Python and Machine Learning for Accounting Faculty

Dr. Richard M. Crowley

Version 2021.07.06

The below is a provisional version of the proposed training. Subjects covered are open to suggestions or recommendations.

Course structure

Each 3 hour session will cover a different methodology, either a machine learning technique or a more fundamental technique upon which machine learning methods can be applied. Each session will include a mix of: 1) An overview of the methodology, how it works, and what it is useful for, 2) An overview of 1 or more papers that use the methodology in the extant economics-based literature, and 3) A walk-through of how to implement the methodology in Python. For those more comfortable with R, a brief summary of how to implement the methods in R will also be provided.

Expectations

Keep an open mind. Some methods may not be common in published papers in accounting *yet*, but they are either already in use in working papers or are already more firmly established in related disciplines. As such, the papers mentioned throughout are often from outside Accounting. Papers are selected such that even if you are not familiar with some of the constructs the papers use, the importance and usefulness of the methodology used should be transparent.

Software

To keep consistent with the python training conducted in June 2021, we will use python along with Jupyter notebook (via Anaconda or Python itself). That being said, all methods covered in the sessions are applicable under *any* python workflow and are also easily implementable using R.

Notes

Suggested readings will be explicitly referenced in the session, and as such would be useful to read (or skim, for longer papers) in advance.

All **extra materials** are in the order that I would recommend reading them in. These are intended to be particularly useful to anyone interested in delving more into the particular methodology of a session but will at most be referenced in passing.

Content

Session 1: Statistical and machine learning regression

Concepts

1. Regression in Python.
   - How to run a simple regression using Pandas and Statsmodels.
2. Sample splitting for predictive analyses.
   - Statistical approach: Training versus Testing samples.
   - Supervised machine learning approach: Training, *Validation*, and Testing samples.
3. Simple machine learning-based regression: Introducing bias to improve out of sample predictions.
   - LASSO: Least Absolute Shrinkage and Selection Operator.
     - This is equivalent to L1 regularization or Least Angle Regression/LARS under some assumptions.
   - Elastic Net: A generalization of LASSO and another algorithm called Ridge Regression.

Suggested reading

- Chahuneau, Victor, Kevin Gimpel, Bryan R. Routledge, Lily Scherlis, and Noah A. Smith. "Word salad: Relating food prices and descriptions." In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 1357-1367. 2012.
  - A simple use of LASSO for understanding menu pricing at restaurants.

Relevant packages/libraries

- Python:
  - For econometrics in python, Statsmodels, linearmodels, and scikit-learn are useful.
  - For LASSO, use `sklearn.linear_model.Lasso` or `sklearn.linear_model.LassoCV` (for automated cross-validation).
  - For elastic net, use `sklearn.linear_model.ElasticNet` or `sklearn.linear_model.ElasticNetCV` (for automated cross-validation).
  - For out-of-sample time series prediction, Facebook's Prophet library is good to check out. It runs on Stan, which is perhaps the most well-known Bayesian inference library.
- R:
  - R has robust econometric tools built in, and many other available. For HDFE, lfe is efficient.
  - For LASSO and elastic net GLMnet is probably the best and is fairly easy to use (including a very efficient cross-validated method); caret also works.
  - Facebook's Prophet library is also available for R.

Extra reading (Optional)

- Tibshirani, Robert. "Regression shrinkage and selection via the lasso." Journal of the Royal Statistical Society: Series B (Methodological) 58, no. 1 (1996): 267-288.
  - Technical details for LASSO.
- Meinshausen, Nicolai. "Relaxed lasso." Computational Statistics & Data Analysis 52, no. 1 (2007): 374-393.
  - A bit of an improvement on LASSO called "Relaxed LASSO" (How to fit with R and glmnet).
  - There is some evidence that it is fairly optimal for both low and high signal to noise ratio problems:
    - Hastie, Trevor, Robert Tibshirani, and Ryan Tibshirani. "Best Subset, Forward Stepwise or Lasso? Analysis and Recommendations Based on Extensive Comparisons." Statistical Science 35, no. 4 (2020): 579-592.

Session 2: Machine-learning drop-ins and ensembling

Concepts

1. Support Vector Machines (SVM).
   - A simple method for classifying data in a supervised manner.
2. XGBoost (e**x**treme **g**radiant **boost**ing).
   - A more complex method using gradient boosting and decision trees.
3. Ensembling.
   - Combining multiple models to get a better model.
     - This is part of what XGBoost does under the hood.

