Risk Estimation

-Introduction to Data Science-

ISL 5.1 (excluding 5.1.5), 6.1.3

Lecturer: Darren Homrighausen, PhD

Preamble:

- We outline some common tasks in a supervised problem and how that relates to risk estimation
- Discuss how the training error is optimistic and how to correct for this optimism
- Directly estimate the risk by resampling

Different notions of risk

There are actually a few types of risk we might consider

The details vary, but the story remains substantially the same

Some possible notions of risk:

- $R_{\text{pred}}(f) = \mathbb{E}[\ell(\hat{f}(X), Y)|\mathcal{D}]$
  (The training data is fixed and we average the loss over a test observation)
- $R(f) = \mathbb{E}[\ell(\hat{f}(X), Y)]$
  (The average is over the training data and a test observation)
- $R_{\text{estimation}}(\beta) = \mathbb{E}[\ell(\hat{\beta}, \beta_*)]$
  (The average is over estimates of a true parameter β_*, say in a linear model)
- $R_{\text{in}}(f)$ is the in-sample risk, which is like $R_{\text{pred}}(f)$, but with the test observation coming from the training values of X
  (See Chapter 7.4 in ESL for a more precise definition)

Risk estimation

Reminder: The risk can be written

$$R(f) = \mathbb{E}\ell(f(X), Y) \leftrightarrow \text{Bias} + \text{Variance}$$

The overriding theme is that we would like to add a judicious amount of bias to get lower risk

As R isn't known, we need to estimate it

As discussed, $\hat{R}(f) = \frac{1}{n}\sum_{i=1}^{n} \ell(f(X_i), Y_i)$ isn't very good

(In fact, one tends to not add bias when estimating R with $\hat{R}$)

Risk estimation scenarios

There are a few reasons why we may want to estimate the risk

Consider these scenarios:

1. I have a procedure f which has some parameters. I'd like to pick values for the parameters with smallest risk
2. I am considering two procedures, f_1 and f_2, for a particular application. I'd like to choose the one with smaller risk
3. I have a procedure f and a particular setting of any available parameters. I'd like to know the risk of this procedure

Suppose we have a risk estimate: call it $\widehat{\mathrm{Risk}}$ and the procedure f depends on parameter β

We use $\widehat{\mathrm{Risk}}$ for scenario 1. to produce a $\hat{\beta}$

If we then use the same $\widehat{\mathrm{Risk}}$ for scenario 3. for procedure f with that parameter $\hat{\beta}$, we will tend to underestimate the risk..

Risk estimation via splitting

The classical way to address this issue is via data splitting

The scenarios from the previous slide correspond to different splits:

1. Training: Used to fit (or train) the considered procedures
2. Validation: Used to score these trained procedures
3. Testing: Used to estimate the prediction risk for the selected procedure

A typical split might be 50%/25%/25%

(It is important to randomly assign observations to these splits)

This has some notable drawbacks:

- There needs to be a very large amount of data
- There can be issues with rare features
- The results can be sensitive to the splits used

Risk estimation without data splitting

Risk estimation: a general form

The reason that $\hat{R}(f) = \frac{1}{n}\sum_{i=1}^{n} \ell(f(X_i), Y_i)$ is a poor estimate of the risk is that it is overly optimistic

The average optimism is

$$\text{opt} =^* \mathbb{E}[R - \hat{R}]$$

Typically, opt is positive as $\hat{R}$ will underestimate the risk

($*$ See ESL, Chapter 7 for details for a more precise statement)

Risk estimation: a general form

It turns out for a variety of ℓ

$$\text{opt} = \frac{2}{n}\sum_{i=1}^{n}\text{Cov}(f(X_i), Y_i)$$

This is related intimately with degrees of freedom

$$\text{df} = \frac{1}{\sigma^2}\sum_{i=1}^{n}\text{Cov}(f(X_i), Y_i) = \frac{n}{2\sigma^2}\text{opt}$$

($\sigma^2 = \mathbb{V}Y_i$)

