Supervised learning: Lasso regularization

Filippo Biscarini (CNR, Milan, Italy)

filippo.biscarini@cnr.it

p > n problems

- when **n** $\gg$ **p** linear and logistic regression have **low variance**
- when **n** ≈ **p** the **variance** gets very high
- when **p > n** the variance tends to **infinite** → the models have no (unique) solution
- additionally, the model matrix will not be full rank (singular), hence not invertible

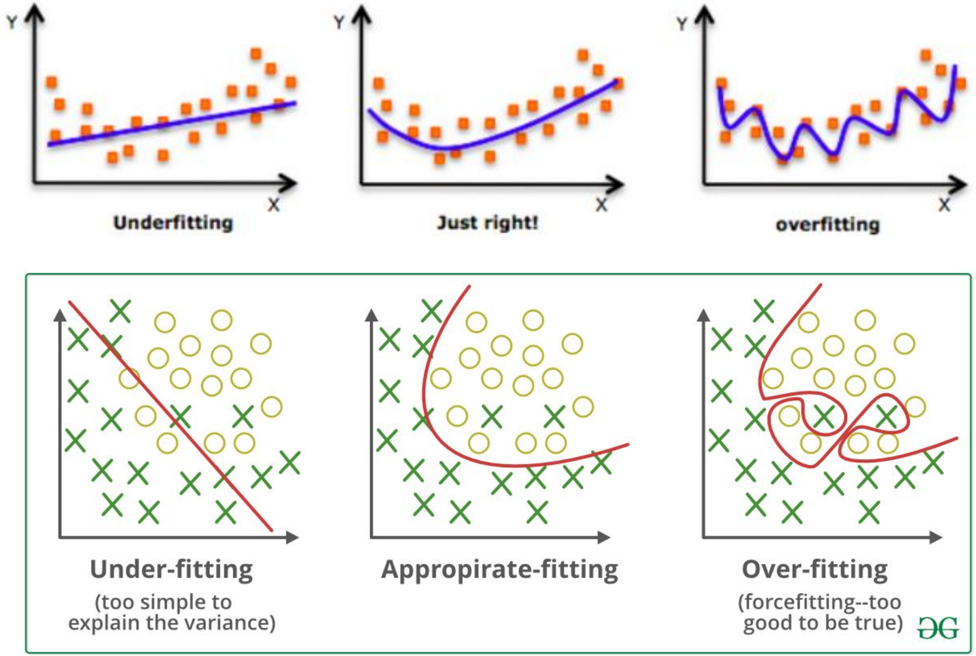
p > n problems

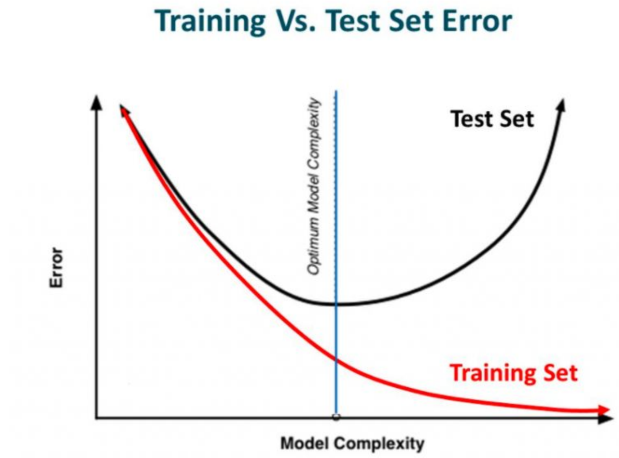
- when **n** $\gg$ **p** linear and logistic regression have **low variance**
- when **n ≈ p** the **variance** gets very high
- when **p > n** the variance tends to **infinite** → the models have no (unique) solution

↓

we need a different approach

Besides: overfitting!





https://www.analyticsvidhya.com/blog/2018/04/fundamentals-deep-learning-regularization-techniques/
https://www.geeksforgeeks.org/underfitting-and-overfitting-in-machine-learning/

Different approach → Shrinkage / Regularization

- the estimated coefficients are **shrunken towards zero**
- all *p* predictors are used in the model, but coefficients are constrained
- also known as **regularization**
- **reduces the variance** of the predictor / classifier
- different types of regularization:
  - Ridge regression
  - **Lasso**
  - Elastic net

Lasso

- Lasso: least **absolute shrinkage** and **selection** operator
- L1-norm: **absolute value** of the coefficients
- Lasso shrinks coefficients towards zero and forces some (many) to be exactly zero → **variable selection**

Tibshirani, R., 1996. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, *58*(1), pp.267-288.

