ML Toolbox

Additional Tools with ML

Introduction

Improving prediction performance

- **Stacking** methods
  - Find optimal combination of lower-level models
  - Utilize different types of methods to build ensemble
- Classification with class imbalance
  - Threshold optimization
  - Cost-sensitive training/ case weights
  - **Over- and under-sampling**

Stacking

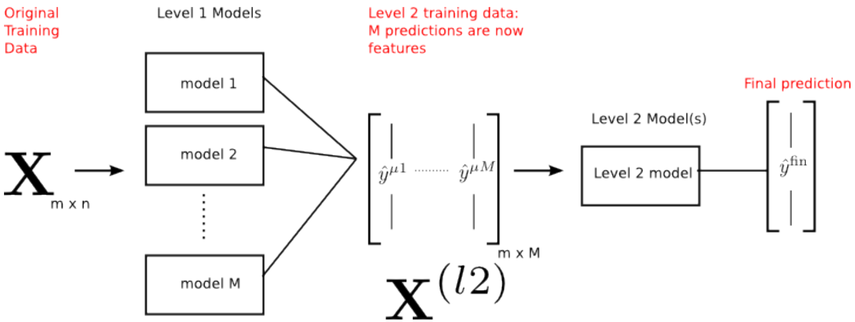
Combining prediction models

- Do we have to select one ML method to build the final model?
  - Performances might differ between regions of the data
- General idea: Combine multiple models' predictions with a **meta-model**
- Utilize diverse lower-level models that "span the space"
- Draw on advantages of different model types

$\rightarrow$ Wolpert 1992, Breiman 1996

Stacking

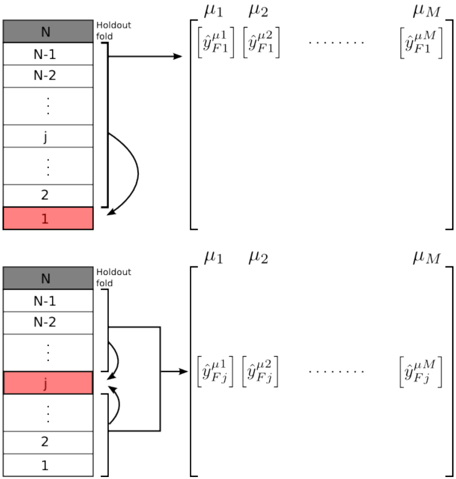
Figure: Stacking process[1]



[1] https://burakhimmetoglu.com/2016/12/01/stacking-models-for-improved-predictions/

Second level data

Figure: Create second level training data[2]



2 https://burakhimmetoglu.com/2016/12/01/stacking-models-for-improved-predictions/

Second level model

Input: Cross-validated predictions $\hat{f}_m^{-i}(x)$

Estimate stacking weights via linear regression...

$$\hat{w}^{st} = \underset{w}{\operatorname{argmin}} \sum_{i=1}^{N} \left[y_i - \sum_{m=1}^{M} w_m \hat{f}_m^{-i}(x_i) \right]^2$$

...or logistic regression for classification problems.

Final prediction

$$\sum_m \hat{w}_m^{st} \hat{f}_m(x)$$

$\rightarrow$ Super learner will perform asymptotically as well as the best base learner (van der Laan et al. 2007)

Stacking process

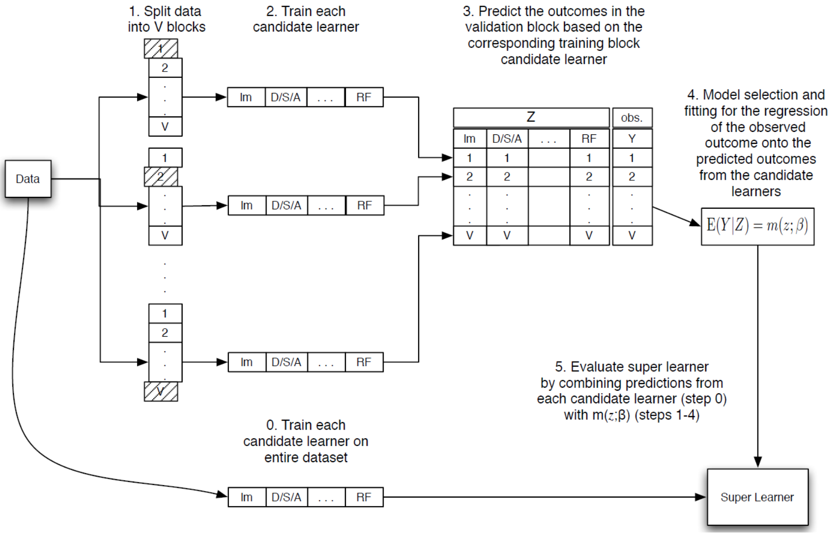
Algorithm 1: Stacking

Parameter : V folds, M base learner ψ_m, metalearning method ϕ
Initialization: Partition training data into V folds

```
for v = 1 to V do
    for m = 1 to M do
        Obtain cross-validated predictions ψ̂_m^{-v} ;
    end
end
Fit meta model φ̂ using ψ̂_m^{-v} as inputs ;
Fit M base models with full training data and save with φ̂ for prediction ;
```

Super Learner

Figure: Super Learner[3]



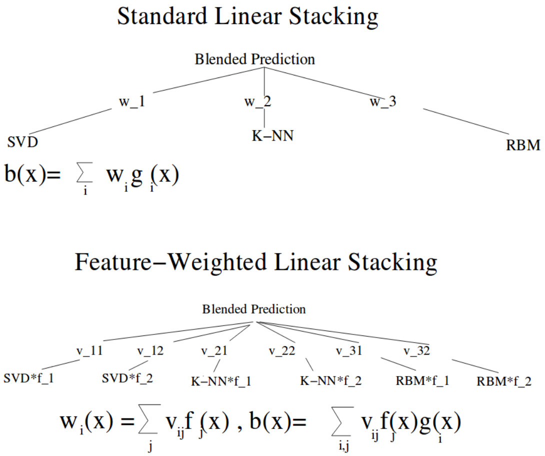
[3] van der Laan et al. (2007)

Extensions

- Blending
  - Throw predictions from various models into one hold-out set
  - Train meta model only on this hold-out set
- Subsemble (Sapp et al. 2014)
  - Introduce additional subset partitions for base learner training
- Feature-Weighted Linear Stacking (Sill et al. 2009)
  - Combines model predictions with meta-features
  - Uses prediction-feature products as meta-model inputs

Feature-Weighted Linear Stacking

Figure: Comparison of SLS and FWLS[4]



[4] Sill et al. (2009)