Example: For multiple regression (i.e. $\hat{f}(X) = \hat{\beta}_{LS}^\top X$),

- $\hat{\beta}_{LS} = (\mathbb{X}^\top\mathbb{X})^{-1}\mathbb{X}^\top Y$ (again, only unique if $\text{rank}(\mathbb{X}) = p$)
- $\hat{f}(X_i) = X_i^\top\hat{\beta}_{LS}$

  $\rightarrow \text{df} = \text{trace}(\mathbb{X}(\mathbb{X}^\top\mathbb{X})^{-1}\mathbb{X}^\top) = \text{rank}(\mathbb{X})$

A risk estimate

Therefore, we get the following general risk estimate:

$$\mathrm{GIC} = \hat{R} + \widehat{\mathrm{opt}}$$

(Writing GIC indicates generalized information criterion)

Differing $\widehat{\mathrm{opt}}$ lead to different risk estimators

$\widehat{\mathrm{opt}}$ depends on:

- a variance estimator $\hat{\sigma}$
- a scaling term

Various forms of risk estimates

$$\text{Akaike's information criterion: } \mathrm{AIC} = \hat{R} + \frac{2}{n} \cdot \mathrm{df} \cdot \hat{\sigma}^2$$

$$\text{Mallow's Cp} = \hat{R} - \hat{\sigma}^2 + \frac{2}{n} \cdot \mathrm{df} \cdot \hat{\sigma}^2$$

$$\text{Schwarz information criterion: } \mathrm{BIC} = \hat{R} + \frac{\log(n)}{n} \cdot \mathrm{df} \cdot \hat{\sigma}^2$$

Including more parameters leads to:

- a smaller $\hat{R}$
- a larger $\widehat{\mathrm{opt}}$

Goal: Now, we can use one of the GIC procedures to tell us which model to use

(As long as $\log n \geq 2$, BIC picks a smaller model than AIC)

Various forms of risk estimates: AIC and BIC

Akaike's Information Criterion (AIC) and the Schwarz/Bayesian Information Criterion (BIC) have alternative formulations:

$$\text{AIC} = \hat{R} + \frac{2}{n} \cdot \text{df} \cdot \hat{\sigma}^2 \qquad \text{or} \qquad n \log(\hat{R}) + 2 \cdot \text{df}$$

$$\text{BIC} = \underbrace{\hat{R} + \frac{\log(n)}{n} \cdot \text{df} \cdot \hat{\sigma}^2}_{\text{Use whenever}} \qquad \text{or} \qquad \underbrace{n \log(\hat{R}) + \log(n) \cdot \text{df}}_{\text{Only use when } n \geq p}$$

Estimating the variance

Some of the risk estimates in the preceding slides rely on a variance estimate: $\hat{\sigma}^2$

This can be a bit tricky in some situations

A general recommendation is to fit a large multiple regression procedure $\rightarrow \hat{f}$ with q features

We can produce a variance estimator as

$$\hat{\sigma}^2 = \frac{1}{n-q} \sum_{i=1}^{n} (Y_i - \hat{f}(X_i))^2$$

Cross-validation

A different approach to risk estimation

Let (X_0, Y_0) be a test observation, identically distributed as an element in $\mathcal{D}$, but also independent of $\mathcal{D}$.

$$R(f) = \mathbb{E}\ell(f(X_0), Y_0) \underbrace{=}_{\text{regression}} \mathbb{E}(Y_0 - f(X_0))^2$$

Of course, the quantity $(Y_0 - f(X_0))^2$ is an unbiased estimator of $R(f)$ and hence we could use it to estimate $R(f)$

However, we don't have any such new observation

And even if we did, this would be a highly variable estimate

(It only depends on one observation afterall)

We can address both of these issues..

