

# Re: data scaling and validation for learning classifier

---

*Source:* <http://sci.tech-archive.net/Archive/sci.stat.math/2009-04/msg00229.html>

---

- *From:* Greg Heath <[heath@xxxxxxxxxxxxxxxxxxx](mailto:heath@xxxxxxxxxxxxxxxxxxx)>
  - *Date:* Thu, 23 Apr 2009 12:53:41 -0700 (PDT)
- 

On Apr 23, 11:10 am, Tim <[timlee...@xxxxxxxxxx](mailto:timlee...@xxxxxxxxxx)> wrote:

Thanks Greg!

WARNING: My definitions of validation and testing are the standard in neural network design (See the [comp.ai.neural-nets FAQ](#)). However, Warren Sarle has warned that some statisticians use the opposite terminology.

Sorry for my poor description. What I want to do is just training and validation on the dataset that is available. The testing set will be provided in the future.

That is not reasonable.

You need a test with currently available nondesign data to determine if the current design should be kept to be tested or used with future data.

Maybe the following will be clearer

Data = Design + Test

Design = Training + Validation

Test = Current Nondesign + Future Nondesign

Sometimes the future nondesign data is really operational data. Then, it is crucial to use the current nondesign data for deciding if the current design is credible or not.

## Re: data scaling and validation for learning classifier

Before reading your post, I am only aware of dividing the dataset available, which I call "original training set", into training set and validation set.

No.

The original data is partitioned into a design set and a nondesign test set. However, if you have to tweak and tune to determine model topology (e.g., projection pursuit or neural net) and/or optimization algorithm hyperparameters, you need to further partition your design set into training and validation sets. Otherwise the training set is the entire design set.

If the current test set performances don't yield acceptable results, then go back to square 1 and try again with repartitioned data, maybe with  $f$  changed. However, it very well may be that the specified performance that you require is not achievable. In that case you either live with your best design or fold your tent.

It is really good to know how validation and testing can both be done by leaving a fold for testing.

It is essential to realize that every design achieved with training and validation should be tested with current nondesign test data before acceptance for use with future test data.

validation and testing can both be done by leaving a fold for testing.

The terminology "fold" implies one complete stage of design and testing. Again, if tweaking and tuning are necessary, validation must be included during design.

Please let me try to describe what I meant to ask. If I do some > feature transformation in preprocessing like scaling, normalization or > standardization, which way is more proper regarding the order between

## Re: data scaling and validation for learning classifier

it and cross validation:

1. firstly scale the features in the WHOLE training set,

No.

then do the cross validation.

Cross validation implies training, validation(if needed) and testing.

When testing, use the same scales on the the training set.

... as the ...

Yes.

2. firstly divide the training set into folds for training and validation.

Improper terminology.

For f-fold XVAL, the data is partitioned into f subsets.  
For each fold, one of the f subsets is used as a test set and the remainder is used for design (training and validation).  
Training data estimates are used to scale all of the data.

In each alternation of cross validation, do the scaling on the SHRUNK training set ,

Nothing is shrunk.

You seem to think of the original data as training data.

Don't.

Think of the original data as design and test data.  
If tweaking and/or tuning is required the design data must be further split into training and validation data. Otherwise the training data is the design data.

Re: data scaling and validation for learning classifier

.. and these scales will apply to the corresponding

validation

set in the same alternation. After cross validation complete, do another scaling on the original training set (train + validation sets), and the scales computed will be applied to the testing set.

I see your point. However, this is only consistent if the design set(training + validation) is used as a new training set for a new design.

So I think what you described in your 2.a is basically what I described in my 2? The argument for my approach 2 is: If you have a training set T and a validation set V, then the samples in V cannot be used for ANY aspect of the learning. So you cannot determine data transformation based on T and V together and then test on V. On the other hand the argument for my approach 1 is: We can use common scaling during cross-validation, for "new\_training + validation" sets, where new\_training and validation sets really are parts of training set.

If you have a design set Des and a Test set Tst, then the samples in Tst cannot be used for ANY aspect of design. Therefore, if tweaking and tuning are necessary, Des should be partitioned into Trn and Val. Otherwise Trn is just Des.

Hope this helps.

Greg

.