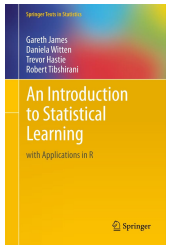
Introduction

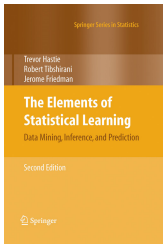
Lecture 1

Termeh Shafie

Course Literature



ISLR2 required
https://www.statlearning.com/



ESL optional

both free online!

Statistical Computing

- We will use the R programming language
  - R: a programming language for statistics
  - RStudio: a useful and convenient IDE for R

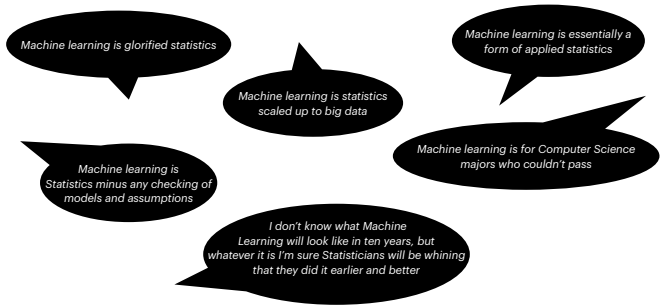
Course Format

- Lectures:
  - mixture of slides, notes and live coding examples
  - less technicalities, more intuition and applications
  - math intermissions when needed with **math cat**

- Practicals:
  - hands on practice in R (using the 'lab' section at the end of each ISLR chapter)
  - opportunity to get help with R for homework

What is Statistical Learning?
What is Machine Learning?

Some (More or Less Provocative) Answers



source: https://www.svds.com/machine-learning-vs-statistics

Statistical Learning vs. Machine Learning

The difference is not one of algorithms or practices but of **goals** and **strategies**

Statisticians focus more on
- uncertainty quantification
- theoretical guarantees on performance
- variations on well-established model classes
- applications in science and medicine

Machine learners focus more on
- algorithms and computation
- empirical performance on benchmark datasets
- inventing complex new methods/models
- applications in tech and industry

the data generating process

Induction vs. Deduction

Deductive inference
the process of reasoning from general premise(s) to reach a logically certain conclusion

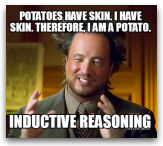
Example
- Premise 1: every person in this room is a student
- Premise 2: every student is older than 10 years
- Conclusions: every person in this room is older than 10 years

the truth of the premises guarantees the truth of the conclusion
but no natural way to deal with uncertainty regarding the premises

Induction vs. Deduction

Inductive inference
constructs or evaluates general propositions that are derived from specific examples

Example
- We drop things several times and they fall each time
- Conclusion: likely things always fall downwards when dropped



but we can never be sure, our conclusion can be wrong!
we draw uncertain conclusions from our relatively limited experiences

statistics are inherently inductive

Statistical Learning

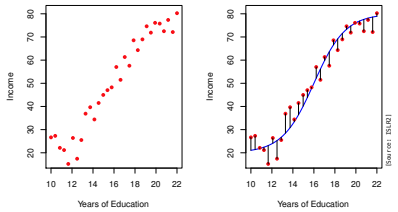
we will only be able to learn if there is something we can learn

- Output Y has something to do with input X
- "Similar inputs" lead to "similar outputs"
- There is a "simple relationship"/"simple rule" to generate output for a given input

we need a prior idea what we are looking for
→ **inductive bias** (learning impossible without such a bias)

- think of linear regression, what is the inductive bias here?

The Fundamental Problem

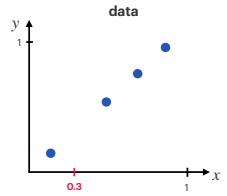


$$Y = f(X) + noise$$

How Estimate $f(X)$?

