Introduction to Machine Learning for Social Scientists

Class 1: Introduction

Edgar Franco Vivanco

Stanford University
Department of Political Science

edgarf1@stanford.edu

Summer 2018

Introduction to Machine Learning for Social Scientists

Why Machine Learning?

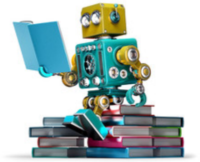
- The machine learning revolution:
  - Self-driving cars.
  - Translation.
  - Predict credit card fraud.
  - Predict consumer preferences.

Why Machine Learning?

- The machine learning revolution:
  - Self-driving cars.
  - Translation.
  - Predict credit card fraud.
  - Predict consumer preferences.
- Write programs to solve these issues became harder and harder.

Machine Learning

- Replication of how humans learn
- Train $\rightarrow$ Test $\rightarrow$ Repeat $\rightarrow$ Predict



Instead of writing programs that solve the problem, write programs that learn how to solve it...

Machine Learning Applications

- Industry
  - Measure consumer opinion
  - Deliver engaging content to users
- Public Sector
  - Predict disease onset
  - Assist criminal sentencing
- Campaigns
  - Classify voters based on likely voting, using consumer information
  - Identify ideology based on social media behavior
- Social Science
  - Infer extent and strategy of Chinese censorship: King, Pan and Roberts (2014)
  - Measure polarization in political institutions: Clinton, Jackman, and Rivers (2004)

Introduction to Machine Learning for Social Scientists

Examples of Learning Problems in Social Science

- Predict who will win the 2020 Presidential Election, based on public opinion polls and economic data.
- Estimate a person's wage based on age, education, and gender.
- Classify articles as either "fake news" or "real news" based on the words and the title
- Identify substantive topics in a collection of documents

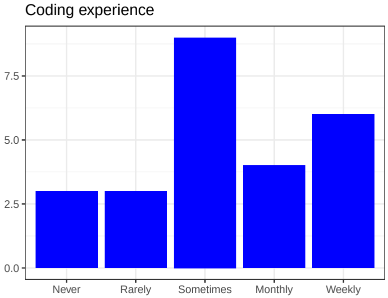
Introduction to Machine Learning for Social Scientists

Course history

- Developed in 2016-2017 by Justin Grimmer (as PoliSci 150)
  - I was a TA for the class
- Modified by Rochelle Terman in 2018
- Challenge:
  - 10 weeks of content into 8
  - Diverse group

Who are you?

- Affiliation:
  - Highschool students: 32%
  - Undergraduate from other universities: 44%
  - Graduate from other universities: 12%
  - Stanford undergrads:
  - Stanford grads: 12%
- OS:
  - Windows: 40%
  - Mac: 56%
  - Linux: 4%



About us

- Me: Edgar Franco Vivanco
- TA: Jesse Yoder
- TA: Haemin Jee

Introductory Approach

- Understand the **concepts** rather than the mechanics

Introductory Approach

- Understand the **concepts** rather than the mechanics
- Understand the **mechanics** rather than the math behind it

Introductory Approach

- Understand the **concepts** rather than the mechanics
- Understand the **mechanics** rather than the math behind it
- At the end of the course you should be able to understand the **intuition**, **strengths** and **weaknesses** of the various approaches.

Learning Goals

Ultimate Goal: Introduce students to modern machine learning techniques and provide the skills necessary to apply these methods widely.

Learning Goals

Ultimate Goal: Introduce students to modern machine learning techniques and provide the skills necessary to apply these methods widely.

Proximate Goals

1. Learn about the **core concepts** in machine learning and statistics, developing skills that are transferable to other types of data and inference problems.

Learning Goals

Ultimate Goal: Introduce students to modern machine learning techniques and provide the skills necessary to apply these methods widely.

Proximate Goals

1. Learn about the **core concepts** in machine learning and statistics, developing skills that are transferable to other types of data and inference problems.
2. Develop their programming abilities in the **R language**.

Learning Goals

Ultimate Goal: Introduce students to modern machine learning techniques and provide the skills necessary to apply these methods widely.

Proximate Goals

1. Learn about the **core concepts** in machine learning and statistics, developing skills that are transferable to other types of data and inference problems.
2. Develop their programming abilities in the **R language**.
3. Familiarize with some the **applications** of the models, including newspaper articles and podcast.
   - Applying ML methods to "real-world problems" requires both quantitative skills + social science reasoning.

