Cross-Validation

Mohammad Emtiyaz Khan
EPFL

Oct 6, 2015

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Motivation

In ridge regression, the parameter $\lambda > 0$ can be tuned to reduce overfitting by reducing model complexity.

$$\min_{\beta} \quad \frac{1}{2N}\sum_{n=1}^{N}[y_n - \widetilde{\boldsymbol{\phi}}(\mathbf{x}_n)^T\boldsymbol{\beta}]^2 \quad + \quad \frac{\lambda}{2N}\sum_{j=1}^{M}\beta_j^2$$

But how do we choose λ?

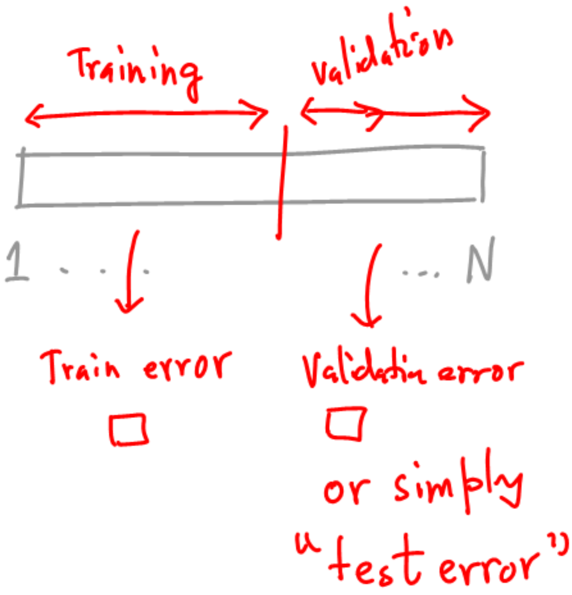
The generalization error

The generalization error of a learning method is the expected prediction error for *unseen* data, i.e. mistakes made on the data that we are going to see in the future. This quantifies how well the method *generalizes*.

Simulating the future

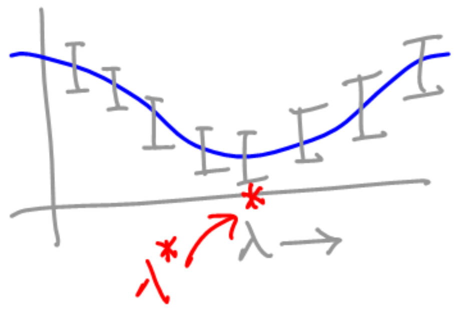
Ideally, we should choose λ to minimize the mistakes that will be made in the future. Obviously, we do not have the future data, but we can always *simulate the future* using the data in hand.

Splitting the data



For this purpose, we split the data into train and validation sets, e.g. 80% as training data and 20% as validation data. We pretend that the validation set is the future data. We fit our model on the training set and compute a prediction-error on the validation set. This gives us an *estimate* of the generalization error (one instant of the future).

To choose λ, use average RMSE.



We plot estimates of the generalization error for many values of λ (grid search). We can then repeat this process for many random splits to obtain confidence in our estimate. ← Q

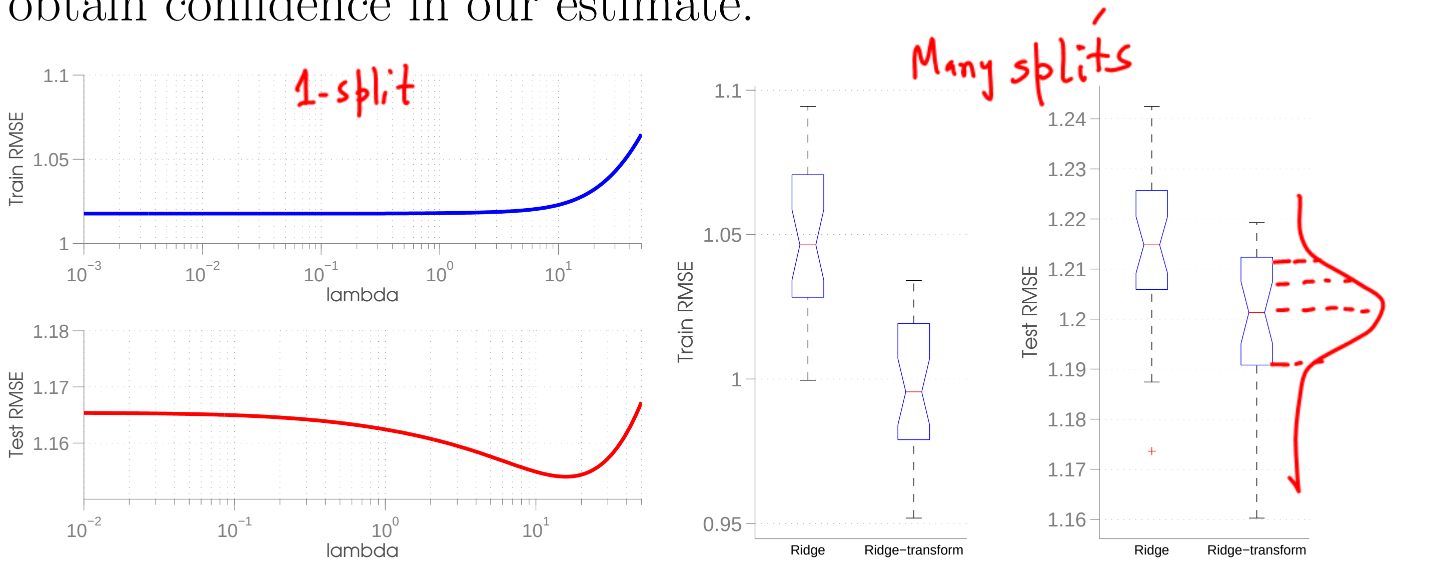
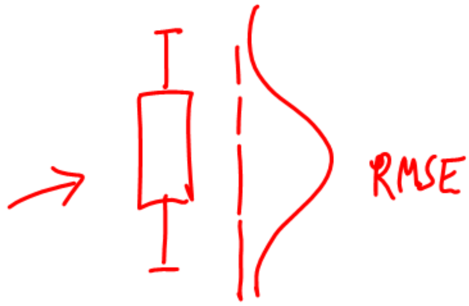