An intuitive idea

Let's set aside one observation and predict it

For example: Set aside (X_1, Y_1) and fit $\hat{f}^{(1)}$ on $(X_2, Y_2), \ldots, (X_n, Y_n)$

(The notation $\hat{f}^{(1)}$ just symbolizes leaving out the first observation before fitting $\hat{f}$)

$$R_1(\hat{f}^{(1)}) = (Y_1 - \hat{f}^{(1)}(X_1))^2$$

As the left off data point is not one of the data points used for estimation,

$$\mathbb{E}R_1(\hat{f}^{(1)}) \approx R(\hat{f})$$

Leave-one-out cross-validation (LOOCV)

Cycling over all observations and taking the average produces leave-one-out cross-validation

$$\mathrm{CV}_n(\hat{f}) = \frac{1}{n}\sum_{i=1}^{n} R_i(\hat{f}^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{f}^{(i)}(X_i))^2.$$

Leave-one-out cross-validation (LOOCV)

This approach has some advantages with respect to the data splitting procedure

- LOOCV can be applied to data sets where data splitting is impractical

  (e.g. very small data sets)
- LOOCV is non-random
- $\mathbb{E}LOOCV \approx R(\hat{f})$, whereas the risk estimate produced via data splitting is usually larger than $R(\hat{f})$

It has some notable deficiencies as well:

- LOOCV can be computationally expensive

  (unless there is some trick (as in multiple regression) we have to refit the procedure n times)
- LOOCV can be high variance

  (Imagine having an extreme observation. This observation will substantially affect LOOCV)

The 'size' of a set

Some notation: Suppose v is a set

Example: if $v = \{1, 4, 10, -\pi\}$

It is useful to have notation for the size of a set

The relevant notion for size is the number of elements in that set

We will use $|v|$

For the v stated above, $|v| = 4$

More general cross-validation schemes

Let $\mathcal{N} = \{1, \ldots, n\}$ be the index set for $\mathcal{D}$

K-fold: Fix $V = \{v_1, \ldots, v_K\}$ such that

- $v_j \cap v_k = \emptyset$ (no observation belongs to more than 1 subset)
- $\bigcup_j v_j = \mathcal{N}$ (the union of all the subsets equals $\mathcal{N}$)

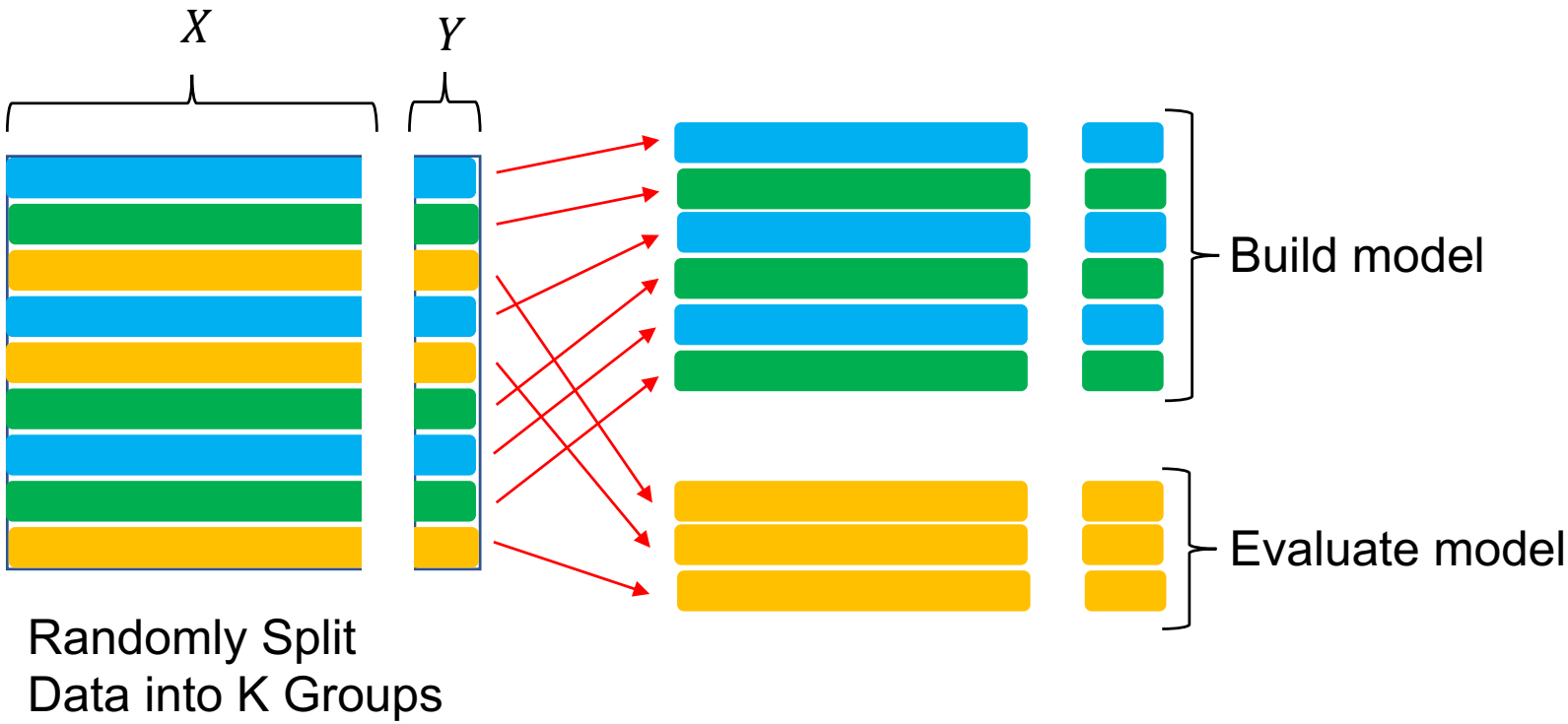
$$\mathrm{CV}_K(\hat{f}) = \frac{1}{K} \sum_{v \in V} \frac{1}{|v|} \sum_{i \in v} (Y_i - \hat{f}^{(v)}(X_i))^2$$

