developing
trade consultants

policy • research • capacity building

Lecture 0: Introduction and Course Overview

Ben Shepherd, Principal.

Ben@Developing-Trade.Com

Motivation

▸ Terms like "algorithm", "machine learning", "deep learning", and "artificial intelligence" are all around us.

▸ Economists are starting to apply these tools in research applications, but it is still very limited in extent.

▸ Most graduate programs do not have dedicated programs on these tools.

▸ But… the basis is familiar to us from applied statistics and econometrics.

  ▸ Some techniques are close to the regression paradigm.
  ▸ Others are further removed, but still recognizable.

Motivation

- Economists have historically dealt almost exclusively with numbers as data.
  - We use a set of numbers to predict another number, for example.

- But recent advances in computer science make it possible to treat text as data as well.

- Application of this approach in economics is also in its infancy.

- There is a link to ML: some applications have been shown to work very well at using text to predict outcomes, whether a numerical variable or a categorical description.

Motivation

▸ Given the relative lack of current applications, there is a huge opportunity for applied researchers to mobilize these tools:

  ▸ Automate and systematize some of the work we already do.
  ▸ Enable us to provide new and creative answers to problems that have received little attention in the past, or incomplete answers.
  ▸ Enable us to ask completely new questions.

▸ The relationship to traditional econometrics is very much about complementarity: every applied economist needs to have a well-stocked econometric toolkit, but investing in developing some ML skills as well potentially has a very high payoff.

▸ Not every problem should be analyzed using ML tools. As a gross rule of thumb, ML is relatively good at prediction, but traditional econometrics is relatively good at inference.

▸ Likely to be a growth area in the future, so an important part of staying current.

Motivation

- For those who enjoy data work and programming, there are already many resources online to help get started with ML.

- But the applications are typically of little direct interest to economists, in particular applied researchers in policy-related fields.

- Examples include:
  - Image recognition.
  - Predictive text.
  - Classification of literary works.
  - Prediction of temperature and similar physical variables.
  - Classification of product reviews.

- So this course is designed to provide a "crash course" in ML techniques, including "text as data", with policy-relevant economics applications.

- Crash course = focus on intuition and application skills, no detailed treatment of technical aspects.
  - Refer to standard texts for detailed derivations and proofs.

Course Outline

Modules

1. Introduction to ML
   ▸ Live session: 7 April @ 8am

2. Introduction to Text Mining
   ▸ Live session: 8 April @ 8am

3. Introduction to Text-Based Prediction and Classification
   ▸ Live session: 12 April @ 8am

Contents

▸ Regression-type ML approaches, focused on the Lasso.
▸ General ML workflow.

▸ How to treat text as data.
▸ Manipulations and basic metrics.

▸ Overview of artificial neural networks.
▸ Use of text as an input into an ANN.

Course Outline

Element	Tasks
1. Lecture videos	▸ Watch in own time, note questions and points to clarify. ▸ Replicate demonstration code.
2. Discussion questions	▸ Prepare answers to be ready for live discussion.
3. R Problems	▸ Adapt demonstration code to answer new questions. Prepare answers and be ready for live discussion.
4. Live discussion	▸ Discuss questions from the lecture, analyze answers to discussion questions, consider answers to R problems.

Tips for Getting the Most out of the Course

▸ The key is "learning by doing": the lectures introduce concepts and motivate the material, but the detail is really in the demonstration code and problems.

▸ Work through the demonstration code in detail, make sure you understand it.
  ▸ There are many ways to code something, so you may find alternative (better!) ways of doing it.

▸ Spend real time with the problems, to find the best solutions you can.
  ▸ If you get stuck on a programming question, use Google: the answer is frequently on a coding forum, and can easily be adapted.

▸ Remember that we are interested in becoming familiar with tools and methods, not immediately developing publication-ready solutions! Think of the demonstrations and problems as starting points for research.

▸ Some of the material is hard. To stay motivated, keep thinking "what will this let me do in my own work?".

Tips for Getting the Most out of the Course

▸ Make full use of online resources.

▸ Each lecture ends with a slide of additional resources.

▸ But this is a very active area, so there are many others to choose from. Use Google, ask colleagues, and keep looking until you find resources that work for you, given your background and interests.

▸ Try to use the discussion questions to help come up with your own research ideas that could use these methods.

Tips for Getting the Most out of the Course