Lasso

- the key is **modifying the cost function** used to solve the model
- a quantity (**penalty**) is added to the cost function

$$J(\beta) = \frac{1}{2n}\left[\sum_{i=1}^{n}(\beta_i X_i - y_i)^2 + \lambda \sum_{j=1}^{p} |\beta_j|\right]$$

Lasso

$$J(\beta) = \frac{1}{2n}\left[\sum_{i=1}^{n}(\beta_i X_i - y_i)^2 + \lambda \sum_{j=1}^{p}|\beta_j|\right]$$

- the lasso penalty includes:
    - **sum** of the **absolute values** of the coefficients
    - **tuning parameter λ**

Tuning parameter λ

- Tuning parameters are **hyperparameters** of the model/method which control some of its properties
- Tuning parameters are typically **tuned** (chosen) via **cross-validation** (model tuning)

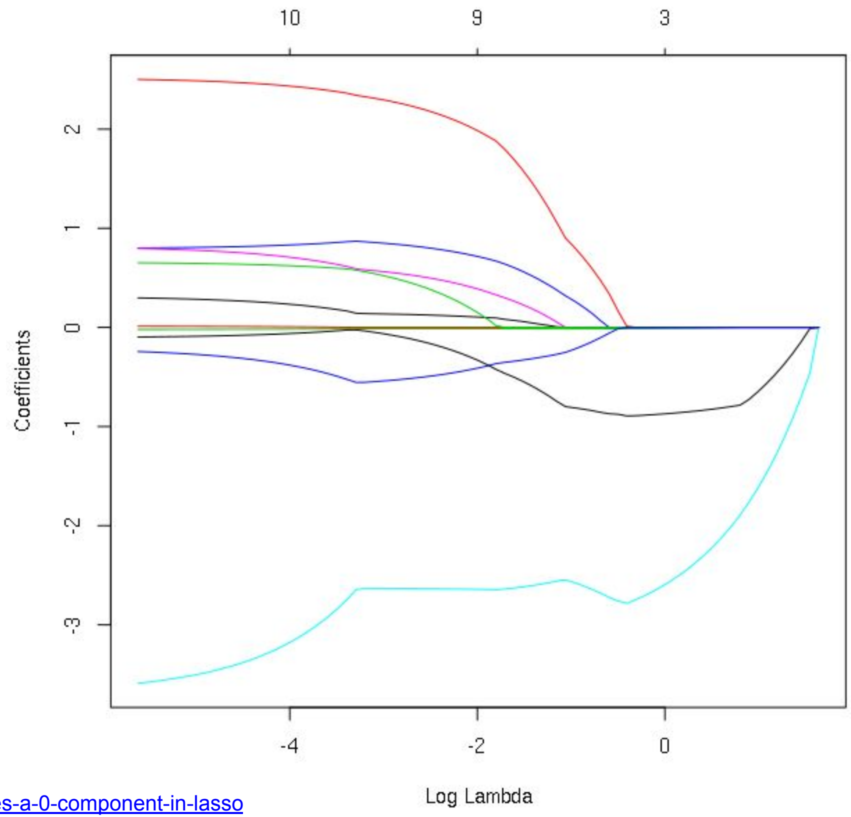
Tuning parameter λ

- Tuning parameters are **hyperparameters** of the model/method which control some of its properties
- Tuning parameters are typically **tuned** (chosen) via **cross-validation** (model tuning)
- In Lasso-penalised regression:
    - if $\lambda = 0 \rightarrow$ no regularization (ordinary regression)
    - if $\lambda \gg 0 \rightarrow$ null model (all coefficients are zero)

$$J(\beta) = \frac{1}{2n}\left[\sum_{i=1}^{n}\left(\beta_i X_i - y_i\right)^2 + \lambda \sum_{j=1}^{p}\left|\beta_j\right|\right]$$