- average over the folds
- average over the validation (e.g. left out) observations
- the values of the loss function

Here, $|v| \approx \frac{n}{K}$

(Example, choosing $K = 2$ splits the data in half $\rightarrow |v| = \frac{n}{2}$ observations in each fold)

More general cross-validation schemes



Bias-Variance Trade-off for CV

LOOCV is nearly unbiased for $R(f)$

However, it is high variance

Data splitting can have a high bias for estimating $R(f)$ due to the reduced training set

K-fold CV with K between 5 and 10 tends to be an "intermediate" solution between these two extremes

Summary time

CV	+	Good at selecting models that make good predictions
	+/-	Generally selects a model larger than necessary
	-	Is computationally demanding, especially if K is large
AIC	+	Good at selecting models that make good predictions (and is asymptotically equivalent to CV)
	+/-	Generally selects a model larger than necessary
BIC	+	Good at selecting the correct model (if this exists)
	-	Generally selects model with poor prediction risk

Aside: There exist impossibility theorems stating that risk estimation procedures good at prediction are bad at model selection (and vice-versa)

Types of Risk

Returning to some possible notions of risk:

- $R_{\text{pred}}(f) = \mathbb{E}[\ell(\hat{f}(X), Y)|\mathcal{D}]$

  (The training data is fixed and we average the loss over a test observation)
- $R(f) = \mathbb{E}[\ell(\hat{f}(X), Y)]$

  (The average is over the training data and a test observation)
- $R_{\text{estimation}}(\beta) = \mathbb{E}[\ell(\hat{\beta}, \beta_*)]$

  (The average is over estimates of a true parameter β_*, say in a linear model)
- $R_{\text{in}}(f)$ is the in-sample risk, which is like $R_{\text{pred}}(f)$, but with the test observation coming from the training values of X

  (See Chapter 7.4 in ESL for a more precise definition)

In most cases, we want a procedure that has good risk in the sense of $R_{\text{pred}}(f)$

(e.g. we observe a data set and want to make predictions with it)

It turns out that

- CV actually estimates $R(f)$
- GIC actually estimates $R_{\text{in}}(f)$

No risk estimate exists that directly estimates $R_{\text{pred}}(f)$

Postamble:

- Discuss how the training error is optimistic and how to correct for this optimism

  (This generates AIC, BIC, Mallows, GCV, ...)
- Directly estimate the risk by resampling

  (Most commonly done with K-Fold CV or the bootstrap)