DS-GA 3001: Tools and Techniques for Machine Learning (Spring 2021)

Instructor: David S. Rosenberg

Course description

This course deals with a range of topics that come up when applying machine learning in practice. Roughly half the course will cover topics connected to machine learning with interventions, such as counterfactual learning, reinforcement learning, and causal inference. Inverse propensity methods for handling biased samples and control variate methods for reducing variance will be given special attention, as these form a common thread of techniques relevant to each of these topics. We will also cover calibrating probability forecasts, interpreting machine learning models, active learning, crowdsourcing and "data programming", as time permits.

Prerequisites

- DS-GA 1003: Machine Learning or equivalent.
- DS-GA 1002: Probability and Statistics or equivalent.
- Comfort with conditional expectations, conditional probability modeling, basic Bayesian statistics, hypothesis testing and confidence intervals.
- Python programming required for most homework assignments.

Schedule

DISCLAIMER: We will cover the majority of the topics below, but the organization and specific topics may change. In particular, the topics of the first 8 weeks can be quite challenging, and if we need to take more time with them, we may **drop some of the topics at the end of the syllabus**.

- **Week 0**: Conditional expectation and variance decomposition
- **Week 1**: Estimating a population mean with a biased sample
- **Week 2**: Machine learning for causal inference
- **Week 3**: Exploration vs exploitation for bandits

- **Week 4**: Counterfactual policy evaluation
- **Week 5**: Counterfactual learning
- **Week 6**: Introduction to reinforcement learning
- **Week 7**: Catch-up and review
- **Week 8**: Calibrated probability predictions
- **Week 9**: Methods for global feature importance
- **Week 10**: Explaining black-box model predictions
- **Week 11**: Crowdsourcing
- **Week 12**: Active learning
- **Week 13**: Weak supervision and "Data Programming"
- **Week 14**: Catch-up, review, and conclusions

Course Requirements and Evaluation

- **(50%) Homework**: 4 – 5 homework assignments; mix of model building and written mathematical exercises to reinforce the main concepts.
- **(20%) Weekly Quizzes**: Concept-check quizzes that reinforce the main ideas from lectures and lab, which students may use any resources to complete.
- **(30%) Project**: In groups of 2–4, reproduce the experiments from a paper of relevance to the course and extend them in some way (e.g. an additional dataset, a new evaluation process, comparing to another method, etc.).

Topic Details

Estimating a population mean with a biased sample

There are certain challenging ideas and techniques that come up repeatedly in the first part of our course (in causal inference, counterfactual learning, and reinforcement learning). We will introduce them here in the simplest possible setting: estimating the mean of a population with a biased sample.

- Imputation, inverse propensity, self-normalization, and [possibly] doubly robust methods Seaman and Vansteelandt (2018); Kang and Schafer (2007)
- Control variates for variance reduction (Owen, 2013, Sec 8.9)

Machine learning for causal inference

When machine learning is applied in practice, it is often used to guide **interventions** in the world that we hope will improve some outcome measure. When we start making interventions, one of the most basic questions we can ask is which of two interventions (such as a treatment and a control) is better. In a basic statistics class, we learn how to estimate the "average treatment effect" (ATE) when individuals are assigned to a treatment or control group with equal probability. In this module, we discuss how to estimate the ATE when individuals are assigned to interventions with probabilities that depend on covariates (i.e. characteristics/features of the individuals). Of course, interventions may have better or worse performance depending on characteristics of the individuals. We will also discuss how to estimate these "conditional average treatment effects".

- Estimating average treatment effects with inverse propensity weighting and imputation
- Two trees algorithm for estimating conditional average treatment effects (CATE) Athey and Imbens (2015a)
- X-learner algorithm for CATE estimation Künzel et al. (2019)
  - (optional) Honest random forest Athey and Imbens (2015b)
  - (optional) Bayesian Additive Regression Trees (BART) Chipman et al. (2010)

Exploration vs exploitation for bandits

How can we balance "exploiting" interventions that worked well before (e.g. suggesting comedy movies for a particular individual) with "exploring" new intervention strategies (e.g. suggesting action movies) that may have better outcomes? In this module, we explore approaches to this classic "explore/exploit" problem. We will start with a focus on the simple "Bernoulli bandit" setting. Then we will introduce the more general contextual bandit setting, and discuss explore/exploit methods for that case as well.