Tuning parameter λ

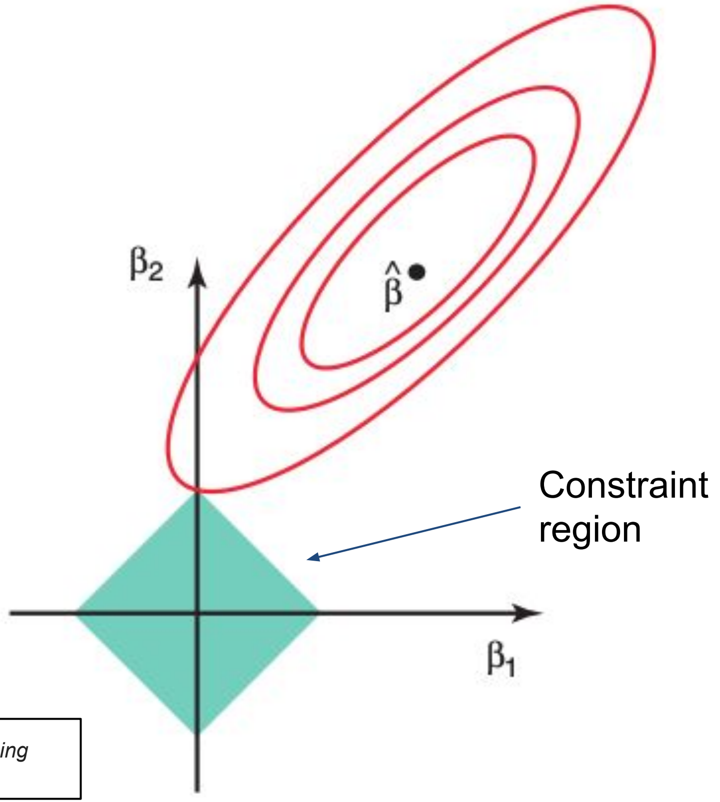
- if $\lambda = 0 \rightarrow$ no regularization (ordinary regression)
- if $\lambda \gg 0 \rightarrow$ null model (all coefficients are zero)



From: https://stats.stackexchange.com/questions/289075/what-is-the-smallest-lambda-that-gives-a-0-component-in-lasso

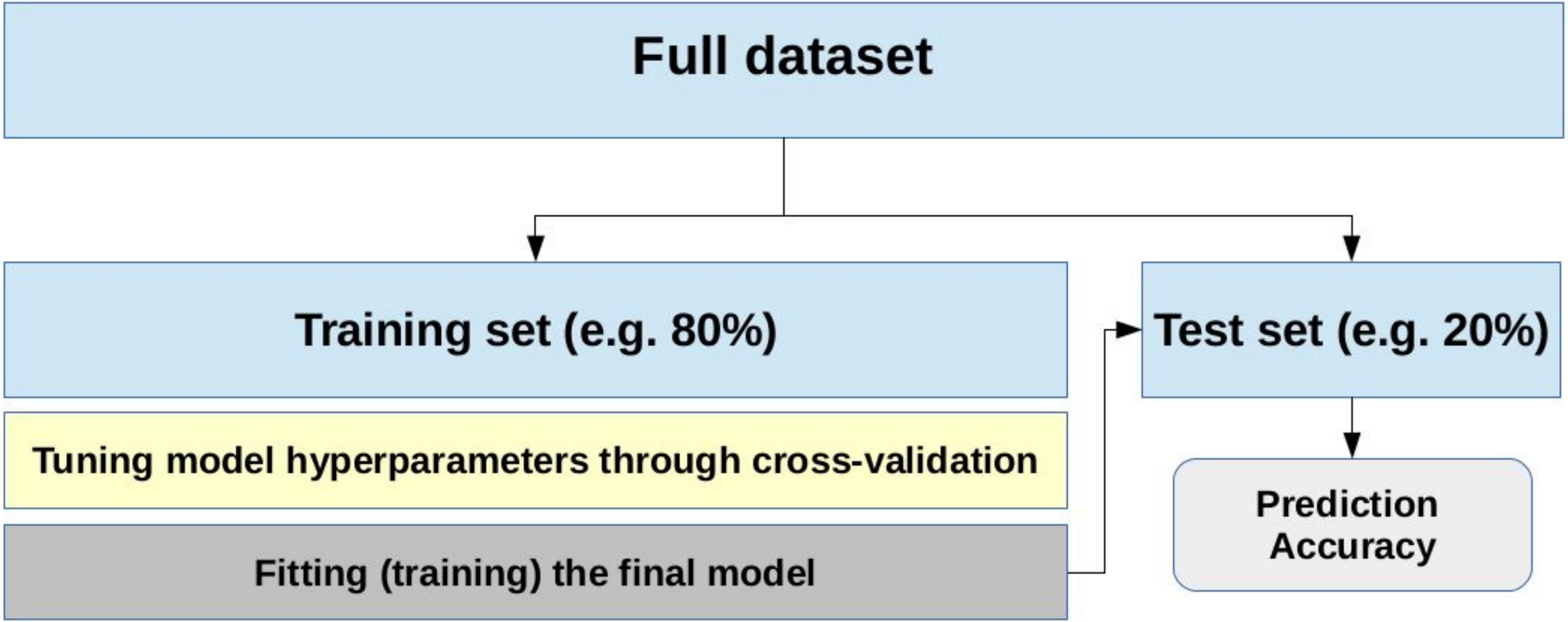
Variable selection property of the Lasso

