Machine Learning: Applications and Opportunities in Social Science Research
ICPSR Summer Program in Quantitative Methods of Social Research
3 August – 7 August, 2020 (9-5pm)

Instructor:
Christopher Hare
Assistant Professor
Department of Political Science
University of California, Davis
cdhare@ucdavis.edu

Course Description:
A growing number of social scientists are taking advantage of machine learning methods to uncover hidden structure in their data, improve model predictive power, and gain a better understanding of complex relationships between variables. This workshops covers the mechanics underlying machine learning methods and discusses how these techniques can be leveraged by social scientists to gain new insight from their data. Specifically, the workshop will cover both supervised and unsupervised methods: decision trees, random forests, boosting, support vector machines, neural networks, deep and adversarial learning, ensemble learning, principal components analysis, factor analysis, and manifold learning/ multidimensional scaling. We will also discuss best practices in fitting and interpreting these models, including cross-validation techniques, bootstrapping, and presenting output. The workshop will demonstrate how these models can be estimated in R.

Recommended Texts and Readings:


15. **R**

   (a) James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani. 2013. *An Introduction to Statistical Learning with Applications in R*. New York: Springer.


16. **Python**

Required software: R is available for download from CRAN (Comprehensive R Archive Network): https://cran.r-project.org/ You’ll want to install the most recent (compatible) version. If installing on Windows, I recommending also downloading and installing Rtools (this is optional, but comes in handy if you ever need to compile or test a package).

I highly recommend the use of a text (or line) editor. Line editors are designed for writing and modifying programming code, and have useful functionality (e.g., macros) for programmers. Chris recommends the line editor Atom, which is open-source and free (https://atom.io/). Sublime and Notepad++ are other popular options.

Many people prefer to use RStudio to run R, which is perfectly fine.

You’ll want to install the following packages in R:

    install.packages(c("BMS","caret","ClassDiscovery","corrplot", "doParallel","dplyr","extraTrees","fastICA","foreach","foreign","gbm", "GenAlgo","ggfortify","ggplot2","kernlab","lavaan","MASS","MCMCpack", "mlbench","nnet","pROC","quadprog","randomForest","RANN"))

Course materials: Course materials (including slides, code, and problem sets) will be available on the course Canvas page.

Tentative Schedule:
This schedule is subject to change:

• Day One: Machine Learning: Theory and Concepts

  Computational Learning Theory and the Development of Machine Learning
  The Bias-Variance Tradeoff and Error Rates
  Model Validation and Tuning
  Resampling Techniques
  Predictions and Counterfactuals
  Quick Review of Linear Regression Models
  Programming in R
  Computing Performance and Practical Tips

• Day Two: Supervised and Semi-Supervised Learning I

  k-nearest Neighbors
  Generalized Linear Models and Extensions
• Day Three: Supervised and Semi-Supervised Learning II


k-nearest Neighbors
Generalized Linear Models and Extensions
Shrinkage/Regularization Methods and the Lasso
Regression Splines and Generalized Additive Models
Linear and Flexible Discriminant Analysis
Naive Bayes
Classification and Regression Trees
Neural Networks and Generative Adversarial Networks
Support Vector Machines and Relevance Vector Machines

• Day Four: Interpretable Machine Learning and Learning Ensembles


Bayesian Model Averaging
Ensemble Methods: Random Forests and Boosting
Assessing Variable Importance and Effects
Partial Dependency Plots and Model Visualization
Ensemble Modeling and Heterogeneous Treatment Effects

• Day Five: Unsupervised Learning

*k*-means Clustering
Principal Components Analysis
Manifold Learning and Multidimensional Scaling
Self-Organizing Maps
Deep Learning
Mixture Models and Latent Class Analysis
Novelty/Outlier Detection