- Gradient bandit algorithms (Sutton and Barto, 2018, Sec 2.8)
  - Using a "baseline" for variance reduction (a control variate technique)
- Thompson sampling for bandits and contextual bandits Chapelle and Li (2011); Russo et al. (2018)

Counterfactual policy evaluation

Suppose we believe that different interventions are preferable for different individuals, depending on their characteristics. Then we want to develop a "policy" that determines the interventions we take as a function of the characteristics of

the individual. Given two policies, the simplest way to compare their performance is with an "A/B test", which basically means deploying the two policies and seeing how they do. However, there can be very high costs to deploying a sub-optimal policy. Furthermore, there is a practical limit to how many policies we can test out and still get a reasonable estimate of the performance of each. In this module, we discuss how we can estimate the performance of a new policy without actually deploying it, using data that was already collected with a different policy. This data, collected from a so-called "logging policy", is called "logged bandit feedback". We will revisit our discussion of imputation, inverse propensity, and doubly robust methods and apply them to the problem of estimating the performance of a policy using logged bandit feedback.

- Extending the imputation, inverse propensity, and doubly robust methods to counterfactual policy evaluation from logged bandit feedback Dudík et al. (2011)

Counterfactual learning

In our module on counterfactual policy evaluation, we discussed some methods for estimating the performance of a new policy using logged bandit feedback. However, the uncertainty of these estimates can vary dramatically, depending on how different the new policy is from the logging policy. In this module, we discuss how to handle this uncertainty when it comes to **learning** an optimal policy from logged bandit feedback.

- Learning from logged bandit feedback (POEM) Swaminathan and Joachims (2015b,a)
- Propensity overfitting (self-normalized estimator) Swaminathan and Joachims (2015c)

Introduction to reinforcement learning

So far we've considered learning and evaluating policies in the contextual bandit setting, where we assume that the contexts we observe are i.i.d. In the reinforcement learning setting, sequences of contexts are grouped together into "episodes", which will have sequential dependencies. In particular, the action we take at one step in the episode may affect the next context we observe. In this module, we study "policy gradient" approaches for learning policies in this setting.

- Empirical risk minimization with black-box loss functions
- Policy gradient methods for reinforcement learning (Sutton and Barto, 2018, Ch 13)
  - Using a "baseline" for variance reduction (a control variate technique)

Calibrated probability predictions

For models that make probabilistic predictions, how can we ensure that they are both "calibrated" (e.g. the "70%" outcomes actually occur 70% of the time) and "sharp" (e.g. the probability predicted for the successful outcome of a surgery isn't just the overall success rate, but varies depending on as many characteristics of the individual as we can). It turns out, even assessing whether a model is calibrated is nontrivial. In this module, we discuss some classic and modern approaches to calibration and to assessing calibration.

- Assessing probabilistic predictions: ℓ_p calibration error, Brier score, and likelihood
- Basic calibration methods: histogram binning and Platt scaling Platt (1999)
- Bias/variance tradeoffs in assessing calibration
- The scaling-binning calibrator Kumar et al. (2019)

Feature importance

There are many methods that purport to measure the relative importance of various features in a model. As one digs in, one finds that there are about as many different methods for defining what is meant by feature importance. In this module, we discuss the many intepretations of "feature importance". We also present some of the most popular approaches to feature importance, along with a discussion of how they can be misinterpreted.

- Permutation feature importance Breiman (2001)
- Partial Dependency Plots (PDP) Friedman (2001)
- Individual Conditional Expectation (ICE) Plots Goldstein et al. (2013)
- Issues with above methods Hooker and Mentch (2019)

Explaining black-box model predictions

The previous module was about the relative importance of features in a model, as a whole. In this module, we discuss how to assess the contributions of each features to a **particular** model prediction. We'll discuss some recent approaches to these "local" model interpretations, as well as some of their issues.