Suggested reading

- Deryugina, Tatyana, Garth Heutel, Nolan H. Miller, David Molitor, and Julian Reif. "The mortality and medical costs of air pollution: Evidence from changes in wind direction." American Economic Review 109, no. 12 (2019): 4178-4219.
  - Applies XGBoost for classifying mortality as it relates to air pollution.

Relevant packages/libraries

- Python:
  - For SVM and ensembling, scikit-learn works well.
  - Python is the primary language for XGBoost: you can use the original xgboost package.
  - Other options for similar methods include LightGBM.
- R
  - For SVM you can use the venerable e1071 library or use kernlab.
  - For XGBoost, there is the xgboost library.
  - For other options, consider LightGBM.
  - For Ensembling, you can use SuperLearner, EnsembleML, or roll your own.

Extra reading (Optional)

- Purda, Lynnette, and David Skillicorn. "Accounting variables, deception, and a bag of words: Assessing the tools of fraud detection." Contemporary Accounting Research 32, no. 3 (2015): 1193-1223.
  - Classification of misreporting with SVM.
- Bao, Yang, Bin Ke, Bin Li, Y. Julia Yu, and Jie Zhang. "Detecting accounting fraud in publicly traded US firms using a machine learning approach." Journal of Accounting Research 58, no. 1 (2020): 199-235.
  - Applies RUSBoost to accounting data to predict accounting fraud.
- Qiu, Yue, Tian Xie, and Y. U. Jun. "Forecast combinations in machine learning." (2020)
  - A straightforward use of Ensemble methods in economics.
- Chen, Tianqi, and Carlos Guestrin. "Xgboost: A scalable tree boosting system." In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785-794. 2016.
  - The technical details behind XGBoost.
- Chernozhukov, Victor, Mert Demirer, Esther Duflo, and Ivan Fernandez-Val. Generic machine learning inference on heterogenous treatment effects in randomized experiments. No. w24678. National Bureau of Economic Research, 2018.
  - The technical details behind the CDDF usage of XGBoost in the Deryugina et al. 2019 paper.

Session 3: Working with text data

Concepts

1. Text in python.
   - Importing from files and working with it.
   - Simple counting methods (e.g., sentiment).
2. Simple pattern matching.
   - Regular expressions on text.
3. Grammar parsing.
   - Automating the process using spaCy or NLTK (the Natural Language Toolkit).
   - Named Entity Recognition (NER).
   - Grammar and parts of speech.
4. Working on data from web pages.
   - Loading in html data and processing it using Beautiful Soup.

Suggested reading: N/A

Relevant packages/libraries

- Python:
  - A lot of traditional data sources and analytics for NLP are included in NLTK.
  - A more modern approach (built on machine learning) is spaCy. Often times spaCy's approach is more accurate, but it doesn't have as many functions as NLTK.
  - Stanford NLP also offers a text parser; the core functions are available in python using Stanza. Some useful functions are only available via Java, however.
- R
  - For general text processing, tidytext is very helpful.
  - For regular expressions, consider using stringr instead of Base R.
  - For simple calculations (sentiment, readability, etc.), quanteda can handle these easily.
  - For grammar parsing, consider spacyr or coreNLP.

Extra reading (Optional)

- Loughran, T. and McDonald, B., 2011. When is a liability not a liability? Textual analysis, dictionaries, and 10-Ks.The Journal of Finance, 66(1), pp.35-65.
  - Perhaps the most influential sentiment paper in Finance and Accounting for the past decade. Where the Loughran McDonald dictionaries come from.
- Antweiler, Werner, and Murray Z. Frank. "Is all that talk just noise? The information content of internet stock message boards." The Journal of finance 59, no. 3 (2004): 1259-1294.
  - A seminal paper in textual analysis in Finance that uses Naive Bayes to classify sentiment.
- Loughran, Tim, and Bill McDonald. "Measuring firm complexity." Available at SSRN 3645372 (2020).
  - A more recent dictionary-oriented study by Loughran and McDonald.
- Loughran, Tim, and Bill McDonald. "Textual Analysis in Finance." Annual Review of Financial Economics 12 (2020): 357-375.
  - A nice and current review paper that takes a somewhat critical look at some of the textual analysis work that has been done in finance and accounting.
- Jurafsky, Dan, Victor Chahuneau, Bryan R. Routledge, and Noah A. Smith. "Narrative framing of consumer sentiment in online restaurant reviews." First Monday (2014).
  - Examines "narrative framing" in reviews of restaurants online, using a fairly simple approach.
- Hope, Ole-Kristian, Danqi Hu, and Hai Lu. "The benefits of specific risk-factor disclosures." Review of Accounting Studies 21, no. 4 (2016): 1005-1045.
  - Examines accounting disclosure specificity using NER.
- Gentzkow, Matthew, Bryan Kelly, and Matt Taddy. "Text as data." Journal of Economic Literature 57, no. 3 (2019): 535-74.
  - A summary of textual data for economic research.

