Model Tuning: Regularization

Motivation: Overfitting often caused by overly-complex models capturing idiosyncrasies in training set.

Regularization: Adding penalty score for complexity to cost function

$$cost_{reg} = cost + \frac{\alpha}{2} \text{ penalty}$$

Automated way to find the right balance between the important features and the overfitting of the model.

Example: Regularization - cost is the sum of the squared errors

Complicated model: 100s of thousands of features

- penalty could be the sum of the weight of each feature

α - regularization strength: larger α means larger penalty

Weight - denotes relative importance of the variable

Idea: large weights correspond to higher complexity

Regularize by penalizing large weights

Two standard types:

L1 regularization, Lasso: penaly $= \|\vec{w}\|_1 = \sum_{j=1}^{m} |w_j|$

L2 regularization, Ridge:

$$\text{penalty} = \|\vec{w}\|_2^2 = \sum_{j=1}^{m} w_j^2$$

Now we minimize total cost

L2 popular, but L1 useful as feature selection approach since most weights shrink to 0 (sparsity)

Note: important to scale features first!

scikit-learn: Models that support regularization typically provide parameters for type and strength.

Sklearn

L2 Ridge regression sklearn.linear_modd.Ridge

L1 Lasso regression sklearn.linear_model.Lasso

Both: sklearn.linear_model.ElasticNet

The strength of regularization $C = \frac{1}{alpha}$

inverse of the reg strength

Logistic regression, add in a penalty of L2 with ~~a~~ $C = 1$