TECHNISCHE UNIVERSITÄT
KAISERSLAUTERN

8.2 LOOCV

Machine Learning 1: Foundations

Marius Kloft *(TUK)*

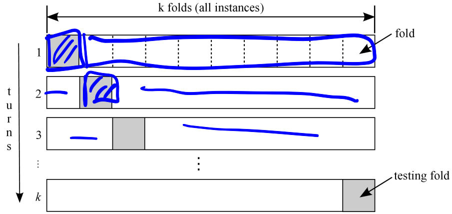
How to Select the Regularization Parameter C?

How to Select the Regularization Parameter C?

Use k-fold cross validation (CV), introduced in lecture 1:

How to Select the Regularization Parameter C?

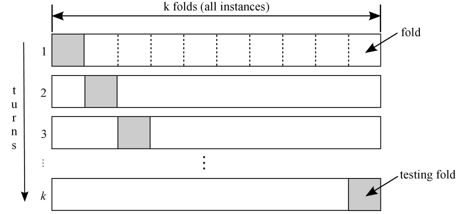
Use k-fold cross validation (CV), introduced in lecture 1:



1: split data into $k \overset{\text{e.g.}}{=} 10$ equally-sized chunks (called “folds”)
2: **for** $i = 1, \ldots, k$ and $C \overset{\text{e.g.}}{\in} \{0.01, 0.1, 1, 10, 100\}$ **do**
3: use ith fold as **test set** and union of all others as **training set**
4: train learner on training set (using C) and test on test set
5: **end for**
6: output learner with lowest average error

How to Select the Regularization Parameter C?

Use k-fold cross validation (CV), introduced in lecture 1:



split data into $k \overset{\text{e.g.}}{=} 10$ equally-sized chunks (called "folds")
for $i = 1, \ldots, k$ and $C \overset{\text{e.g.}}{\in} \{0.01, 0.1, 1, 10, 100\}$ **do**
    use ith fold as **test set** and union of all others as **training set**
    train learner on training set (using C) and test on test set
end for
output learner with lowest average error

Similarly, can select constants in other learning methods, e.g.:

- RBF-kernel width in SVM, learning rate in ANNs, etc.

But What Does **Error** Mean in Regression?

In binary classification, we had $y \in \{-1, +1\}$

- so we could just count the fraction of correctly classified test instances (the **accuracy**)

But What Does **Error** Mean in Regression?

In binary classification, we had $y \in \{-1, +1\}$

- so we could just count the fraction of correctly classified test instances (the **accuracy**)

In regression, y can attain any value: $y \in \mathbb{R}$

- whether the prediction is right or wrong is not the point here
- the point is by how much the prediction is wrong

But What Does **Error** Mean in Regression?

In binary classification, we had $y \in \{-1, +1\}$

- so we could just count the fraction of correctly classified test instances (the **accuracy**)

In regression, y can attain any value: $y \in \mathbb{R}$

- whether the prediction is right or wrong is not the point here
- the point is by how much the prediction is wrong

The common error measure in regression is:

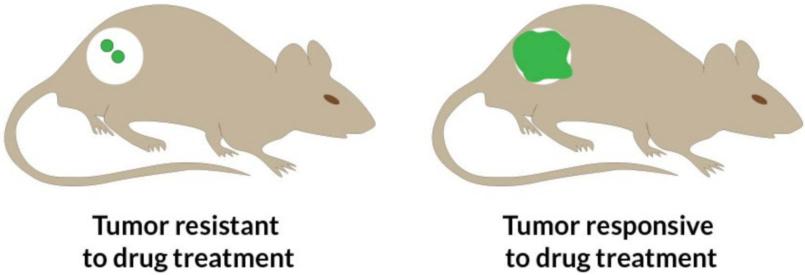
Definition

Let $\{(\mathbf{x}_1, y_1), \ldots (\mathbf{x}_n, y_n)\}$ be a test set, and let f be a learned regression function. The **root mean squared error (RMSE)** of f is:

$$\text{RSME}(f) := \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\underline{f(\mathbf{x}_i)} - \underline{y_i})^2}$$

