What can economists learn from machine learning?

Kevin Michael Frick

University of Bologna

Machine learning is becoming increasingly popular in economics as it can be used as a tool to make accurate predictions of human behavior. This paper analyzes a select few notable examples, explaining the processes involved from a computer engineering point of view. An "operational definition of machine learning" is introduced first, then the position of machine learning as a subset of research on artificial intelligence is clarified. An overview of how machine learning can be used to process and interpret big data is then given and accompanied by introductory paragraphs detailing the inner workings of relevant modeling approaches.

This writing sample was originally authored in 2019 and is published, together with other works, in a book with the aim of contributing to the public understanding of science, written for a non-technical audience. It is assumed that readers have an undergraduate-level knowledge of calculus and linear algebra. The original citation is Frick, K. M. (2020). What can economists learn from machine learning? In Astrazioni stenografiche. Concetti chiave per vivere consapevolmente la nostra società. *Bononia University Press. ISBN 978-88-6923-676-1. Some citations have been updated, e.g. in the case of working papers that have since been published.*

1 Introduction to machine learning

Athey (2018) provides an "operational definition of machine learning", defining it as a "field that develops algorithms designed to be applied to datasets". This definition is useful as a starting point to explore what machine learning can be used for. Some of its main subfields are pattern recognition (e.g. classifying pictures), reinforcement learning (e.g. automatically developing a robot control system) and natural language processing (e.g. translating ambiguous and context-dependent sentences). Moreover, applications of machine learning to policy analysis problems (e.g. estimating the impact of a public transport cost cut on air pollution levels) are particularly interesting to economists (Kleinberg, Ludwig, Mullainathan, & Obermeyer, 2015).

Many machine learning algorithms are able to not only discover a wide array of nonlinear relationships in big data but to actively strive to reduce the complexity of the model they are generating in order to make it easier for a human being to understand these relationships. Machine learning models, however, have their limits too. As both statisticians and economists know, correlation does not imply causation. Causation, however, is what economists are most interested about: gauging policy effects or evaluating market fluctuations require causal inference. However, research on machine learning has, for the most part, focused on prediction on historic datasets (Mackenzie, 2015).

One of the applications of machine learning that is enjoying huge popularity today is using such algorithms as a tool for predicting human behavior when confronted with different situations. The reasons machine learning is so widely used are easy to infer: big samples of data are much easier to come by than big samples of people, as well as faster to study and free of consequences on real society. Moreover, machine learning allows researchers to experiment on a variety of samples from around the world since, thanks to the Open Data phenomenon, many public and private entities are releasing data on their operations that is constantly being updated and added upon. Kosinski, Stillwell, and Graepel (2013) used common statistical methods such as logistic regression to predict, with surprising accuracy (more than 85% in all cases), sexual orientation, ethnicity and position on the political spectrum of volunteers based on Facebook likes, demographic profiles and results of psychometric tests. These kind of data are, by default and except for psychometric tests, publicly available.

Machine learning models outperform traditional structural models and can *learn* some "irrational" human behavior, such as adversity to risk, ambiguity and incomplete information.

Machine learning models can also be used to compensate for the absence of a control group. Varian (2014) argues that predictions for "what would have happened without intervening" (e.g. changing a policy) made on the basis of an applied machine learning model can be even *better* than a control group because it can take into account spurious variables (e.g. the effects of differing weather between two cities on sales) that a control group cannot.

2 Lasso and ridge regression

Lasso and ridge regression are two subsets of regularized regression. In a model with P predictor variables whose associated weights are $\mathbf{b}$, regularized regression is carried out by imposing a penalty term of the form $(1-\alpha)\lambda\|\mathbf{b}\| + \alpha\lambda\|\mathbf{b}\|^2$ and then trying to minimize the sum of this term and the sum of squared residuals. This method is called *elastic net regression*, of which lasso (*least absolute shrinkage and selection operator*) regression is a special case where $\alpha = 0$ and ridge regression is a special case where $\alpha = 1$. These methods attempt to reduce the number of non-zero regression coefficients (Varian, 2014) in order to reduce complexity, leading to a model with less overfit (and thus better out-of-sample performance) that is more easily interpreted by a human.

