test set. This can be seen as training on the test set and thus it has "tainted" the purity of the test set. To avoid this, an additional split is made such that we get a training set, a validation set, and a test set, where you often see a $(80/10/10)\%$ ratio. The two models can then be compared on the validation set and the performance of the chosen model can be estimated from the test set.

This way of splitting up the data has some clear benefits and is thus also often used. There is a drawback, however, and that is that we are not fully utilizing a lot of the data in this way. Basically, 20 % of the data are only used to provide a single number of performance and does not necessarily allow an uncertainty or confidence interval of this measurement to be calculated. Thus, other methods of estimating model performance are developed.

One of the most used and well-known ones is the k-fold cross validation (CV). In k-fold cross validation the entire dataset is split up into k chunks which are randomly drawn subsamples (without replacement). In the first iteration, the model is trained on the first $k-1$ subsamples and evaluated on the last k subsample. In the second iteration the evaluation subsample is a new one. This process is continued k times until all samples in the dataset have been trained and evaluated on [56]. For an illustration of this, see Figure 2.6.

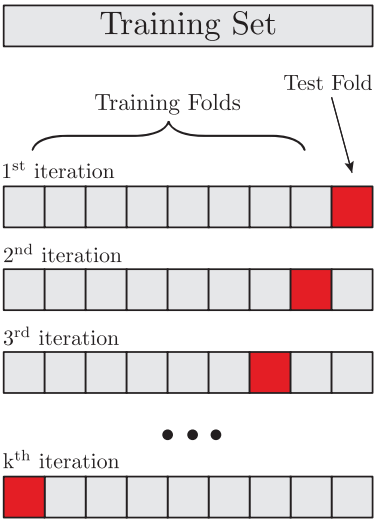

Figure 2.6: k-fold cross validation.

The process yields k estimates of the performance of the model which can then be averaged to form a single performance number and the variability of the performance can even be gauged[22]. The disadvantage of k-fold CV is that the performance estimate is now slightly biased, however, this effect is generally very small. The biggest disadvantage is the computational burden related to doing k-fold CV where $k \gg 1$. A compromise often used in applied machine learning is $k = 5$ which is also what is used in this project.

[22] Special care has to be taken here since the k different performance values are not independent.

Special care has to be taken when dealing with time series data. Here the problem of *data leakage* is often introduced inadvertently. Data leakage is when the model is exposed to information from the test set that it was not supposed to be exposed to. In the case of time series data, if the data is split by the usual k-fold CV, then each subsample contains events from all times and the model does not learn how to predict future events. To circumvent this problem, a special type of k-fold CV for time series data has to be used. Here all samples up to a specific time, eg. all houses sold before 2018, is used for training and then the model is evaluated on the performance of samples after the event, e.g. houses sold in 2018. For an illustration of this, see Figure 2.7.

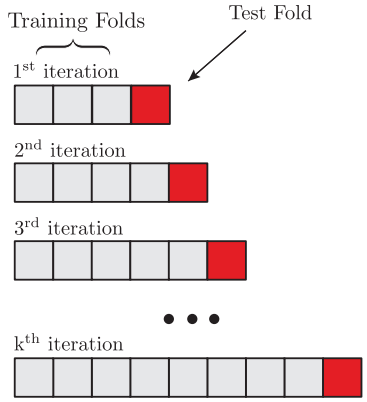

Figure 2.7: k-fold cross validation for time series data.

2.4.3 *Early Stopping*

Most modern machine learning models are trained iteratively. This is the case for both (boosted) decision trees and neural networks, both of which are used in this project. In this context, iteratively means that the model starts off with an initial guess of the parameters of the model and "learns" a new and better set of values by looking at the