- Local Interpretable Model-agnostic Explanations (LIME) Ribeiro et al. (2016)
- Shapley Additive Explanation (SHAP) Lundberg and Lee (2017); Lundberg et al. (2020)
- Debate about SHAP and similar interpretability methods Sundararajan and Najmi (2019); Kumar et al. (2020); Alvarez-Melis and Jaakkola (2018)

Crowdsourcing

For many problems in the real world, a major expense (time and money) in building a machine learning model is in the collection of labeled data. In this module and the following two modules we will address several aspects of this problem. In this module, we discuss how we can use "crowd workers" (generally non-expert, and with varying error rates) to generate reasonably reliable labels for our data. In particular, how many crowd workers should we get to label each example? How do we automatically resolve disagreements?

- Jointly estimating worker skill and ground truth with Dawid-Skene two-coin model Dawid and Skene (1979); Raykar et al. (2010); Zhang et al. (2016,?)
- Incorporating example difficulty Zhou et al. (2015)
- How many labels do we need per example? Khetan et al. (2017)

Active learning

Given a large pool of unlabeled examples and a finite budget for labeling these examples, can we do better than randomly sampling unlabeled examples to be labeled? This is the core question of the "active learning" problem. In this module, we discuss some classic approaches to active learning, as well as some refinements.

- Uncertainty Sampling Lewis and Catlett (1994)
- Query-by-committee Settles (2009)
- Selection with simpler proxy models Coleman et al. (2020)
- Evaluating active learning methods Yang and Loog (2016)

Weak supervision and "Data Programming"

Rather than labeling individual examples, we can consider getting experts to write "rules" for generating labels. For example, a rule might be "If the radiologist's report has the phrase 'is cancerous' then the corresponding image should be labeled as 'shows cancer'." In this module we discuss how we might use these imprecise rules to generate a useful training set of "weakly labeled" data.

- Human-generated rules as weak supervision (SNORKEL) Ratner et al. (2016)
- Matrix factorization for multitask weak supervision Ratner et al. (2018)

Academic Integrity Policy:

The course conforms to NYU's policy on academic integrity for students.

Moses Statement

Academic accommodations are available for students with disabilities. The Moses Center website is http://www.nyu.edu/csd. Please contact the Moses Center for Students with Disabilities (212-998-4980 or mosescsd@nyu.edu) for further information. Students who are requesting academic accommodations are advised to reach out to the Moses Center as early as possible in the semester for assistance.

References

Alvarez-Melis, D. and Jaakkola, T. S. (2018). On the robustness of interpretability methods. *CoRR*.

Athey, S. and Imbens, G. (2015a). Machine learning for estimating heterogeneous causal effects. Research papers, Stanford University, Graduate School of Business.

Athey, S. and Imbens, G. (2015b). Recursive partitioning for heterogeneous causal effects. *CoRR*.

Breiman, L. (2001). Random forests. *Mach. Learn.*, 45(1):5–32.

Chapelle, O. and Li, L. (2011). An empirical evaluation of thompson sampling. In *Proceedings of the 24th International Conference on Neural Information Processing Systems*, NIPS'11, pages 2249–2257, Red Hook, NY, USA. Curran Associates Inc.

Chipman, H. A., George, E. I., and McCulloch, R. E. (2010). Bart: Bayesian additive regression trees. *Ann. Appl. Stat.*, 4(1):266–298.

Coleman, C., Yeh, C., Mussmann, S., Mirzasoleiman, B., Bailis, P., Liang, P., Leskovec, J., and Zaharia, M. (2020). Selection via proxy: Efficient data selection for deep learning. In *International Conference on Learning Representations (ICLR)*.

Dawid, A. P. and Skene, A. M. (1979). Maximum likelihood estimation of observer error-rates using the em algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):20–28.

Dudík, M., Langford, J., and Li, L. (2011). Doubly robust policy evaluation and learning. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, pages 1097–1104, Madison, WI, USA. Omnipress.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Ann. Statist.*, 29(5):1189–1232.

Goldstein, A., Kapelner, A., Bleich, J., and Pitkin, E. (2013). Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *CoRR*.

Hooker, G. and Mentch, L. (2019). Please stop permuting features: an explanation and alternatives. *CoRR*.

Kang, J. D. Y. and Schafer, J. L. (2007). Demystifying double robustness: a comparison of alternative strategies for estimating a population mean from incomplete data. *Statistical Science*, 22(4):523–539.