Sometimes We Have Very Little Data

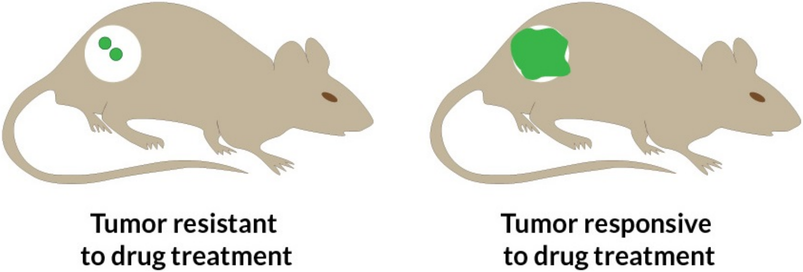
Sometimes We Have Very Little Data



Predicting the effect of an anti-cancer drug on tumors in mice

- typically $n << 100$

Sometimes We Have Very Little Data



Predicting the effect of an anti-cancer drug on tumors in mice

- typically $n << 100$

How can we use **as much data as possible**
in cross-validation?

Leave ONE Point Out For Testing and Use ALL Others For Training:

Definition

Leave-one-out cross-validation (LOOCV) is k-fold CV

- with $k := n$

Leave ONE Point Out For Testing and Use ALL Others For Training:

Definition

Leave-one-out cross-validation (LOOCV) is k-fold CV

- with $k := n$

In other words:

- we have as many folds as data points
- each fold contains only a single point

Leave ONE Point Out For Testing and Use ALL Others For Training:

Definition

Leave-one-out cross-validation (LOOCV) is k-fold CV

- with $k := n$

In other words:

- we have as many folds as data points
- each fold contains only a single point

Theoretically, LOOCV is the best procedure to select constants, such as C

Leave ONE Point Out For Testing and Use ALL Others For Training:

Definition

Leave-one-out cross-validation (LOOCV) is k-fold CV

- with $k := n$

In other words:

- we have as many folds as data points
- each fold contains only a single point

Theoretically, LOOCV is the best procedure to select constants, such as C

But what could be a problem with LOOCV?

LOOCV is Usually Super Slow

Involves a loop over all data points: $O(n)$

- In each iteration, train learner with $n-1$ data points:
  - is $O(d^3)$ for RR

Total LOOCV (for RR): $O(d^3 n)$

LOOCV is Usually Super Slow

Involves a loop over all data points: $O(n)$

- In each iteration, train learner with $n - 1$ data points:
  - is $O(d^3)$ for RR

Total LOOCV (for RR): $O(d^3 n)$

can we do ~~the~~ LOOCV faster?

- for ridge regression: yes!
- for classification: no!

even for other error measures

LOOCV is Usually Super Slow

Involves a loop over all data points: $O(n)$

- In each iteration, train learner with $n - 1$ data points:
  - is $O(d^3)$ for RR

Total LOOCV (for RR): $O(d^3 n)$

- for ridge regression: yes!
- for classification: no!

Question here!

Can we get rid of the loop over all data points?

LOOCV Trick for RR

The LOOCV error is:

$$\mathrm{RSME}_{\mathrm{loocv}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\mathbf{x}_i^\top \mathbf{w}_i - y_i\right)^2}$$

where:

- $\mathbf{w}_i$ is RR solution when ith data point is left out at training

LOOCV Trick for RR

The LOOCV error is:

$$\mathrm{RSME}_{\mathrm{loocv}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\mathbf{x}_i^\top \mathbf{w}_i - y_i\right)^2}$$

where:

- $\mathbf{w}_i$ is RR solution when ith data point is left out at training

Recall: $$\mathbf{w}_{\mathrm{RR}} = \Big(\underbrace{XX^\top}_{=\sum_{i=1}^n \mathbf{x}_i\mathbf{x}_i^\top} + \tfrac{1}{2C} I\Big)^{-1} \underbrace{Xy}_{=\sum_{i=1}^n \mathbf{x}_i y_i}$$