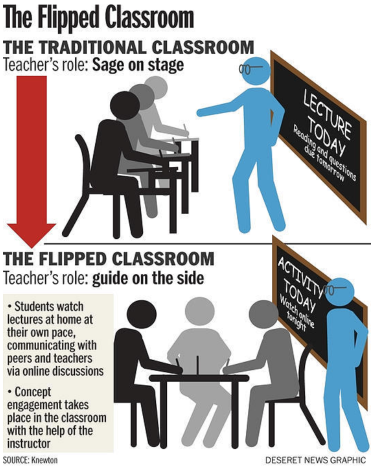
Learning Approach

Semi flipped classroom

- Teaching matters.
- 1/2 lecture, 1/2 coding in R
- Bring your laptop, and close it when necessary (laptop policy)
- Install, R, RStudio, and R markdown now!

Sections

- Review lecture materials, finish exercises
- Improve R programming



Course structure

- **Week 1 and 2:** Introduction to basic concepts and intro to R
- **Week 2 to 5:** Supervised learning
  - Simple and multiple regression
  - Classification
  - Cross-validation
- **Week 6:** Advanced supervised learning (LASSO)
- **Week 7:** Unsupervised learning
- **Week 8:** Review and group presentations

Grading Policy

- **40%:** 5 problems sets (8% each):
  - Learning by doing
  - Collaboration is encouraged
  - Submission via Canvas **on time**.
  - Normally posted after Wed class, due before Wed class next week (unless noted)

Grading Policy

- **40%:** 5 problems sets (8% each):
  - Learning by doing
  - Collaboration is encouraged
  - Submission via Canvas **on time**.
  - Normally posted after Wed class, due before Wed class next week (unless noted)
- **15%:** In class mid-term

Grading Policy

- **40%:** 5 problems sets (8% each):
  - Learning by doing
  - Collaboration is encouraged
  - Submission via Canvas **on time**.
  - Normally posted after Wed class, due before Wed class next week (unless noted)
- **15%:** In class mid-term
- **15%:** Final group project: Teach the class about a ML topic we didn't cover

Grading Policy

- **40%:** 5 problems sets (8% each):
  - Learning by doing
  - Collaboration is encouraged
  - Submission via Canvas **on time**.
  - Normally posted after Wed class, due before Wed class next week (unless noted)
- **15%:** In class mid-term
- **15%:** Final group project: Teach the class about a ML topic we didn't cover
- **20%:** Final exam

Grading Policy

- **40%:** 5 problems sets (8% each):
  - Learning by doing
  - Collaboration is encouraged
  - Submission via Canvas **on time**.
  - Normally posted after Wed class, due before Wed class next week (unless noted)
- **15%:** In class mid-term
- **15%:** Final group project: Teach the class about a ML topic we didn't cover
- **20%:** Final exam
- **10%:** Participation:
  - Attend class, ask questions, do not use your computer for something else than taking notes or working on class code.
  - Post on Canvas
  - Actively participate in weekly sections

Grading Policy and Accommodations

- All grades are final
  - No grade revision (but open to discussion on how to improve)
  - There is no curve
- Extensions will be given only to students with a documented emergency or illness.
- Let me know ASAP if you need special accommodations.

Materials & Communication

Canvas

- Lecture Notes, Code, and Data
- Homework (Assigned and returned)
- Questions and discussions
- Communicate with instructors and with each other

Email Policy

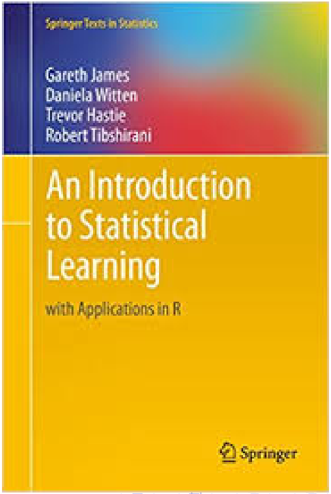
- Use Canvas first
- cc me in every communication with TAs (Allow 12hrs)

Office Hours

- Me (Wed 3.40pm to 5.40pm) https://www.wejoinin.com/sheets/veaqw
- Haemin (Mon 3:00pm to 5:00pm) https://www.wejoinin.com/hjee@stanford.edu
- Jesse (Th 2:30-4:30): https://www.wejoinin.com/sheets/sarpo

Book

- This book concentrates more on the applications of the methods and less on the mathematical details.
- You can read the book for free here.



Calibrating Expectations

- It is important to understand what are we learning and what are we not learning in the course.

