0.5 | 0.36 | 6.1 | 0.071 | 17 | 102 | 0.9978 | 3.35 | 0.8 | 10.5 | 5 |
| 6.7 | 0.58 | 0.08 | 1.8 | 0.097 | 15 | 65 | 0.9959 | 3.28 | 0.54 | 9.2 | 5 |
| 7.5 | 0.5 | 0.36 | 6.1 | 0.071 | 17 | 102 | 0.9978 | 3.35 | 0.8 | 10.5 | 5 |
| 5.6 | 0.615 | 0 | 1.6 | 0.089 | 16 | 59 | 0.9943 | 3.58 | 0.52 | 9.9 | 5 |
| 7.8 | 0.61 | 0.29 | 1.6 | 0.114 | 9 | 29 | 0.9974 | 3.26 | 1.56 | 9.1 | 5 |
| 8.9 | 0.62 | 0.18 | 3.8 | 0.176 | 52 | 145 | 0.9986 | 3.16 | 0.88 | 9.2 | 5 |
| 8.9 | 0.62 | 0.19 | 3.9 | 0.17 | 51 | 148 | 0.9986 | 3.17 | 0.93 | 9.2 | 5 |

Training set

Test set

The main idea is to fit (train) the model only with the training set. And then measure the model performance using the test set samples. The more similar are the test output with the model prediction, the better the model performance.

| fixed acidity | volatile acid | citric acid | residual sug | chlorides | free sulfur d | total sulfur c | density | pH | sulphates | alcohol | quality | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 7.4 | 0.7 | 0 | 1.9 | 0.076 | 11 | 34 | 0.9978 | 3.51 | 0.56 | 9.4 | 5 | |
| 7.8 | 0.88 | 0 | 2.6 | 0.098 | 25 | 67 | 0.9968 | 3.2 | 0.68 | 9.8 | 5 | |
| 7.8 | 0.76 | 0.04 | 2.3 | 0.092 | 15 | 54 | 0.997 | 3.26 | 0.65 | 9.8 | 5 | |
| 11.2 | 0.28 | 0.56 | 1.9 | 0.075 | 17 | 60 | 0.998 | 3.16 | 0.58 | 9.8 | 6 | |
| 7.4 | 0.7 | 0 | 1.9 | 0.076 | 11 | 34 | 0.9978 | 3.51 | 0.56 | 9.4 | 5 | |
| 7.4 | 0.66 | 0 | 1.8 | 0.075 | 13 | 40 | 0.9978 | 3.51 | 0.56 | 9.4 | 5 | |
| 7.9 | 0.6 | 0.06 | 1.6 | 0.069 | 15 | 59 | 0.9964 | 3.3 | 0.46 | 9.4 | 5 | |
| 7.3 | 0.65 | 0 | 1.2 | 0.065 | 15 | 21 | 0.9946 | 3.39 | 0.47 | 10 | 7 | |
| 7.8 | 0.58 | 0.02 | 2 | 0.073 | 9 | 18 | 0.9968 | 3.36 | 0.57 | 9.5 | 7 | |
| 7.5 | 0.5 | 0.36 | 6.1 | 0.071 | 17 | 102 | 0.9978 | 3.35 | 0.8 | 10.5 | 5 | |
| 6.7 | 0.58 | 0.08 | 1.8 | 0.097 | 15 | 65 | 0.9959 | 3.28 | 0.54 | 9.2 | 5 | |
| 7.5 | 0.5 | 0.36 | 6.1 | 0.071 | 17 | 102 | 0.9978 | 3.35 | 0.8 | 10.5 | 5 | |
| 5.6 | 0.615 | 0 | 1.6 | 0.089 | 16 | 59 | 0.9943 | 3.58 | 0.52 | 9.9 | 5 | 4 |
| 7.8 | 0.61 | 0.29 | 1.6 | 0.114 | 9 | 29 | 0.9974 | 3.26 | 1.56 | 9.1 | 5 | 5 |
| 8.9 | 0.62 | 0.18 | 3.8 | 0.176 | 52 | 145 | 0.9986 | 3.16 | 0.88 | 9.2 | 5 | 6 |
| 8.9 | 0.62 | 0.19 | 3.9 | 0.17 | 51 | 148 | 0.9986 | 3.17 | 0.93 | 9.2 | 5 | 5 |

Prediction

## 3.2 Cross-validation

In cross-validation, the dataset is divided into several folds. Then, the model is evaluated several times, the exact times as folds are, where the testing set is set as a different fold each time.

## 3.3 Performance metrics for classification models

In supervised learning, a classification problem learns to recognize different categories. In this case, the output variable is categoric.

| Test set output | Model prediction |
|---|---|
| C | B |
| B | B |
| B | B |
| A | A |
| C | C |
| A | B |
| B | B |

The metrics that can be used for measuring the model performance is:

- **Accuracy**: It calculates the percentage of correct predictions. Its values range between 0 and 1, where 1 represents that all the test samples were predicted correctly.

- **Confusion matrix**: It is a square matrix of k rows and k columns, where k is the number of categories or classes. Each row and column represent the test and predicted values, respectively. A cell represents the percentage of test samples of the row class predicted as the column class. See examples.

