Lecture 7
Introduction to Machine Learning

Pierre Biscaye
Université Clermont Auvergne

Data Science for Economics

Note: Materials for this lecture are drawn from the UC Berkeley D-Lab's Python Machine Learning course.

What is machine learning?

- Machine learning is ”training” a model to perform some task based on data
- Tasks can be broadly defined
- Works best with large, complex datasets where “classical” statistical analysis would be impractical
- Types of machine learning:
  - Supervised ML: regression, classification
  - Unsupervised ML: clustering, dimensionality reduction
  - Other: reinforcement learning, semi-supervised learning, deep learning

ML is an increasingly important tool – in data science and in economics

ML-Enabled Econometrics with Unstructured Data

Paper Session

Sunday, Jan. 5, 2025 · 10:15 AM - 12:15 PM (PST)

Hilton San Francisco Union Square, Union Square 17 and 18

Hosted By: AMERICAN ECONOMIC ASSOCIATION
Chair: Szymon Sacher, Stanford University

Unstructured Data, Econometric Models, and Estimation Bias

Max Wei, University of Southern California
Nikhil Malik, University of Southern California

View Abstract

Debiasing Machine-Learning- or AI-Generated Regressors in Partial Linear Models

Jingwen Zhang, University of Washington
Wendao Xue, University of Washington
Yifan Yu, University of Texas-Austin
Yong Tan, University of Washington

View Abstract

Inference for Regression with Variables Generated by AI or Machine Learning

Laura Battaglia, Oxford University
Timothy Christensen, University College London
Stephen Hansen, University College London
Szymon Sacher, Stanford University

View Abstract · Download Preview (PDF, 1.03 MB)

Demand Estimation with Text and Image Data

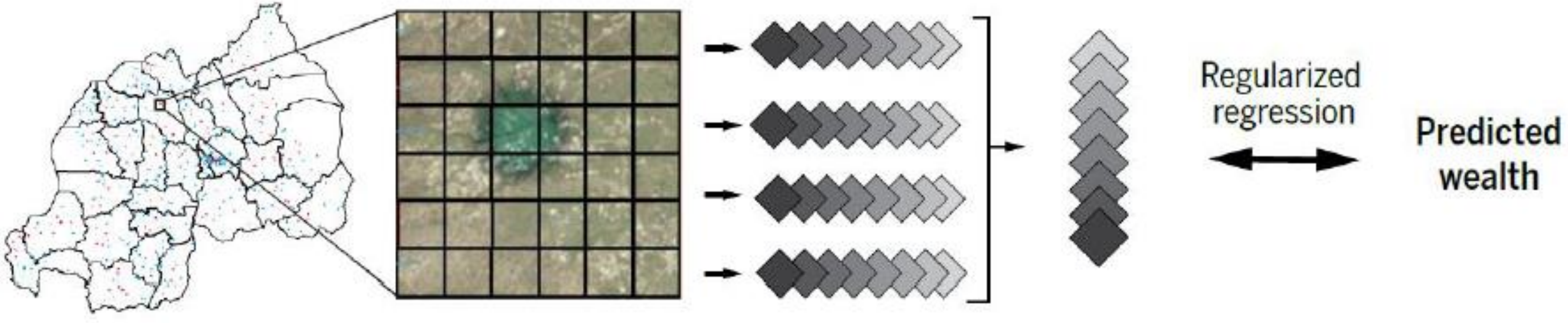
Giovanni Compiani, University of Chicago
Ilya Morozov, Northwestern University
Stephen Seiler, Imperial College London

View Abstract

We've seen applications already

Satellite imagery for poverty mapping (e.g., Jean et al 2016)

1. Feature extraction: transfer learning
   1. Better performance than raw features or PCA
2. Spatial join at "cluster" level of satellite and survey data
3. Modeling: ridge regression
4. Prediction: repeated cross-validation



Machine learning in this course

- Machine learning is a huge field
- What we will cover: examples of where to start in coding machine learning models
  - Introduction to the most standard algorithms via the most common packages.
  - Useful examples of machine learning code that covers a broad set of tasks it can perform.
- Python implementation
  - We'll mainly be using the **scikit-learn** package: has all of the standard algorithms, tons of support, and can run on a regular laptop.
  - Alternative packages you may explore on your own: ISLP, Pytorch
- Objective: get you familiar with machine learning so you can go into more depth with more confidence

Outline of what we will cover

- Preparing data for machine learning
- Supervised ML:
  - Regression: OLS, nearest neighbors, ridge, lasso
  - Classification: logistic regression, support vector machines, decision trees, random forests
  - Feature engineering
  - Regularization (avoiding overfitting)
  - Evaluating performance
  - Hyperparameter choice and validation
- Unsupervised ML:
  - Clustering
  - Dimensionality reduction