LOOCV Trick for RR

The LOOCV error is:

$$\text{RSME}_{\text{loocv}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\mathbf{x}_i^\top \mathbf{w}_i - y_i\right)^2}$$

where:

- $\mathbf{w}_i$ is RR solution when ith data point is left out at training

Recall: $$\mathbf{w}_{\text{RR}} = \Big(\underbrace{XX^\top}_{=\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^\top} + \frac{1}{2C}I\Big)^{-1}\underbrace{Xy}_{=\sum_{i=1}^{n}\mathbf{x}_i y_i}$$

Thus: $$\mathbf{w}_i = \left(XX^\top \boxed{-\mathbf{x}_i\mathbf{x}_i^\top} + \frac{1}{2C}I\right)^{-1}\left(Xy \boxed{-\mathbf{x}_i y_i}\right)$$

LOOCV Trick for RR

The LOOCV error is:

$$\mathrm{RSME}_{\mathrm{loocv}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\mathbf{x}_i^\top \mathbf{w}_i - y_i\right)^2}$$

where:

- $\mathbf{w}_i$ is RR solution when ith data point is left out at training

Recall: $$\mathbf{w}_{\mathrm{RR}} = \Big(\underbrace{XX^\top}_{=\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^\top} + \tfrac{1}{2C}I\Big)^{-1}\underbrace{Xy}_{=\sum_{i=1}^{n}\mathbf{x}_i y_i}$$

Thus: $$\mathbf{w}_i = \left(XX^\top - \mathbf{x}_i\mathbf{x}_i^\top + \tfrac{1}{2C}I\right)^{-1}\left(Xy - \mathbf{x}_i y_i\right)$$

Problem:

- Need to invert the matrix occuring in $\mathbf{w}_i$ for all $i = 1, \ldots, n$
- Each inversion is $O(d^3) \;\Rightarrow\;$ total: $O(d^3 n)$

LOOCV Trick for RR

The LOOCV error is:

$$\text{RSME}_{\text{loocv}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\mathbf{x}_i^\top \mathbf{w}_i - y_i\right)^2}$$

where:

- $\mathbf{w}_i$ is RR solution when ith data point is left out at training

Recall: $$\mathbf{w}_{\text{RR}} = \Big(\underbrace{XX^\top}_{=\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^\top} + \tfrac{1}{2C}I\Big)^{-1}\underbrace{Xy}_{=\sum_{i=1}^{n}\mathbf{x}_i y_i}$$

Thus: $$\mathbf{w}_i = \left(XX^\top - \mathbf{x}_i\mathbf{x}_i^\top + \tfrac{1}{2C}I\right)^{-1}\left(Xy - \mathbf{x}_i y_i\right)$$

Problem:

- Need to invert the matrix occuring in $\mathbf{w}_i$ for all $i = 1, \ldots, n$
- Each inversion is $O(d^3)$ $\Rightarrow$ total: $O(d^3 n)$

Turns out: ONE matrix inversion suffices (total of $O(d^3)$).

LOOCV Trick for RR

The LOOCV error is:

$$\mathrm{RSME}_{\mathrm{loocv}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\mathbf{x}_i^\top \mathbf{w}_i - y_i\right)^2}$$

where:

- $\mathbf{w}_i$ is RR solution when ith data point is left out at training

Recall: $$\mathbf{w}_{\mathrm{RR}} = \Big(\underbrace{XX^\top}_{=\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^\top} + \tfrac{1}{2C} I\Big)^{-1} \underbrace{Xy}_{=\sum_{i=1}^{n}\mathbf{x}_i y_i}$$

Thus: $$\mathbf{w}_i = \left(XX^\top - \mathbf{x}_i\mathbf{x}_i^\top + \tfrac{1}{2C} I\right)^{-1}(Xy - \mathbf{x}_i y_i)$$

Problem:

- Need to invert the matrix occuring in $\mathbf{w}_i$ for all $i = 1, \ldots, n$
- Each inversion is $O(d^3)$ $\Rightarrow$ total: $O(d^3 n)$

Turns out: ONE matrix inversion suffices (total of $O(d^3)$).