There is also a numerical computing interpretation of regularization. A problem is said to be *ill-posed* if it requires the inversion of an "almost-singular" matrix, which has small eigenvalues. A "measure of singularity" is the *condition number*, which given a matrix A is the product $\|A^{-1}\|\|A\|$ for a self-consistent norm $\|\cdot\|$[1]. Therefore, if A has small eigenvalues, since those of A^{-1} are their reciprocals, they will be large, so the condition number will also be large and the problem will be ill-posed. In linear regression, if the data matrix X is such that the matrix X^TX has small eigenvalues, the calculation of the weight vector $\mathbf{b} = (X^TX)^{-1}X^T\mathbf{y}$ will be ill-posed. Adding regularization means having a different expression for the weight vector, that is $\mathbf{b} = (X^TX + \lambda I)^{-1}X^T\mathbf{y}$ in the case of ridge regression, which means that a value λ is added to each diagonal element of the matrix, making eigenvalues larger and lowering the condition number, making the problem less ill-posed.

The computational complexity of lasso and ridge regression is very low: given a dataset of size

[1] A self-consistent norm satisfies $\|A^{-1}\|\|A\| \geq \|A^{-1}A\|$

n and a model with m features, the computational complexity is $O(m^3 + m^2 n)$ [2] (Efron, Hastie, Johnstone, Tibshirani, & others, 2004) for both models. This means that regularized regression becomes slower and heavier very quickly when increasing the number of parameters of the model, but scales much better with data set size. This is definitely a plus for a model to be used in big data analysis. This also means that such a model can be used on lower-powered devices such as mobile phones, on smaller datasets and with fewer features, without draining the battery, thus making it feasible to implement machine learning inside consumer applications.

2.1 Learning risk- and ambiguity-averse behavior

Peysakhovich and Naecker (2017) published a comparison between traditional economic models in the domain of risk (where the outcome of an event is uncertain, but the observer has full information about its probability distribution) and in the domain of ambiguity (where the outcome of an event is uncertain and the observer can only estimate it based on given data). The paper builds its dataset using Amazon Mechanical Turk (MTurk), a crowdsourcing platform that allows researchers to hire human workers to perform certain tasks: for example, in this case, rating how likely they would be to play certain games of chance. MTurk is widely used in economic literature and there is substantial evidence that MTurk datasets are just as representative, if not more so (Paolacci & Chandler, 2014), than traditional datasets. Participants were given instructions about the experiment, which required them to enter a numerical value that represented their "willingness to play" a certain lottery; a lottery was defined as an urn containing some red, green and blue balls, totaling 100. Each color had an associated monetary prize. Participants were then asked to complete a comprehension quiz and data from those who answered incorrectly were not considered in the final sample. This led to a final sample size of 315 people. This sample was then split into a training set (70%) and a test set (30%). Splitting data into a training and a test set in order to counteract in-sample overfitting is standard practice in machine learning literature and will be mentioned several times throughout this paper.

The paper uses regularized regression methods as their machine learning algorithm, which are very basic by themselves, only being able to uncover linear relationships. In order to allow their models to discover non-linear patterns, they perform *basis expansion* on their predictors, that is they apply a family of transformations to the features in order for the fitted model to be a non-linear function.

The authors show that in the domain of risk machine learning models are able to rediscover the expected utility model with probability weighting (EUP) and perform just as well as the model itself in terms of average squared error, which was the metric used to gauge prediction accuracy throughout the paper. In the domain of ambiguity, however, regularized regression not only outperforms traditional economic models, but is also able to *learn* human ambiguity-aversion. The example proposed in the paper is that of two lotteries, with the same payoff, one having 50% odds of winning and the other having uniformly drawn odds in $[0, 1]$ with an average of 50%: humans prefer the first and machine learning is able to predict this behavior. This also highlights how machine learning is not only useful as a prediction tool but can also be used to advance and improve current economic models, for example in this case by taking into account ambiguity aversion.

Another point heavily driven home throughout the paper is the method used to evaluate machine

[2] A detailed explanation of big-O notation is outside the scope of this paper. It can be interpreted as an upper bound for the number of mathematical operations required to compute an algorithm.

learning algorithms for *out-of-sample* accuracy instead of *in-sample* fit: traditional statistical evaluation of models tends to err on the side of overfitting. However, training the machine learning model on only part of the available data allows for testing its accuracy on the remainder, thus heavily penalizing models that can have high in-sample accuracy but are terrible predictors when used on different data from the ones it was trained on - like real world data (Kohavi, 1995).