Python ML coding resources

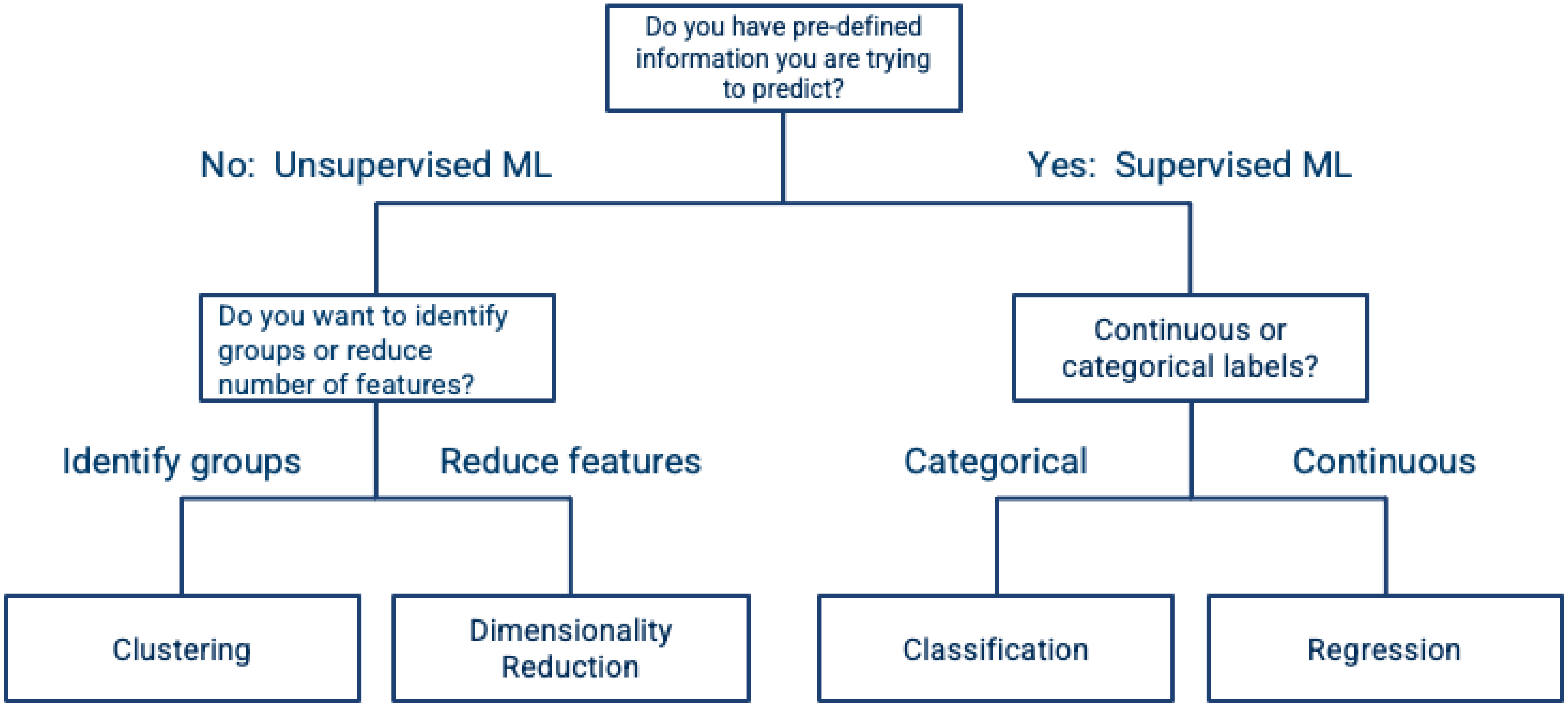
- https://github.com/dlab-berkeley/Python-Machine-Learning
  - The primary source for the notebooks in this course
- https://aeturrell.github.io/coding-for-economists/ml-intro.html
  - Nice overview following a similar structure to the D-Lab course
- https://github.com/jdnmiguel/Applied-ML
  - Master's course in applied machine learning from Jeremy do Nascimento Miguel, primarily using ISLP
- There are many more you can explore!

How do ML algorithms learn?

1. Preprocess your data
2. Specify a model
3. Train the model
4. Evaluate the model
5. Start again: cross-validate to improve performance