- We use training data to estimate $\hat{f}(X)$
- This allows us to predict Y when we know X: $\hat{Y} = \hat{f}(X)$
  - **Parametric methods** (we estimates the components of f)
    1. Functional form assumption e.g.
       $$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \varepsilon$$
    2. Estimation: a way to get $\hat{\beta}_j$ (e.g. OLS)
  - **Non-parametric methods**
    - No functional form assumption (e.g. splines)
    - Very flexible (both an advantage and disadvantage)
    - Usually requires more data

Example

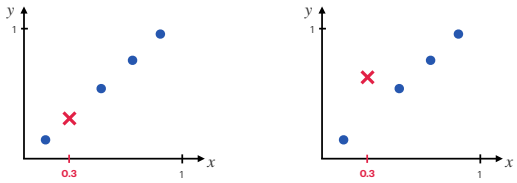


- Given: data with input-output pairs (X, Y)
- Goal: learn function that predicts Y values from the X values, i.e. $f : X \to Y$

what do you think is the value of $f(X = 0.3)$?

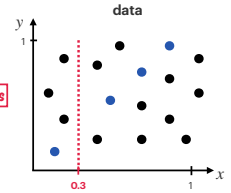
Example

here are two guesses



which one is better?

Example



the data generating process

- assume I tell you that the values y are generated by a uniform random number generator
- what would you now guess as $f(0.3)$?

Statistical Learning

$$Y = f(X) + noise$$

"statistical learning refers to a set of approaches for estimating f"
[ISLR2]

Considerations when choosing among methods:

- Supervised or unsupervised task?
- Is the outcome continuous or discrete?
- What is your goal: prediction or inference?
- How well does the model match the data generating process?
- Likelihood-based or algorithmic method?
- How big is n? How much flexibility is needed?

Supervised or Unsupervised?

- **Supervised learning**
  given training data examples$(x_1, y_1), \ldots, (x_n, y_n)$, we construct a function $\hat{f}(x)$ for predicting future values of y given x
  - Regression
  - Classification

- **Unsupervised learning**
  given training data examples$x_1, \ldots, x_n$, we compute some summaries such as cluster assignments, a low-dimensional projection, or parameters of the probability distribution of the x's.
  - Dimension reduction (e.g., PCA, ICA.)
  - Clustering

The Supervised Learning Problem

Starting point:

- Outcome measurement Y (also called dependent variable, response, target)
- Vector of p predictor measurements X (also called inputs, regressors, covariates, features, independent variables)
- In the regression problem, Y is quantitative (e.g income, price, blood pressure).
- In the classification problem, Y takes values in a finite, unordered set (survived/died, digit 0-9, cancer class of tissue sample, spam/legit email).
- We have training data$(x_1, y_1), \ldots, (x_n, y_n)$ which are observations (examples, instances) of these measurements

Goal:

On the basis of the training data we want to

- Accurately predict unseen test cases
- Understand which inputs affect the outcome, and how they do so
- Assess the quality of our predictions and inferences

Unsupervised Learning

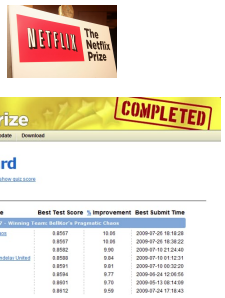
- No outcome variable, just a set of predictors (features) measured on a set of samples
- Objective is more fuzzy:
  - find groups of samples that behave similarly
  - find features that behave similarly
  - find linear combinations of features with the most variation
  - ⋮
- Difficult to know "how well" your are doing
- Can be useful as a pre-processing step for supervised learning

The Netflix Prize

- Competition started in October 2006. Training data is ratings for 18,000 movies by 400,000 Netflix customers, each rating between 1 and 5
- Training data is very sparse: about 98% missing
- Objective is to predict the rating for a set of 1 million customer-movie pairs that are missing in the training data
- Netflix's original algorithm achieved a root MSE of 0.953
- The first team to achieve a 10% improvement wins one million dollars

is this a supervised or unsupervised problem?