3 Decision trees

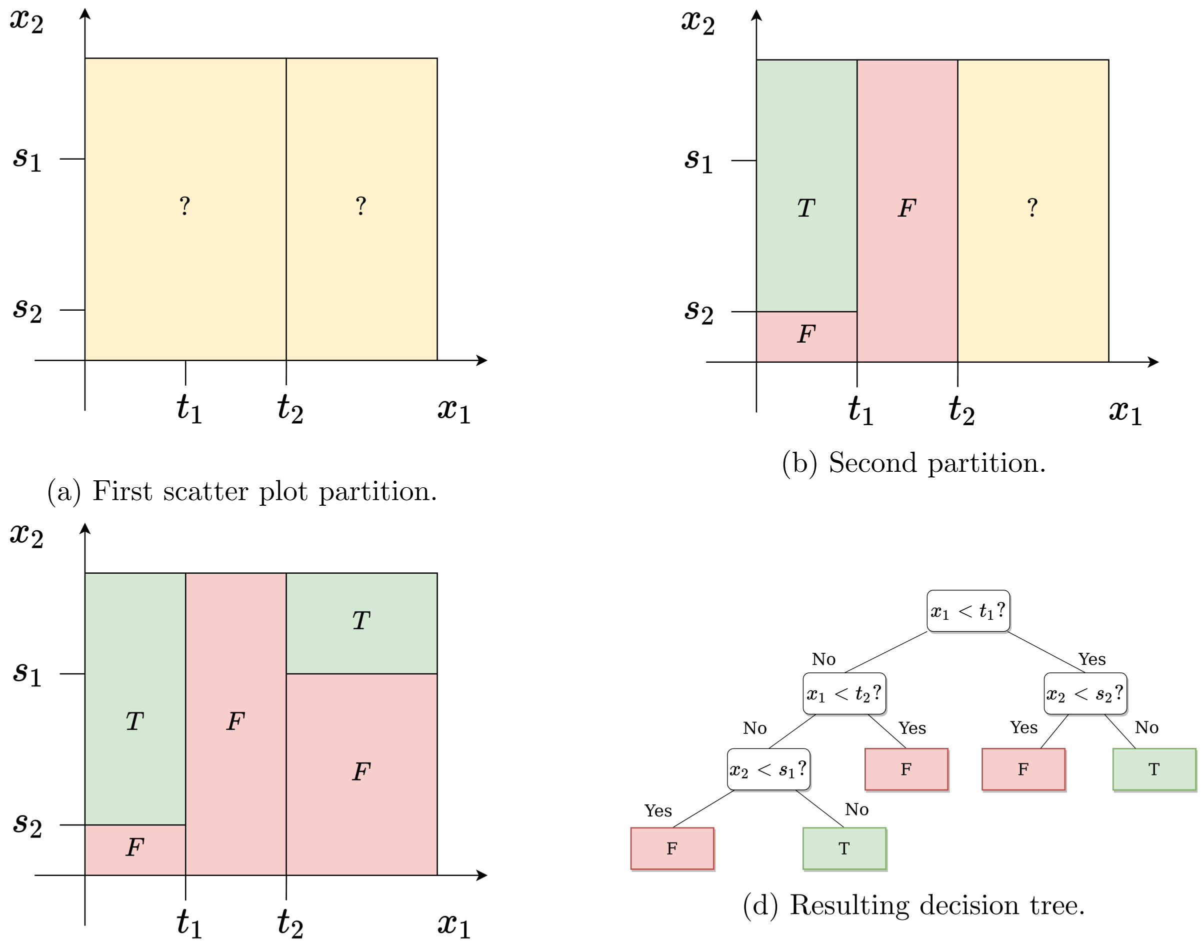
A problem which requires predicting a true/false outcome y based on a set of explanatory variables (called features in machine learning) is known as a binary classification problem. The goal of *decision tree learning* is to construct a tree that leads to a decision about how to classify the observation (Hastie, Tibshirani, & Friedman, 2013). A graphical explanation of how decision trees are built is best applied to the case of binary trees (i.e. every non-leaf node of the tree has two children) and two explanatory variables. Consider the dataset as a scatter plot of points in the variables x_1 and x_2 and a continuous response y. The first decision partitions the space into two regions A and B according to the mean value of y. The second decision partitions A into A1 and A2 and B into B1 and B2 according to the same rule. One or more of these regions are finally partitioned again and then some stopping rule terminates the algorithm. This results in the partition plots and decision tree depicted in our elaborations, fig. 1c fig. 1d.