How does this trick work?

Skipping the Matrix Inversion—Here's the Trick:

Write: $$\mathbf{w}_i = (\underbrace{XX^\top + \frac{1}{2C}I}_{=:A} - \underbrace{\mathbf{x}_i\mathbf{x}_i^\top}_{\mathbf{u}\mathbf{u}^\top})^{-1}(Xy - \mathbf{x}_i y_i)$$

Skipping the Matrix Inversion—Here's the Trick:

Write: $$\mathbf{w}_i = \Big(\underbrace{XX^\top + \frac{1}{2C}I}_{=:A} - \underbrace{\mathbf{x}_i\mathbf{x}_i^\top}_{\mathbf{u}\mathbf{u}^\top}\Big)^{-1}(Xy - \mathbf{x}_i y_i)$$

Apply the following theorem:

Theorem (Sherman-Morrison formula)

Let $A \in \mathbb{R}^{d\times d}$ be an invertible matrix, and let $\mathbf{u} \in \mathbb{R}^d$.
If $\mathbf{u}^\top A^{-1}\mathbf{u} \neq 1$, then:

$$(A - \mathbf{u}\mathbf{u}^\top)^{-1} = A^{-1} + \frac{A^{-1}\mathbf{u}\mathbf{u}^\top A^{-1}}{1 - \mathbf{u}^\top A^{-1}\mathbf{u}}$$

Full rank

rank 1

Skipping the Matrix Inversion—Here's the Trick:

Write: $$\mathbf{w}_i = \big(\underbrace{XX^\top + \frac{1}{2C}I}_{=:A} - \underbrace{\mathbf{x}_i\mathbf{x}_i^\top}_{\mathbf{u}\mathbf{u}^\top}\big)^{-1}(Xy - \mathbf{x}_i y_i)$$

Apply the following theorem:

Theorem (Sherman-Morrison formula)

Let $A \in \mathbb{R}^{d\times d}$ be an invertible matrix, and let $\mathbf{u} \in \mathbb{R}^d$.
If $\mathbf{u}^\top A^{-1}\mathbf{u} \neq 1$, then:

$$(A - \mathbf{u}\mathbf{u}^\top)^{-1} = A^{-1} + \frac{A^{-1}\mathbf{u}\mathbf{u}^\top A^{-1}}{1 - \mathbf{u}^\top A^{-1}\mathbf{u}}$$

Thus: $$\mathrm{RSME}_{\mathrm{loocv}} = \sqrt{\textstyle\sum_{i=1}^n \left(\mathbf{x}_i^\top \mathbf{w}_i - y_i\right)^2}$$

$$= \sqrt{\sum_{i=1}^n \left(\mathbf{x}_i^\top \left(A^{-1} + \frac{A^{-1}\mathbf{x}_i\mathbf{x}_i^\top A^{-1}}{1 - \mathbf{x}_i^\top A^{-1}\mathbf{x}_i}\right)(Xy - \mathbf{x}_i y_i) - y_i\right)^2}$$

The Last Equation From the Previous Slide ...

... shows already that we can come along with a total of $O(d^3)$ to compute the LOOCV error.

The Last Equation From the Previous Slide ...

... shows already that we can come along with a total of $O(d^3)$ to compute the LOOCV error.

But we can further simplify the expression ~~(shown at the board)~~

The Last Equation From the Previous Slide ...

... shows already that we can come along with a total of $O(d^3)$ to compute the LOOCV error.

