



Statistical Learning

Lecture 1: Bias-Variance Tradeoff

Yi He

January 9, 2023

Plan for Today

1. Introduction
2. Regression Function
3. Empirical Risk Minimization
4. Bias-Variance Tradeoff
5. Cross Validation

Introduction

Our Team



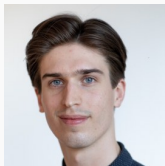
Dr. Yi He

Associate Professor (QE)

- I give all the lectures and am the chief examiner. Bring your mobile phone to the class and join live polling.
- Tutorials are by Rutger Poldermans and myself.
- Computer labs by Niels Marijnen and Floris Holstege.



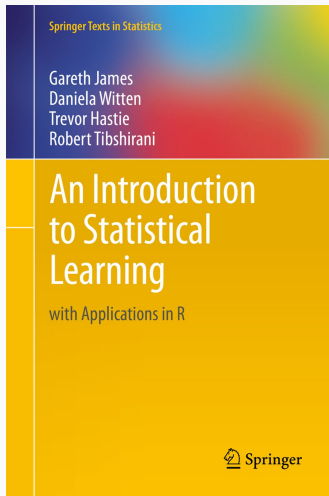
Rutger



Niels & Floris

Textbook: ISLR (1st Edition)

- Authors' PDF link on Canvas
- Chapters 1–8
- Textbook exercises in tutorials
- Applied exercises in labs



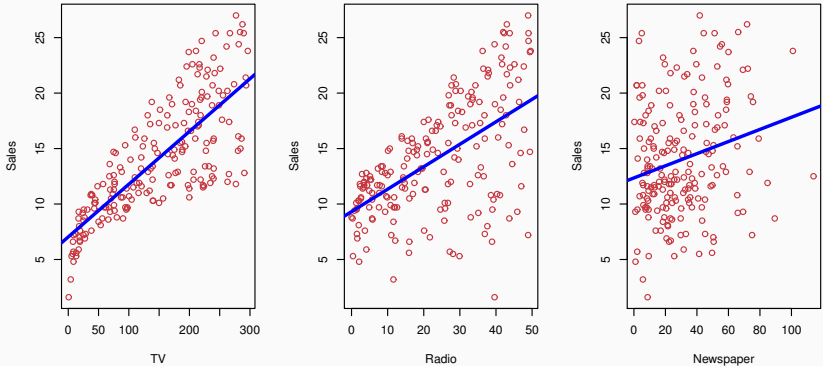
Schedule and Assessment

Date	Topics	Assignments
9 Jan	Bias-Variance Tradeoff	A1 available
11 Jan	High-Dimensional Linear Regression	
16 Jan	Shrinkage and Dimension Reduction	
18 Jan	Classification	A2 available, A1 due
23 Jan	Nonlinear Models	
25 Jan	Tree-based Methods	
30 Jan	Course Review	A2 due
3 Feb	Exam	Check Rooster

- Group computer assignments: $20\%+20\%=40\%$
- Written exam (theory), 2 hours: 60%
- Sign up for an assignment group (max 3 students) voluntarily **until 11 Jan** on Canvas. Otherwise we assign you to open groups randomly on 12 Jan.

Advertising Dataset

- Sales of a particular product
- Advertising budgets for three different media
- TV, Radio, Newspaper
- Adjusting advertising budgets may (indirectly) increase sales



How to Predict Sales?

- $Y = \text{Sales}$ is a *response* or *target*
- $\text{TV}, \text{Radio}, \text{Newspaper}$ are *features* or *predictors*
- $X = (\text{TV}, \text{Radio}, \text{Newspaper})^T$
- Our prediction is a function of features:

$$\widehat{\text{Sales}} = f(\text{TV}, \text{Radio}, \text{Newspaper}),$$

where f is some prediction rule.

- What prediction rule shall we use?

Regression Function

Squared Loss Function

- **Independent** test observation Y, X of the training sample.
- Given any **candidate** prediction rule $f \in \mathcal{F}$, the *squared loss* is given by

$$\ell(f(X), Y) = (Y - f(X))^2.$$

- The *population risk* (= expected loss)

$$R(f) \equiv \mathbb{E}[\ell(f(X), Y)] = \mathbb{E}[(Y - f(X))^2], \quad f \in \mathcal{F},$$

which depends on the joint distribution of Y, X .