Focus in this course: implementation, not theory

Choosing the appropriate ML task

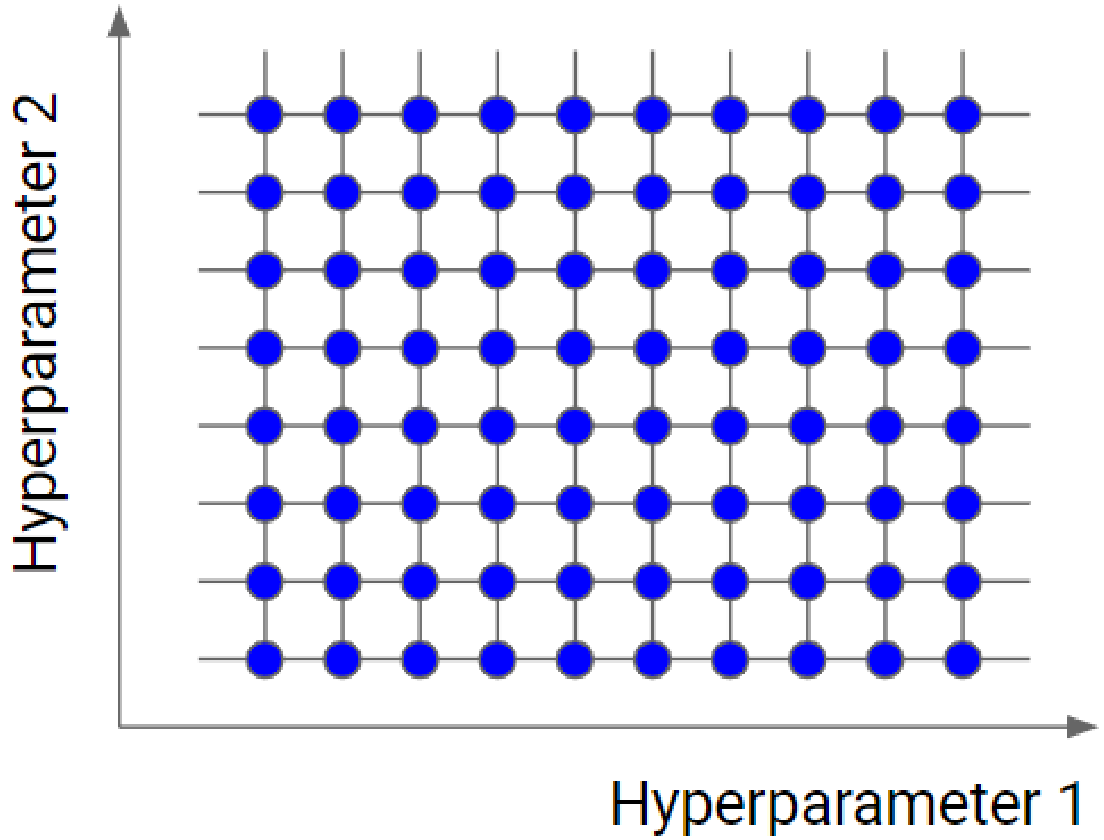


Choosing a model

- Not every model or algorithm works for every type of data.
- Regression: OLS, nearest neighbors, ridge, lasso, …
- Classification: logistic regression, decision trees, support vector machines, random forests, …
- Understanding the workings of models can help us choose the best one – we also often just try many different ones!

Choosing model parameters

- Models will have different hyperparameters that can be changed to optimize the results
- Most commonly, parameters are optimized by using grid search where many possible combinations are tried.



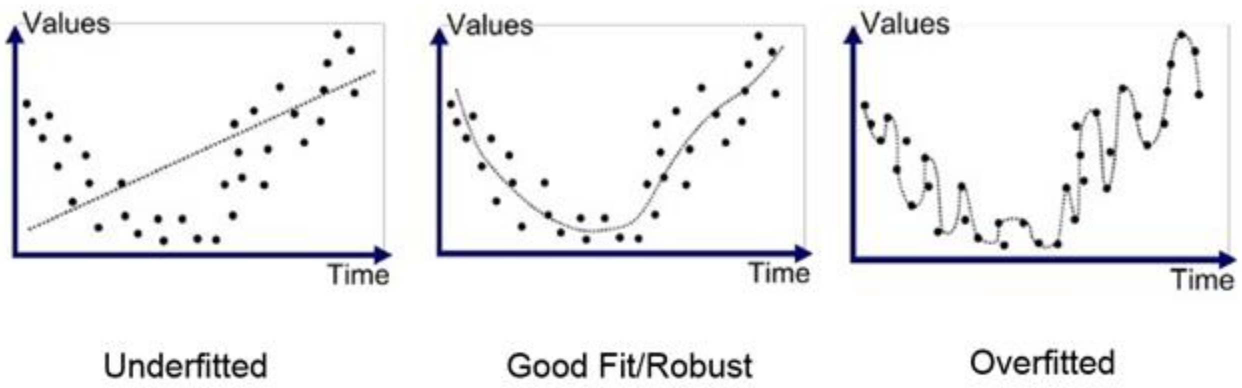
Source

Training a model

- Models are trained with preprocessed **training** data, followed by an evaluation on **test** data.
- Models have a **cost function** (also called the “loss” or “objective” function) that evaluates how well the model is performing on the data.
- Models have an **optimizer** that changes the predictions of the model to minimize the cost.
- We will look at specific examples of cost functions and optimizers in specific model architectures.

Evaluating a model

- Specific metrics for evaluating the quality of a model is based on task. They may differ from the cost function.
- Generally, we want models that perform well on both the training and the test data.
- A model that performs poorly on the training data is called **underfit**.
- A model that performs well on the training data and poorly on the testing data set is called **overfit**.



Source

Cross-validating a model

- The objective of the model is to accurately predict new data.
- To accomplish this, we want to avoid overfitting to any set of training data.
- One way to do this is to split the training data sample into additional sub-samples (or **folds**) and doing **cross-validation** on these folds.
- We test how models trained on all folds but one perform on predicting values in the remaining fold, and iterate across folds.
- The result is a model that performs better across sub-samples of training data, rather than fitting any one sample too closely.