Khetan, A., Lipton, Z. C., and Anandkumar, A. (2017). Learning from noisy singly-labeled data. *CoRR*.

Kumar, A., Liang, P., and Ma, T. (2019). Verified uncertainty calibration. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Kumar, I. E., Venkatasubramanian, S., Scheidegger, C., and Friedler, S. (2020). Problems with shapley-value-based explanations as feature importance measures. *CoRR*.

Künzel, S. R., Sekhon, J. S., Bickel, P. J., and Yu, B. (2019). Metalearners for estimating heterogeneous treatment effects using machine learning. *Proceedings of the National Academy of Sciences*, 116(10):4156–4165.

Lewis, D. D. and Catlett, J. (1994). Heterogeneous uncertainty sampling for supervised learning. In Cohen, W. W. and Hirsh, H., editors, *Proceedings of ICML-94, 11th International Conference on Machine Learning*, pages 148–156, New Brunswick, US. Morgan Kaufmann Publishers, San Francisco, US.

Lundberg, S. and Lee, S.-I. (2017). A unified approach to interpreting model predictions. pages 4765–4774.

Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., Katz, R., Himmelfarb, J., Bansal, N., and Lee, S.-I. (2020). From local explanations to global understanding with explainable ai for trees. *Nature Machine Intelligence*, 2(1):56–67.

Owen, A. B. (2013). *Monte Carlo theory, methods and examples.*

Platt, J. C. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *ADVANCES IN LARGE MARGIN CLASSIFIERS*, pages 61–74. MIT Press.

Ratner, A., Hancock, B., Dunnmon, J., Sala, F., Pandey, S., and Ré, C. (2018). Training complex models with multi-task weak supervision. *CoRR*.

Ratner, A., Sa, C. D., Wu, S., Selsam, D., and Ré, C. (2016). Data programming: Creating large training sets, quickly. In *Proceedings of the 30th International Conference on Neural Information Processing*, page 3574–3582.

Raykar, V. C., Yu, S., Zhao, L. H., Valadez, G. H., Florin, C., Bogoni, L., and Moy, L. (2010). Learning from crowds. *Journal of Machine Learning Research*, 11:1297–1322.

Ribeiro, M., Singh, S., and Guestrin, C. (2016). "why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 97–101, San Diego, California. Association for Computational Linguistics.

Russo, D. J., Roy, B. V., Kazerouni, A., Osband, I., and Wen, Z. (2018). A tutorial on thompson sampling. *Foundations and Trends® in Machine Learning*, 11(1):1–96.

Seaman, S. R. and Vansteelandt, S. (2018). Introduction to double robust methods for incomplete data. *Statistical Science*, 33(2):184–197.

Settles, B. (2009). Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison.

Sundararajan, M. and Najmi, A. (2019). The many shapley values for model explanation. *CoRR*.

Sutton, R. S. and Barto, A. G. (2018). *Reinforcement Learning: An Introduction*. A Bradford Book, Cambridge, MA, USA.

Swaminathan, A. and Joachims, T. (2015a). Batch learning from logged bandit feedback through counterfactual risk minimization. *Journal of Machine Learning Research*, 16(52):1731–1755.

Swaminathan, A. and Joachims, T. (2015b). Counterfactual risk minimization: Learning from logged bandit feedback. volume 37 of *Proceedings of Machine Learning Research*, pages 814–823, Lille, France. PMLR.

Swaminathan, A. and Joachims, T. (2015c). The self-normalized estimator for counterfactual learning. In Cortes, C., Lawrence, N. D., Lee, D. D., Sugiyama, M., and Garnett, R., editors, *Advances in Neural Information Processing Systems 28*, pages 3231–3239. Curran Associates, Inc.

Yang, Y. and Loog, M. (2016). A benchmark and comparison of active learning for logistic regression. *CoRR*.

Zhang, Y., Chen, X., Zhou, D., and Jordan, M. I. (2016). Spectral methods meet em: A provably optimal algorithm for crowdsourcing. *Journal of Machine Learning Research*, 17(102):1–44.

Zhou, D., Liu, Q., Platt, J. C., Meek, C., and Shah, N. B. (2015). Regularized minimax conditional entropy for crowdsourcing. *CoRR*.