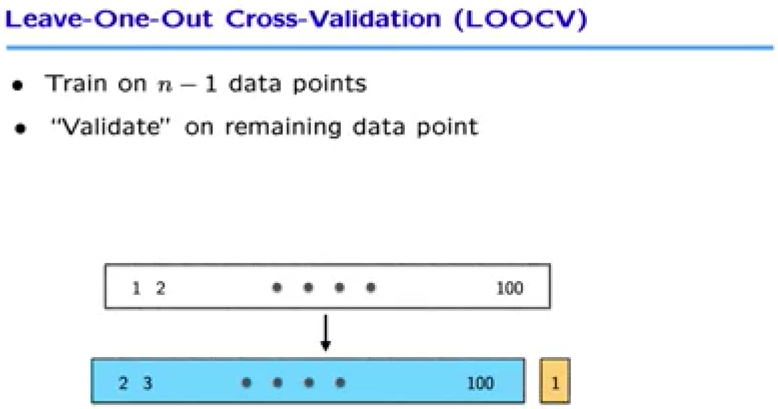
Cross-Validation Techniques

One way of removing the drawbacks of using the validation process that we described earlier, is a variation of the same idea which comes with the name of **Leave One Out Cross-Validation.**

Here is what the idea is:



Leave-One-Out Cross-Validation (LOOCV)

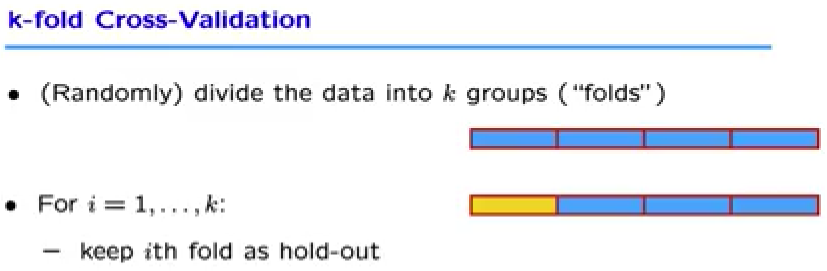
- Train on $n-1$ data points
- "Validate" on remaining data point

Instead of dividing our data set into two large pieces, what we do is set aside just one data point and run our regression on the remaining ones. So, we use, let's say in this example, 99 data points to build a regression model, and then we see what kind of prediction it produces on the remaining data point, and see if it is good or bad. And then, we repeat this process each time setting aside a different point. So, again we train on these data points, create a prediction for the one that was left out and see what kind of error we get on the one that was left out, and then take the average of the prediction errors that we get on all of those repetitions.

So, with this method, we have some advantages. We do not have this random choice of validation sets, wherein we choose many possible validation sets and we use all the data in our training or essentially all of the data. But there are some drawbacks, of course. We have to run the regression n times.

Each time that we leave one data point out, we run a regression and then we keep doing that over and over. Well, this is not always so bad. If we do just ordinary linear regression, we do not need to repeat all the computational work n times, there are certain shortcuts, there are certain formulas that allow us to do less computation but still get to the results of those n regressions and so, use this particular method. On the other hand, there is another drawback that the prediction errors that we get, they are highly dependent because each time we are training on essentially all the data or almost all of the data. So, the errors that we get tend to be very correlated in each repetition of this leave one out method, and so we do not really have n independent assessments. The amount of information that we get in some sense is less than n times what we would get from a single validation set. So, we may want to do something a little more practical.

The most popular way of mitigating the drawbacks of the validation method in which we use a single validation set is actually the so-called **K-fold Cross-Validation**. We divide the data into K groups. These are commonly called folds and then we proceed as follows, for each one of the folds, we do the following.



k-fold Cross-Validation

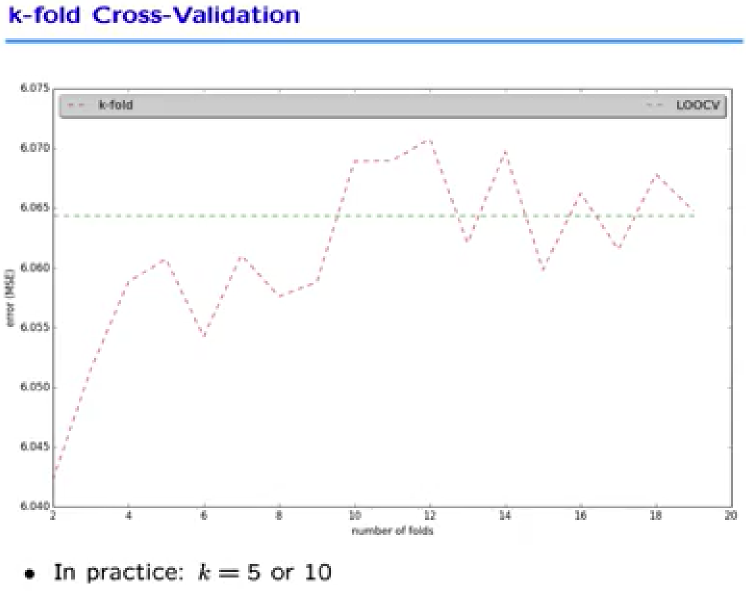
- (Randomly) divide the data into k groups ("folds")
- For $i = 1, \ldots, k$:
  - keep ith fold as hold-out

Let us say, for the first fold. We leave it out as a holdout, we train a regression model using the remaining data. Once we have that regression model, we have a predictor. We use that predictor to make predictions for the points in the holdout set. We compare it with the actual values and we see what kind of prediction errors we get, and then we repeat this process on all other holdout sets. Again, we are holding out this particular part of our data set. Training the remaining data, come up with predictions on this part and see what kind of mean squared error we get and we do that for all the K folds that we have. So, we repeat this process of building a regression model K times. For each time that we do it, we find the mean squared error that we have on the hold out data and we take the average of this, and that gives us a summary score for a particular regression method.

So, we have fixed a particular regression method, and for that approach, we run it K times and we get those mean squared errors on the remaining data points and we get a summary score for that particular regression method. If we want to compare different regression methods, for example, different choices of variables to use. We just repeat this process for every candidate method and at the end, we are going to choose the method that has performed best.

An important detail that comes in here is what should K be?

In practice, K is chosen to be a moderate number of the order, let us say, between 5-10.



k-fold Cross-Validation

- In practice: $k = 5$ or 10

Let us see how this works in practice. When K is very large, we believe that the results are most trustworthy. In the case of K the results of the validation i.e. the evaluation of the mean squared error is most trustworthy. So, we take that to be the gold standard of a validation method. But when K is very large, this would be very expensive to do. We want to have a moderate value of K and here are the results: the reported or estimated mean squared errors using different choices of K, different choices of the number of folds. We see that when we have about 10 folds or so, what K fold cross-validation reports is essentially the same as what we have gotten with a huge value of K. So, that's some evidence that a moderate value of K suffices. We do not need to chop our data set into too many pieces and we do not need to repeat the regressions too many times.

So, this is the story about validating predictors and choosing between different predictors. All of our discussion about validation so far has been in terms of mean squared error and we assess mean squared errors by doing some kind of validation and in the end, we pick the best predictor, i.e. the best method for doing predictions by looking at the performance on various types of validation sets.