- Lasso operates variable selection
- Lasso yields sparse models
- Improves interpretability

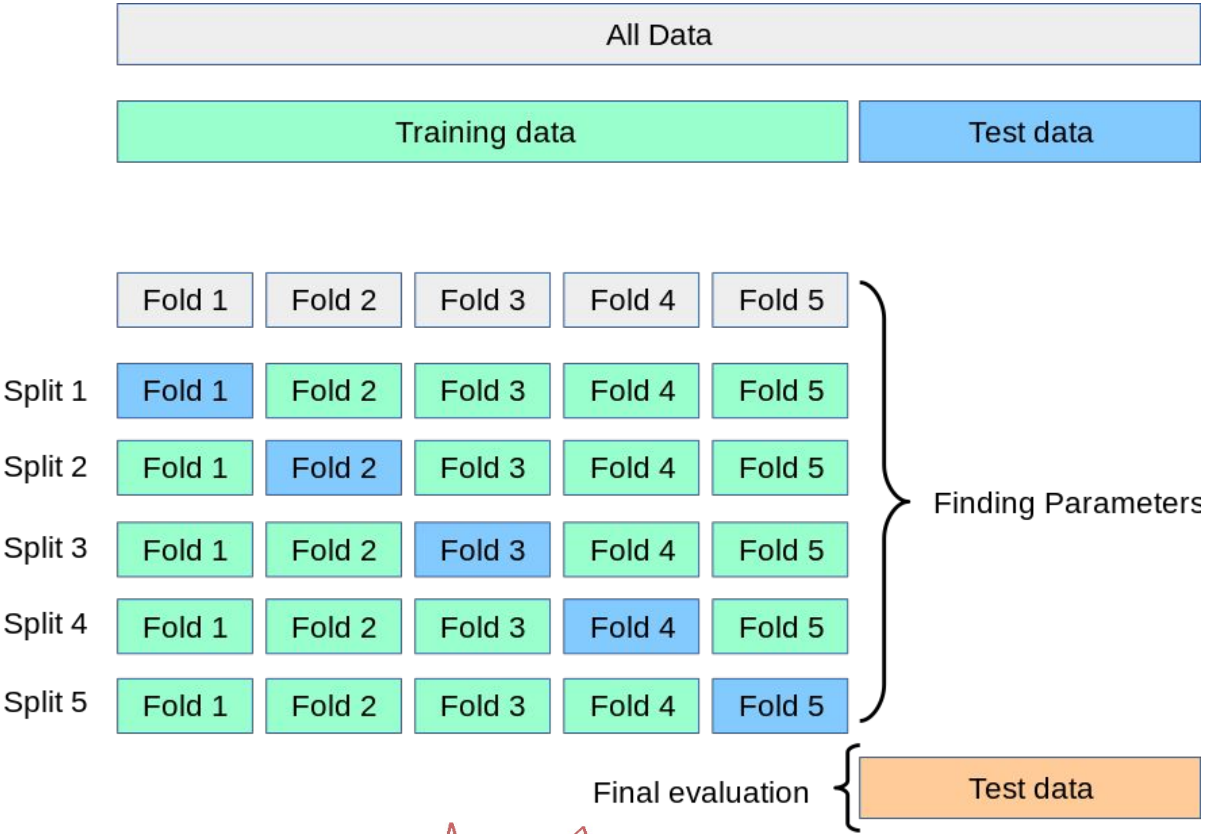


Source: James, G., Witten, D., Hastie, T. and Tibshirani, R., 2013. *An introduction to statistical learning* (Vol. 112, p. 18). New York: springer.

Model tuning



Model tuning



Model tuning

1. choose a **grid of λ values** and compute the **cross-validation error** for each value of λ
2. select the tuning parameter (λ) value for which the cross-validation error is smallest
3. **refit** the **final model** on the **training set** using the selected value of the tuning parameter
4. use this trained model on the **test set** to get a valid estimate of the predictive ability of the model

Lasso-penalised logistic regression

- demonstration 5.1
- demonstration 6.1
- exercise 6.1

→ 5.lasso.Rmd

→ 6.lasso_with_tidymodels.Rmd