The restriction to binary trees is usually preferred because of computational efficiency considerations and this case can, of course, be generalized to an arbitrary number of predictors. A decision tree is easily interpreted by a human and the most and least important factors in deciding whether to predict “true” or “false” can be discerned by simply reading which predictors are or are not used by the decision tree. Moreover, tree models are able to uncover more subtle relationships between data: a simple regression might find a linear or polynomial relationship between a variable and a feature, while a tree could show that only extremely high or low values of that feature alter the variable and that the feature has no influence when its value is in the middle of its range (Quinlan, 1990). Tree models, however, tend to overfit and grow too much trying to cover every variation on the training dataset. To counteract this tendency, trees are usually *pruned* by imposing a cost for complexity, which in turn is usually defined as the number of terminal nodes (leaves) (Kim & Koehler, 1995).

3.1 Initial play in matrix games

The field of game theory has a wide array of applications, from the military (Haywood, 1954) to sales and management (Saloner, 1991). One of the most common models in game theory is the normal form game, or matrix game. In a two-player matrix game, the row and the column player are both given the same matrix that outlines the expected payoffs (which can be negative) for both players’ choices. The two players then make their choices at the same time, or otherwise without knowing the other player’s choice before making theirs. If one player could predict the other player’s action with reasonably more certainty than random guessing, they could have a better chance at making the best choice they can to maximize their payoffs.

While standard game-theoretical models rely on the assumption of common knowledge of rationality (everyone is fully rational and everyone knows it), in the last decades behavioral economists have developed alternative models that relax this assumption for the sake of psychological realism



(a) First scatter plot partition.

(b) Second partition.

(c) Third partition.

(d) Resulting decision tree.

and predictive accuracy. Level-k play (Nagel, 1995) is one of these models. In level-k play, a player is said to be level-0 if they choose their behavior ignoring the information they have on the other players and act randomly. A level-1 player assumes the other players are level-0 and chooses the play that maximizes expected payoff under this hypothesis. Reasoning inductively, for any value of k, level-k players are those who assume their opponents are level-$(k-1)$ and act accordingly. A textbook example of level-k reasoning are the possible strategies for the Keynesian beauty contest, in which players are asked to choose a number and rewarded according to how close the number is to a certain fraction of the average of the other players' choices. If participants are asked to choose a number between 1 and 100, and rewarded if their number is close to half the average of the other players' choices, level-0 players will pick such a number at random from a uniform distribution; level-1 players will know that the average of a uniform distribution between 0 and 100 is 50 and will therefore pick 25 as their guess; level-2 players will assume everyone else is picking 25 and will therefore pick 13, and so on and so forth.

Level-1(α) play (Fudenberg & Liang, 2019) is a variation of level-1 play which accounts for human risk aversion by choosing the action that corresponds to level-1 play on a game whose payoffs are a real-valued root of those of the original game: for every payoff u, this kind of play considers a payoff $f(u) = u^{\alpha}, \alpha < 1$. When the game has actions whose payoffs are close to those of the level-1 action but have lower variation, this kind of play achieves better accuracy than standard level-1.

Fudenberg and Liang (2019) tried using machine learning algorithms to predict which of various strategy profiles best fits any given game, then using that model to predict the row player's initial play. The games participants were asked to play were generated in different ways: for the intial dataset, 86 games were selected from six game theory papers; afterwards, after noticing that level-1(α) play was a very good predictor of human behavior, achieving 89% accuracy, when ran on a different set of games with randomly-generated payoffs, the authors noted that it would be more efficient to focus on games where level-1(α) was *not* a good predictor. To this end, the paper details how these games were generated: a rule was first trained to predict how often level-1(α) play would be preferred by human players, then random games were generated and those where more than 50% of players would follow this model were discarded until a dataset of 200 games had been generated.

After generating these games, the authors used MTurk to assign 40 players to each game and analyzed the data, confirming their prediction that level-1(α) play would not do a good job of predicting initial play in such games.

On their first test run the authors find that a bagged decision tree is able to discover the same human ambiguity-aversion pattern covered in Peysakhovich and Naecker (2017) by predicting that a play with similar expected payoff but lower variation is preferred to one with higher variation, even if the first one is the level-1 action. The paper continues by using machine learning to generate games where level-1(α) is not accurate in order to uncover further hidden patterns.

Training a decision tree on a dataset with randomly generated games leads to a tree that splits games into four classes. The first two classes always predict the Nash equilibrium as initial play; the latter two predict adherence to level-1(α) play. This tree results in 79% accuracy and 69% completeness, which is better than either always predicting either Nash or level-1(α).