- I will minimize the usage of brackets:

$$R(f) = \mathbb{E}[(Y - f(X))^2] = \mathbb{E}(Y - f(X))^2$$

Same for other formulas: **I won't repeat this announcement.**

- Recall the quadratic *risk function*

$$R(f) \equiv \mathbb{E} (Y - f(X))^2$$

- Tute Q2: the ideal prediction function minimizing $R(f)$ is given by

$$\mu(x) = \mathbb{E} [Y|X = x].$$

This is called the **regression** function.

- The regression function is **unknown** in practice.

Prediction Model

- The regression function $\mu(x)$ represents the systematic information that X provides about Y
- Without loss of generality, we can always construct

$$\epsilon = Y - \mu(X),$$

and the following regression model

$$Y = \mu(X) + \epsilon$$

where ϵ captures measurement errors and other discrepancies.

- Statistical learning aims to estimate the **fixed but unknown** function $\mu(x)$ from a set of data.
- State-of-art statistical learning methods usually means the machine learning methods with theoretical guarantee.

Empirical Risk Minimization

Empirical Risk Function

- A random sample $S = \{y_i, x_i : i = 1, \dots, n\}$ from some population distribution, satisfying the prediction model

$$y_i = \mu(x_i) + \varepsilon_i.$$

- The **empirical** risk for a **candidate** prediction rule $f \in \mathcal{F}$:

$$R_n(f) = \frac{1}{n} \sum_{i=1}^n \ell(f(x_i), y_i) = \frac{1}{n} \sum_{i=1}^n (y_i - f(x_i))^2$$

which is an estimator of the **population** risk function

$$R(f) \equiv \mathbb{E} (Y - f(X))^2.$$

- Shall we use the global empirical risk minimizer

$$\hat{f} = \operatorname{argmin}_{f \in \mathcal{F}} R_n(f)?$$

Go to www.menti.com and use the code 2810 8564

Do you think we should use the global empirical minimizer
(e.g., the least squares estimator)?

 Mentimeter

0

Yes, because the empirical risk approximates the
population risk

0

No, because the empirical risk is not equal to the
population risk

0

Maybe, it depends on the problems



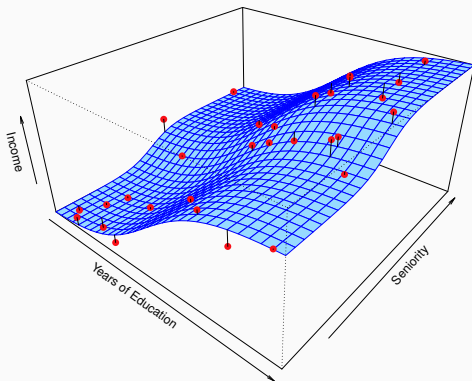
- One may wish that approximation error of the risk function

$$\sup_{f \in \mathcal{F}} |R_n(f) - R(f)| \text{ to be small.}$$

- In machine learning, we are generally interested in flexible models where \mathcal{F} is large.
- More flexible model uses larger \mathcal{F} , enlarges the approximation error and increases the variance of \hat{f} in general.
- If \mathcal{F} is very large, even the uniform convergence may fail as the variance of $R_n(f)$ becomes too large \Rightarrow the variance of \hat{f} is too large.

Simulation Example

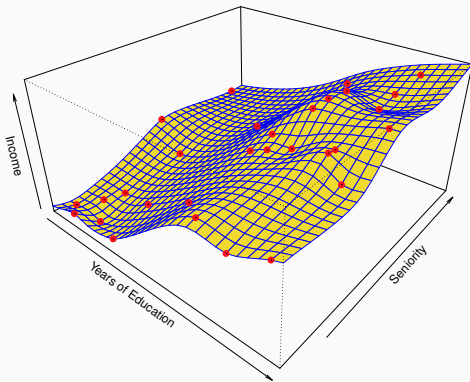
- $\text{Income} = \mu(\text{Years of Education}, \text{Seniority}) + \epsilon$
- True model shown below



ISLR Figure 2.3

Simulation Example: Continued

- When \mathcal{F} is large enough, we can fit the training data perfectly!
- That is, $y_i = \hat{f}(x_i)$ for all $i = 1, \dots, n$.
- Training MSE=0: better than the ideal predictor?



- Restrict to a smaller domain $\mathcal{H} \subset \mathcal{F}$ such that the law of large numbers still holds (or at least approximately so).
- The *regularized* estimator

$$\hat{f} = \operatorname{argmin}_{f \in \mathcal{H}} R_n(f)$$

- The *oracle* estimator

$$f_{\text{oracle}} = \operatorname{argmin}_{f \in \mathcal{H}} R(f)$$

- This may cause a bias $f_{\text{oracle}} - f_{\text{ideal}}$ if \mathcal{H} is too small: bias-variance tradeoff.