But we can further simplify the expression ~~(shown at the board)~~ and obtain:

Theorem

The LOOCV-RMSE of ridge regression can be computed in $O(d^3)$ through:

$$\text{RSME}_{\text{loocv}} = \sqrt{\sum_{i=1}^{n} \left(\frac{\mathbf{x}_i^\top \mathbf{w}_{\text{RR}} - y_i}{1 - \mathbf{x}_i^\top A^{-1} \mathbf{x}_i} \right)^2},$$

where $A := XX^\top + \frac{1}{2C} I.$

The Last Equation From the Previous Slide ...

... shows already that we can come along with a total of $O(d^3)$ to compute the LOOCV error.

But we can further simplify the expression ~~(shown at the board)~~ and obtain:

Theorem

The LOOCV-RMSE of ridge regression can be computed in $O(d^3)$ through:

$$\mathrm{RSME}_{\mathrm{loocv}} = \sqrt{\sum_{i=1}^{n} \left(\frac{\mathbf{x}_i^\top \mathbf{w}_{\mathrm{RR}} - y_i}{1 - \mathbf{x}_i^\top A^{-1} \mathbf{x}_i} \right)^2},$$

where $A := XX^\top + \frac{1}{2C} I$.

Order of computation:

- first compute A^{-1}, then $\mathbf{w}_{\mathrm{RR}}$, and last $\mathrm{RSME}_{\mathrm{loocv}}$

Proof

Recalling $\mathbf{w}_{\text{RR}} = A^{-1}Xy$ and denoting $\beta_i := \mathbf{x}_i^\top A^{-1}\mathbf{x}_i$, it is:

$\text{RSME}_{\text{loocv}}$

$$= \sqrt{\sum_{i=1}^{n} \left(\mathbf{x}_i^\top \left(A^{-1} + \frac{A^{-1}\mathbf{x}_i\mathbf{x}_i^\top A^{-1}}{1 - \mathbf{x}_i^\top A^{-1}\mathbf{x}_i} \right) (Xy - \mathbf{x}_i y_i) - y_i \right)^2}$$

Proof

Recalling $\mathbf{w}_{\mathrm{RR}} = A^{-1}Xy$ and denoting $\beta_i := \mathbf{x}_i^\top A^{-1}\mathbf{x}_i$, it is:

$\mathrm{RSME}_{\mathrm{loocv}}$

$$= \sqrt{\sum_{i=1}^{n}\left(\mathbf{x}_i^\top\left(A^{-1} + \frac{A^{-1}\mathbf{x}_i\mathbf{x}_i^\top A^{-1}}{1-\mathbf{x}_i^\top A^{-1}\mathbf{x}_i}\right)(Xy - \mathbf{x}_i y_i) - y_i\right)^2}$$

$$= \sqrt{\sum_{i=1}^{n}\left(\mathbf{x}_i^\top \mathbf{w}_{\mathrm{RR}} + \frac{\beta_i \mathbf{x}_i^\top \mathbf{w}_{\mathrm{RR}}}{1-\beta_i} - \beta_i y_i - \frac{\beta_i^2}{1-\beta_i} y_i - y_i\right)^2}$$

Proof

Recalling $\mathbf{w}_{\text{RR}} = A^{-1}Xy$ and denoting $\beta_i := \mathbf{x}_i^\top A^{-1}\mathbf{x}_i$, it is:

$\text{RSME}_{\text{loocv}}$

$$= \sqrt{\sum_{i=1}^{n} \left(\mathbf{x}_i^\top \left(A^{-1} + \frac{A^{-1}\mathbf{x}_i\mathbf{x}_i^\top A^{-1}}{1 - \mathbf{x}_i^\top A^{-1}\mathbf{x}_i} \right) (Xy - \mathbf{x}_i y_i) - y_i \right)^2}$$

$$= \sqrt{\sum_{i=1}^{n} \left(\mathbf{x}_i^\top \mathbf{w}_{\text{RR}} + \frac{\beta_i \mathbf{x}_i^\top \mathbf{w}_{\text{RR}}}{1 - \beta_i} - \beta_i y_i - \frac{\beta_i^2}{1 - \beta_i} y_i - y_i \right)^2}$$

$$= \sqrt{\sum_{i=1}^{n} \left(\left(1 + \frac{\beta_i}{1 - \beta_i} \right) \mathbf{x}_i^\top \mathbf{w}_{\text{RR}} - \left(\beta_i + \frac{\beta_i^2}{1 - \beta_i} + 1 \right) y_i \right)^2}$$