Session 4: Algorithms on text

Concepts

1. Classifying text in aggregate: Supervised approach.
   - Classifying words based on sources of text (books, articles, etc.).
2. Breaking away from words.
   - Using embeddings to convert text into numeric data.
3. Classifying text in aggregate: Unsupervised approach.
   - Using Latent Dirichlet Allocation to quickly summarize what's in a document.
4. Dimensionality reduction.
   - t-SNE and UMAP – similar in spirit to PCA, but more robust.

Suggested reading

- Hassan, Tarek A., Stephan Hollander, Laurence Van Lent, and Ahmed Tahoun. "Firm-level political risk: Measurement and effects." The Quarterly Journal of Economics 134, no. 4 (2019): 2135-2202.
  – A supervised approach to classifying accounting text as being related to political risk.
- Huang, Allen H., Reuven Lehavy, Amy Y. Zang, and Rong Zheng. "Analyst information discovery and interpretation roles: A topic modeling approach." Management Science 64, no. 6 (2018): 2833-2855.
  – A nice conceptual application of LDA for unsupervised classification of text.

Relevant packages/libraries

- Python:
  – For LDA and word2vec, gensim is both easy to use and fast to run.
  – TensorFlow has embedding implementations available – harder to implement than those of gensim, but TensorFlow includes more modern approaches and should usually run faster.
- R
  – For word embeddings, you can use word2vec
  – stm (Structural Topic Models) is a nice twist on LDA. lda, topicmodels, and RMallet also work.
  – TensorFlow is also available in R.

Extra reading (Optional)

- Hassan, Tarek Alexander, Stephan Hollander, Laurence Van Lent, and Ahmed Tahoun. Firm-level exposure to epidemic diseases: COVID-19, SARS, and H1N1. No. w26971. National Bureau of Economic Research, 2020.
  – A very straight-forward application of textual data that is quite timely.
- Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. "Efficient estimation of word representations in vector space." arXiv preprint arXiv:1301.3781 (2013).
  – Technical details for word2vec.
- Blei, David M., Andrew Y. Ng, and Michael I. Jordan. "Latent Dirichlet allocation." the Journal of machine Learning research 3 (2003): 993-1022.
  – Technical details for LDA.
- Brown, Nerissa C., Richard M. Crowley, and W. Brooke Elliott. "What are you saying? Using topic to detect financial misreporting." Journal of Accounting Research 58, no. 1 (2020): 237-291.
  – Another nice use of LDA: predicting corporate misreporting based on annual report text.
- Roberts, M.E., Stewart, B.M., Tingley, D., Lucas, C., Leder-Luis, J., Gadarian, S.K., Albertson, B. and Rand, D.G., 2014. Structural topic models for open-ended survey responses. American Journal of Political Science, 58(4), pp.1064-1082.
  – A more extensible LDA model that allows one to impose a regression-like structure on them.
- De Choudhury, Munmun, Michael Gamon, Scott Counts, and Eric Horvitz. "Predicting depression via social media." In Proceedings of the International AAAI Conference on Web and Social Media, vol. 7, no. 1. 2013.
  – Depression prediction using social media.

Session 5: Economics approaches to machine learning

Concepts

1. Fairness and bias.
   - Using SHAP (**Sh**apley **a**dditive ex**p**lanations) to evaluate bias (e.g., gender bias).
2. Machine learning for causality.
   - Using double/debiased/Neyman ML for treatment effects.

Suggested reading

- Wich, Maximilian, Jan Bauer, and Georg Groh. "Impact of politically biased data on hate speech classification." In Proceedings of the Fourth Workshop on Online Abuse and Harms, pp. 54-64. 2020.
  - A paper using SHAP to understand an impact of political bias.
- Chernozhukov, Victor, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, and Whitney Newey. "Double/debiased/Neyman machine learning of treatment effects." American Economic Review 107, no. 5 (2017): 261-65.
  - An influential new method for using machine learning for causality.