Bias-Variance Tradeoff

Reducible VS Irreducible Errors

- Consider an estimator $\hat{f}(x)$, possibly regularized, of the true regression function $\mu(x)$ depending only on the *training* sample
- Consider **independent** test observation Y, X from the same prediction model

$$Y = \mu(X) + \epsilon, \quad \mathbb{E}[\epsilon|X] = 0.$$

- Tute Q6: the prediction error

$$\begin{aligned}\mathbb{E}(Y - \hat{f}(X))^2 &= \mathbb{E}(\mu(X) + \epsilon - \hat{f}(X))^2 \\ &= \mathbb{E}(\mu(X) - \hat{f}(X) + \epsilon)^2 \\ &= \underbrace{\mathbb{E}(\mu(X) - \hat{f}(X))^2}_{\text{reducible}} + \mathbb{E}\epsilon^2\end{aligned}$$

- Irreducible component $\mathbb{E}\epsilon^2 = \text{Var}(\epsilon)$ does **not** depend on \hat{f} .

Bias-Variance Tradeoff

By law of iterated expectations,

$$\mathbb{E}(\mu(X) - \hat{f}(X))^2 = \mathbb{E}L(X),$$

where $L(x)$ denotes the reducible error at a given point x such that

$$L(x) \equiv \mathbb{E}(\mu(x) - \hat{f}(x))^2.$$

Tute Q3: we can decompose that

$$L(x) = \underbrace{(\mu(x) - \mathbb{E}\hat{f}(x))^2}_{\text{bias}^2} + \text{Var}(\hat{f}(x)).$$

- To improve the overall efficiency, we may consider biased estimators (often due to regularization) to reduce the estimation variance.

Cross Validation

Recall the regularized estimator

$$\hat{f} = \operatorname{argmin}_{f \in \mathcal{H}} R_n(f), \quad \mathcal{H} \subset \mathcal{F}$$

Suppose the subspace $\mathcal{H} = \mathcal{H}(\theta) \subset \mathcal{F}$ depends on some hyperparameters (tuning parameters) θ to be specified. Denote the corresponding estimator by

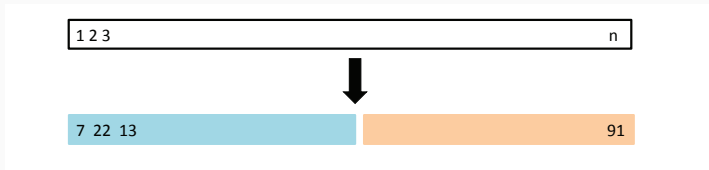
$$\hat{f}_\theta = \operatorname{argmin}_{f \in \mathcal{H}(\theta)} R_n(f).$$

How to tune the hyperparameters?

Validation Set Approach

Divide the total sample of size n , possibly randomly, into

1. *training* set of size n_t
2. *validation* set of size $n_o = n - n_t$



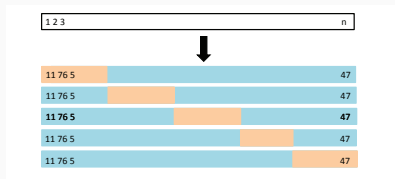
ISLR Figure 5.1

- Training and validation sets are exclusive.
- Fit the estimator \hat{f}_θ to the training set, evaluate its empirical risk on the validation set.

k -fold Cross Validation

- Divide the database into k (almost) equal-sized parts.
- Partition is often random.
- Leave out part i as validation set, fit the model to the other $k - 1$ parts combined as training set. The mean squared error on the validation set is denoted by $\text{MSE}_i(\theta)$.
- k -fold CV estimate

$$CV_{(k)}(\theta) = \frac{1}{k} \sum_{i=1}^k \text{MSE}_i(\theta)$$



Bias-Variance Tradeoff

- $k = n$ yields *Leave-one-out* CV: validation set size $n_o = 1$.
- Usually $k = 5$ or $k = 10$ in practice.

Large k (LOOCV):

- MSE_i are highly correlated, even for independent data
- ... Law of large numbers may fail !
- Computationally intensive

Small k (e.g., $k = 5$ and $k = 10$):

- Each training set only $\frac{k-1}{k}$ as big as the original one, and typically leads to overestimation.
- Less correlated estimates from each fold
- Smaller variance than LOOCV