Proof

Recalling $\mathbf{w}_{\mathrm{RR}} = A^{-1}Xy$ and denoting $\beta_i := \mathbf{x}_i^\top A^{-1}\mathbf{x}_i$, it is:

$\mathrm{RSME}_{\mathrm{loocv}}$

$$= \sqrt{\sum_{i=1}^{n}\left(\mathbf{x}_i^\top\left(A^{-1} + \frac{A^{-1}\mathbf{x}_i\mathbf{x}_i^\top A^{-1}}{1-\mathbf{x}_i^\top A^{-1}\mathbf{x}_i}\right)(Xy - \mathbf{x}_i y_i) - y_i\right)^2}$$

$$= \sqrt{\sum_{i=1}^{n}\left(\mathbf{x}_i^\top \mathbf{w}_{\mathrm{RR}} + \frac{\beta_i \mathbf{x}_i^\top \mathbf{w}_{\mathrm{RR}}}{1-\beta_i} - \beta_i y_i - \frac{\beta_i^2}{1-\beta_i} y_i - y_i\right)^2}$$

$$= \sqrt{\sum_{i=1}^{n}\left(\left(1 + \frac{\beta_i}{1-\beta_i}\right)\mathbf{x}_i^\top \mathbf{w}_{\mathrm{RR}} - \left(\beta_i + \frac{\beta_i^2}{1-\beta_i} + 1\right) y_i\right)^2}$$

$$= \sqrt{\sum_{i=1}^{n}\left(\frac{\mathbf{x}_i^\top \mathbf{w}_{\mathrm{RR}} - y_i}{1-\beta_i}\right)^2}$$

Proof

Recalling $\mathbf{w}_{\mathrm{RR}} = A^{-1}Xy$ and denoting $\beta_i := \mathbf{x}_i^\top A^{-1}\mathbf{x}_i$, it is:

$\mathrm{RSME}_{\mathrm{loocv}}$

$$= \sqrt{\sum_{i=1}^{n}\left(\mathbf{x}_i^\top\left(A^{-1} + \frac{A^{-1}\mathbf{x}_i\mathbf{x}_i^\top A^{-1}}{1-\mathbf{x}_i^\top A^{-1}\mathbf{x}_i}\right)(Xy - \mathbf{x}_i y_i) - y_i\right)^2}$$

$$= \sqrt{\sum_{i=1}^{n}\left(\mathbf{x}_i^\top\mathbf{w}_{\mathrm{RR}} + \frac{\beta_i\mathbf{x}_i^\top\mathbf{w}_{\mathrm{RR}}}{1-\beta_i} - \beta_i y_i - \frac{\beta_i^2}{1-\beta_i}y_i - y_i\right)^2}$$

$$= \sqrt{\sum_{i=1}^{n}\left(\left(1+\frac{\beta_i}{1-\beta_i}\right)\mathbf{x}_i^\top\mathbf{w}_{\mathrm{RR}} - \left(\beta_i + \frac{\beta_i^2}{1-\beta_i} + 1\right)y_i\right)^2}$$

$$= \sqrt{\sum_{i=1}^{n}\left(\frac{\mathbf{x}_i^\top\mathbf{w}_{\mathrm{RR}} - y_i}{1-\beta_i}\right)^2} = \sqrt{\sum_{i=1}^{n}\left(\frac{\mathbf{x}_i^\top\mathbf{w}_{\mathrm{RR}} - y_i}{1-\mathbf{x}_i^\top A^{-1}\mathbf{x}_i}\right)^2}$$