The Netflix Prize



Outcome Continuous or Discrete?

- **Continuous Outcomes: Regression**
  - Linear regression
  - lasso
  - elastic net
  - smoothing splines
  - KNN
  - support vector regression
  - regression trees

- **Discrete Outcomes: Classification**
  - Logistic regression
  - LDA
  - QDA
  - KNN
  - support vector machines
  - classification tree

Prediction or Inference?

Prediction

- We are only interested making an accurate prediction of y given x
  - Examples: predicting disease risk, detecting disease, predicting survival
- In this case, the prediction function $f(x)$ can be treated as a "black box"
- Example methods:
  - KNN
  - random forests
  - SVMs
  - smoothing splines
  - Gaussian processes
  - neural networks

  ...generally speaking, flexible/nonparametric methods

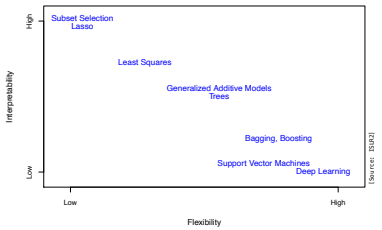
Prediction or Inference?

Inference

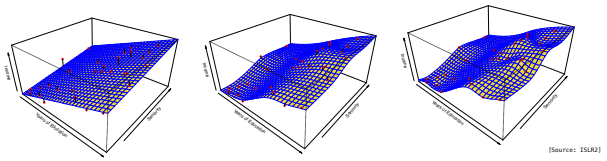
- We are more interested in understanding the relationship between x and y
- Typically involves inference for some parameters
  Examples: causal inference, genetic disease variants, finding biomarkers
- Interpretability is key: Which variables in x are important? What is the relationship between these variables and y?
- Example methods:
  - Linear regression
  - logistic regression and GLMs
  - lasso
  - elastic net
  - Bayesian models

  ...generally speaking, parametric or model-based methods

Tradeoff: Flexibility and Interpretability



Tradeoff: Accuracy and Interpretability



Does Model Match the Data Generating Process?

- Every method involves assumptions about the data distribution
  i.e. the data generating process (DGP)
- **Likelihood-based methods** are based on a probabilistic model for the data
  - assumptions are explicit ⇒ tend to be more interpretable
- **Algorithmic methods** specify an algorithm or objective function to optimize
  - assumptions are implicit ⇒ tend to be less interpretable
- Even the simplest method performs optimally if its assumptions match the DGP
- But if little is known about the DGP, then a more flexible method may be preferable
- Note: some methods are kind of in-between

Likelihood-based vs. Algorithmic method?

Likelihood-based methods

Examples: linear regression, logistic regression, GLMs, Bayesian models, Probabilistic PCA

Advantages:
- Interpretability
- Dependency structures identified
- Uncertainty quantification is straightforward
- Performance can be improved by theory
- Correctness/optimality guarantees

Disadvantages:
- Computationally intensive
- Less flexible

Likelihood-based vs. Algorithmic method?

Algorithmic methods

Examples: CART, random forests, neural networks, SVMs, ensembles, hierarchical clustering

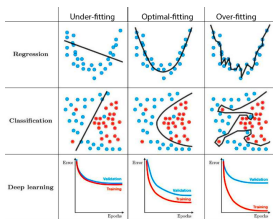
Advantages:
- Computationally fast
- Simpler to implement
- Excellent performance in practice

Disadvantages:
- Less interpretable
- Correctness/optimality requires more
- Uncertainty quantification often difficult

Sample Size vs. Flexibility

Statistical Concerns
- Overfitting and underfitting

Computational Concerns
- Complexity



The Goal of Statistical Learning

...is to "get knowledge" from the data, so that the information can be used for prediction, identification, understanding [ESL]

There is no free lunch in statistics: no one method dominates all others over all possible data sets

This Week's Practical: R you ready?