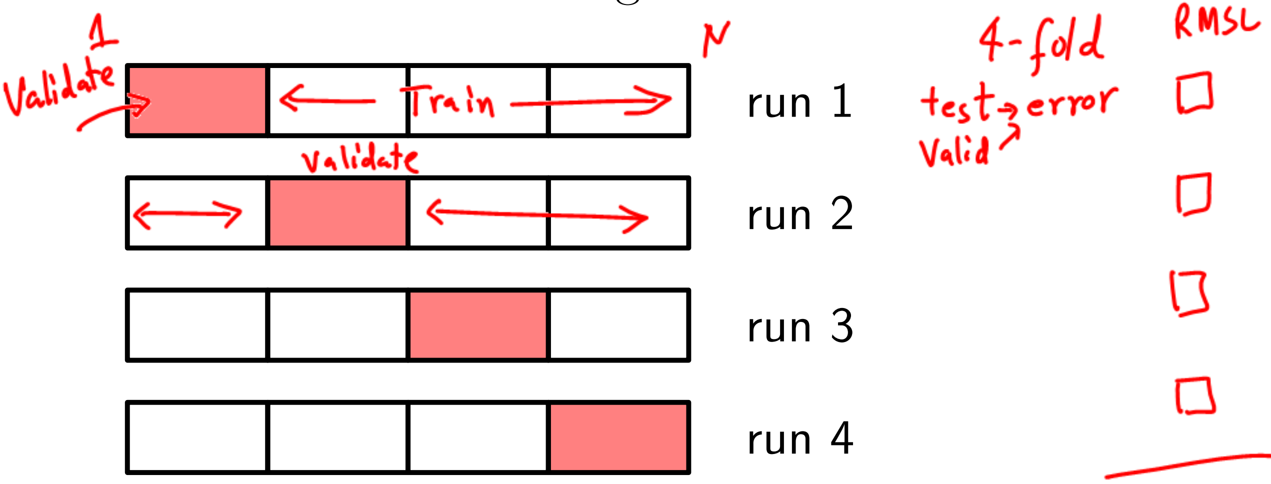
Figure 1: The left figure shows ridge regression results for a 50-50 split. The right one shows a comparison with and without feature transformations. The improvement is very little and might be insignificant.

Cross-validation



Random splits are not the most efficient way to compute the error.

K-fold cross-validation allows us to do this efficiently. We randomly partition the data into K groups. We train on $K-1$ groups and test on the remaining group. We repeat this until we have tested on all K sets. We then average the results.



average RMSE

Cross-validation returns an unbiased estimate of the *generalization error* and its variance.

Question!

Additional Notes

Pseudo code for CV

```
% given K splits (yk, Xk)
for i = 1:length(vals)
    lambda = vals(i);
    for k = 1:K
        % Compute beta for subgroups other than k
        beta = ...
        % train & test error on k'th subgroup
        errTrSub(k) = computeCost(yk, Xk, beta);
        errTeSub(k) = computeCost(yk, Xk, beta);
    end
    % compute average of train and test errors
    errTr(i) = mean(errTrSub(k));
    errTe(i) = mean(errTeSub(k));
end
[errStar, lambdaStar] = min(errTe);
```

To do

- Implement CV and gain experience to set λ and K.
- Details on unbiasedness of cross-validation is in Section 7.10 in the book by Hastie, Tibshirani, and Friedman (HTF).
- Read about bootstrap in Section 7.11 in HTF book. This method is related to random splitting and is a very popular method.