If we have a binary classification problem, 1(true) or 0(false), we can divide the samples into true positives, true negatives, false negatives, and false positives. For example, a clinic test for a specific illness.

- True positives (tp): The number of samples correctly predicted as belonging to the positive class.
- False positives (fp): The number of samples incorrectly predicted as belonging to the positive class.
- True negatives (tn): The number of samples correctly predicted as NOT belonging to the positive class.
- False negatives (fn): The number of samples incorrectly predicted as NOT belonging to the positive class.

With these values, we can calculate:

- **Precision** $= \frac{tp}{tp+fp}$ Of all the samples predicted as 1, how many were 1.
- **Recall** $= \frac{tp}{tp+fn}$ Of all the samples that were 1, how many were predicted as 1.
- **F1** $= 2 * \frac{precision*recall}{precision+recall}$ It is like a average between precision and recall.

The values of these metrics (precision, recall, and f1 score) range between 0 and 1, where 1 represents that all the predictions are correct.

To generalize, when we have a multi-classification problem (more than 2 classes), we can use Macro-Precision, Macro-Recall, and Macro-F1 scores. The term macro means that the metrics are calculated for each class, and then the result is the average of all classes' scores.

In the case of imbalanced datasets, the use of accuracy could be misleading. Instead, we recommend using Macro-F1.

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Ytest | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 |
| Predictions | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

In the case of a dataset with 90% of samples with value 1. We can fit a model that always predicts 1.

Model results:

Accuracy: 0.9 F1: 0.947 Macro F1: 0.473

## 3.3 Performance metrics for regression models

In supervised learning, a regression problem consists into predict an output variable based on several input variables. In this case, the output variable is numeric.

| Test set output | Model prediction |
|---|---|
| 9.5 | 9.3 |
| 2.4 | 3.1 |
| 1.6 | 0.9 |
| 5.2 | 4.8 |
| 6.6 | 7.2 |
| 7.3 | 7.2 |
| 3.9 | 3.5 |

Two simple metrics that we can use for measuring the model performance are

- **Mean Absolute Error (MAE):** It is the average of the absolute differences between the real output value $y_i$ and the predicted one $\hat{y}_i$. It is easy to interpret because represents the average of the errors.

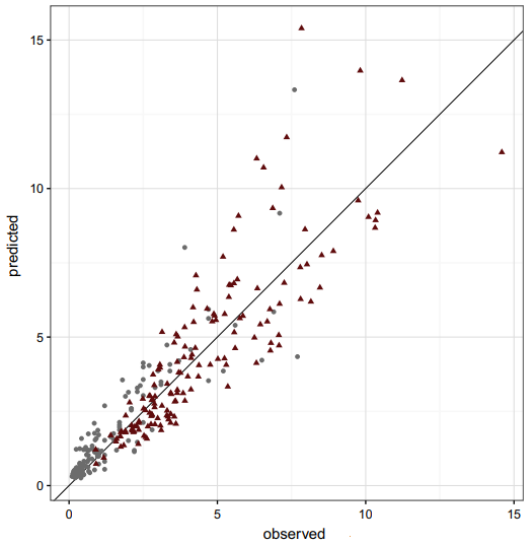
$$\mathrm{MAE}(y, \hat{y}) = \frac{\sum_i | y_i - \hat{y}_i |}{n}$$

- **Mean Square Error (MSE)**: It is the average of the square differences between the real output value $y_i$ and the predicted one $\hat{y}_i$. If we use an statistical comparison is better to use MSE.

$$\mathrm{MSE}(y, \hat{y}) = \frac{\sum_i (y_i - \hat{y}_i)^2}{n}$$

In the cases of MAE and MSE, the values depend on the feature range.

In addition, we can generate a scatter plot of the real values versus the predicted ones. In general, we expected that the data follow a line with an angle of 45 grades. The more the data follow that line, the better the model.



Finally, we have another metric:

- **$R^2$ score, coefficient of determination**: It measures how well the model works based on the variance of the independent variable.

$$R^2(y,\hat{y}) = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}$$

Interpretation:

$R^2 = 1$: It is the best value and indicates that the model predicts the independent variable perfectly.

$R^2 = 0$: It means that predictions are as good as random guesses based on the mean and the variance of the independent variable.

$R^2 < 0$: It means that predictions are worse than random.


*Figure 1 Taken from https://towardsdatascience.com/*

Comparison of correlation coefficient and coefficient of determination:

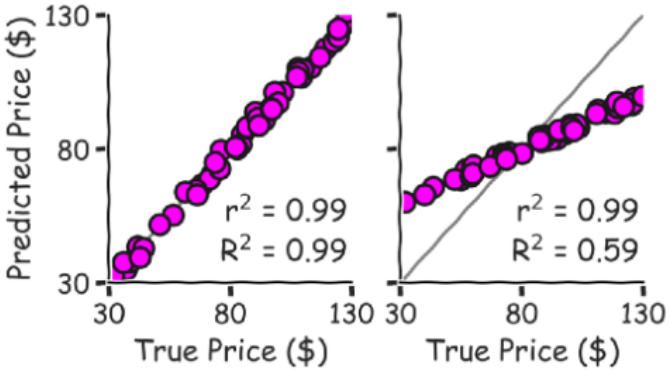



*Figure 2 Taken from https://towardsdatascience.com/*