Relevant packages/libraries

- Python:
  - For SHAP, slundberg/shap is good for both calculating and visualizing.
  - For causal machine learning, look into DoubleML and microsoft/EconML.
- R
  - For bias, ModelOriented/shapper, liuyanguu/SHAPforxgboost, bgreenwell/fastshap and ModelOriented/DALEX are all good choices.
  - For causal machine learning, look into MCKnaus/dmlmt and DoubleML/doubleml-for-r.

Extra reading (Optional)

- Lundberg, Scott, and Su-In Lee. "A unified approach to interpreting model predictions." arXiv preprint arXiv:1705.07874 (2017).
  - The technical details behind SHAP
- Shapley, Lloyd S. "A value for n-person games." Contributions to the Theory of Games 2, no. 28 (1953): 307-317.
  - The game theory foundation of the SHAP algorithm.
- Athey, Susan. "The impact of machine learning on economics." In The economics of artificial intelligence: An agenda, pp. 507-547. University of Chicago Press, 2018.
  - A forward looking review of ML in Economics.
- Gentzkow, Matthew, Jesse M. Shapiro, and Matt Taddy. "Measuring group differences in high-dimensional choices: method and application to congressional speech." Econometrica 87, no. 4 (2019): 1307-1340.
  - Another method focused on improving causality using machine learning.
- Chernozhukov, Victor, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. "Double/debiased machine learning for treatment and structural parameters." (2018): C1-C68.
  - The technical details behind the double/debiased/Neyman ML paper.
- Kleinberg, Jon, Jens Ludwig, Sendhil Mullainathan, and Ziad Obermeyer. "Prediction policy problems." American Economic Review 105, no. 5 (2015): 491-95.
  - An overview of when prediction is more important than causation in policy problems, along with an explanation and example of how to solve such problems.
- Athey, Susan, and Guido W. Imbens. "The state of applied econometrics: Causality and policy evaluation." Journal of Economic Perspectives 31, no. 2 (2017): 3-32.
  - A review paper with a focus on policy evaluation and a bit of last week's material.

Session 6: Neural networks

Concepts

1. Neural networks.
   - What are they?
   - What types exist?
   - Why use them, and what are the trade offs?
2. Implementing a neural network for supervised classification.
   - Implementing the network layer-by-layer in Keras.
3. Leveraging off-the-shelf neural networks.
   - Use world-class algorithms with minimal coding required.
   - Example: Text similarity with Universal Sentence Encoder.

Suggested reading: N/A

Relevant packages/libraries

- Python:
  - Platforms for neural networks: Keras, TensorFlow, Theano, PyTorch, PaddlePaddle (for Chinese language problems).
  - For visual problems: Caffe and scikit-image.
  - Pre-built algorithms available from: TensorFlow Hub, TensorFlow/modles, Hugging Face, PyTorch Hub, ONNX/models, and PaddleHub (for Chinese language problems; however, some models are outdated!).
- R
  - Platforms for neural networks: Keras, TensorFlow, and nnet.

Extra reading (Optional)

- Crowley, Richard M., Wenli Huang, and Hai Lu. "Executive Tweets." Rotman School of Management Working Paper (2020).
  - An application of USE to compare the meaning of tweets.
- Cer, Daniel, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant et al. "Universal sentence encoder." arXiv preprint arXiv:1803.11175 (2018).
  - Technical details for USE.
- Huang, Allen, Hui Wang, and Yi Yang. "The Informativeness of Text, the Deep Learning Approach." (2020).
  - Applies BERT with a bit of transfer learning.
- Liu, Liu, Daria Dzyabura, and Natalie Mizik. "Visual listening in: Extracting brand image portrayed on social media." Marketing Science 39, no. 4 (2020): 669-686.
  - Examines how consumers portray brands in images posted on Instagram versus how brands portray themselves on their own Instagram account.
- Zhang, Shunyuan, Dokyun DK Lee, Param Vir Singh, and Kannan Srinivasan. "How much is an image worth? Airbnb property demand estimation leveraging large scale image analytics." Airbnb Property Demand Estimation Leveraging Large Scale Image Analytics (May 25, 2017) (2017).
  - Examines why images taken by Airbnb's photographers lead to higher revenue for properties, pinning over half the effect to the quality of posted images.