Calibrating Expectations

- It is important to understand what are we learning and what are we not learning in the course.
- We are NOT going into the technical details of machine learning methods (optimization algorithms and theoretical properties)

Calibrating Expectations

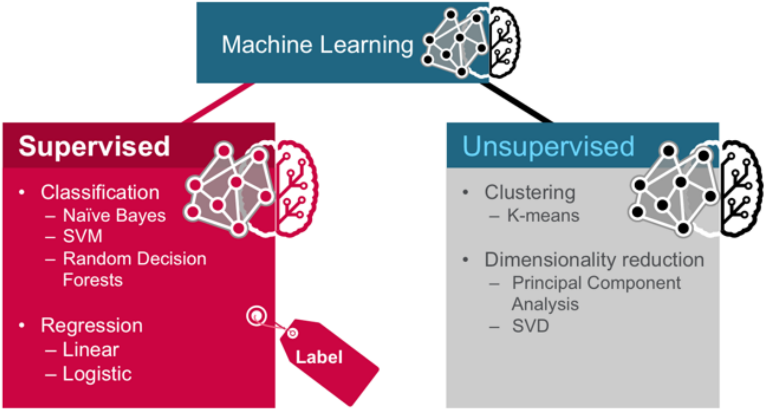
- It is important to understand what are we learning and what are we not learning in the course.
- We are NOT going into the technical details of machine learning methods (optimization algorithms and theoretical properties)
- We are NOT covering all the machine learning tools

Calibrating Expectations

- It is important to understand what are we learning and what are we not learning in the course.
- We are NOT going into the technical details of machine learning methods (optimization algorithms and theoretical properties)
- We are NOT covering all the machine learning tools
- We are NOT teaching you how to be a professional programmer or software developer.

Questions???

ML methods

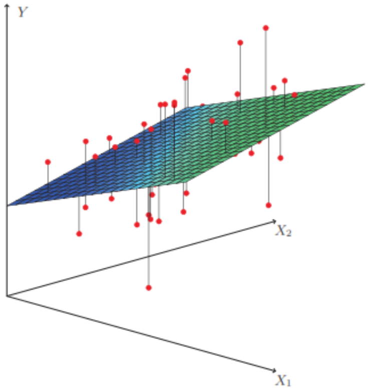

Machine Learning
Supervised
• Classification
– Naïve Bayes
– SVM
– Random Decision Forests
• Regression
– Linear
– Logistic
Label
Unsupervised
• Clustering
– K-means
• Dimensionality reduction
– Principal Component Analysis
– SVD

Supervised Methods:

- **Simple idea:** Human coders categorize a set of documents by hand, they create a gold standard.
- The algorithm then "learns" how to sort documents into categories.
- Steps:
  - Build a training set
  - Apply the method
  - Validate and classify the remaining documents

Individual Classification

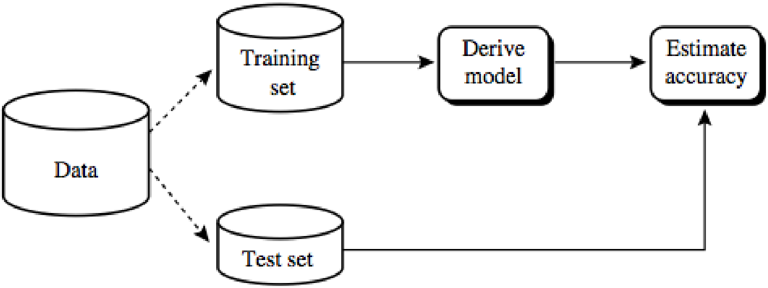
- Classify individual documents, cases, rows, into categories:
- Different models:
  - Linear Regression
  - Logistic Regression
  - LASSO.
  - Multinomial regressions
  - Support vector machines.
  - Random forest
  - Neural network



Divide your data

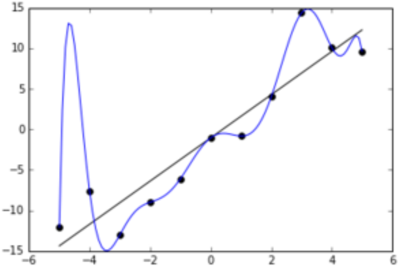
- **Training set:** A set of examples used to fit the parameters and learn. These are already classified by human coders to create a "gold standard".
- **Test set:** A set that follows the same probability distribution and is used to test de model.

Divide your data



Over fitting

- Over-fitting occurs when a model estimates a model that only works well for the training set, where we know our result.
- Risk: We are not really learning!



Performance

To asses the quality of our data we compare our classifications with the real data or the "gold standard".

Guess \ Actual	Yes	No
Yes		
No		

Performance

Guess \ Actual	Yes	No
Yes	True positive	False positive
No	False Negative	True Negative

How to be successful in this course

- Practice, practice, practice.

How to be successful in this course

- Practice, practice, practice.
- Program a little bit every day.

How to be successful in this course

- Practice, practice, practice.
- Program a little bit every day.
  - At the end of this course you'll have between 60-100 hrs of coding experience
- Collaborate
- Ask questions, either in class or talking directly to us.
- Stay organized

NEXT

- R!
  - Install R, R studio and R Markdown
  - If you have no or little experience with R, take these online tutorials before next class:
    - www.datacamp.com
    - Section Intro to basics
- Readings:
  - Google DeepMind's AI program learns human navigation skills
  - How babies learn and why robots can't compete
    - Podcast version
- Enroll in a section via Canvas