4 Bagged trees, random forests and gradient boosting

A very popular technique for improving the quality of machine learning predictions is using *model ensembles* (Karpathy, 2019) that is averaging the inferences from multiple models which can even be trained on different datasets. Some common ensembles of decision tree models are bagged trees, random forests and boosted trees.

As mentioned, decision trees can grow too much and overfit the dataset if they are not effectively pruned. Aside from pruning, another technique that helps deal with overfitting is bagging. This technique has its roots in the statistical practice known as bootstrapping (Breiman, 1996), that is drawing with replacement multiple samples from a dataset, which might be even bigger than the dataset itself. A decision tree is then fitted on each sample and the resulting predictions are finally averaged or majority-voted.

Random forest building (Biau & Scornet, 2016) is a more complex machine learning algorithm which uses multiple, unpruned trees that are restricted to use a randomly chosen subset of the full set of predictors. To classify an observation, a majority vote is used in the case of classification and an average in the case of regression. Random forests, unlike trees, are harder to interpret as the number of trees is usually in the thousands. They can, however, still offer useful insight about which variables contribute the most to prediction accuracy. Random forests can be seen as a special case of a bagged tree in which each subtree is restricted to only use a subset of features for its classification or regression.

Gradient boosting (Hastie et al., 2013) is a third, fundamentally different method of dealing with overfitting trees using an ensemble of "weak decision trees" trained on altered datasets that give greater weight to data points that have been misclassified by other "weak" trees. Gradient boosting starts from a very inaccurate model (e.g. as inaccurate as $f_0(\mathbf{x}) = \bar{y}$, that is always predicting the mean value of the dependent variable) and fits a decision tree on the error function $\mathcal{L}(\mathbf{y}, f_0(\mathbf{x}))$. The resulting piecewise function $h_k(\mathbf{x}, \mathcal{L})$ is added to the previous "weak" estimator, so the recursive formula becomes $f_{k+1}(\mathbf{x}) = f_k(\mathbf{x}) + h_k(\mathbf{x}, \mathcal{L}(\mathbf{x}, f_k(\mathbf{x}))$. The final output is, like in random forests and bagged trees, a majority vote or an average between all outputs.

4.1 The impact of minimum wage on salaries

There is a vast literature on the effects of minimum wage laws but no ultimate consensus on its effects, both on employment and actual quality of life of the affected workers. In fact, in 2017 two studies analyzed Seattle's minimum wage laws and reached polar opposite conclusions: University of Washington's found it was "costing jobs" while UC Berkeley's concluded it "hasn't cut jobs" (Gill, 2018). Therefore, even leaving ethical concerns and arguments aside, it is difficult for policymakers to decide whether to implement, raise or lower minimum wages. Most American studies on the effects of minimum wage laws have historically focused on what happened to minimum wage *jobs*. For example, more than half of the people who work for minimum wage or less have jobs in the food preparation industry (Gill, 2018). This, however, tends to shift the focus from the effects on people to the effect on jobs. It is tautological that minimum wage jobs are affected from minimum wage laws, but these jobs may be held by different individuals before and after the implementation of the policy.

A better question might be "which demographics are more likely to work for minimum wage?" which is the question that Cengiz, Dube, Lindner, and Zentler-Munro (2021) tries to answer. The paper uses machine learning tools to build a prediction model able to build samples containing the same number of minimum wage workers as commonly used samples, but less non-minimum wage workers. The author uses data from the 1996-2017 CPS-Outgoing Rotation Group (CPS-ORG). The predictors used in training decision trees are age, education, citizenship, gender, population density of area of residency, marital status, ethnicity, and whether or not an individual has served in the US military forces. The whole dataset is divided into three samples: training, test and left-out. The left-out sample, composed of all remaining observations, is not used in the paper but only for unrelated estimations.

In the training sample, the author excludes those who have jobs where receiving tips is the norm as well as those who are from states that received a minimum wage increase in the last year (in order to only test on environment where the job market has adjusted). The test sample is composed of the whole set of observations in states that *did* receive a minimum wage increase in the *subsequent* year and that are not present in the training sample. This is a form of "data set manufacturing" that is usually actively avoided in machine learning literature. However, in this case the model performing well on markets that did not receive minimum wage hikes means that it is not taking into account these hikes and therefore it should not matter whether these are or are not present in the test set; if they are not, and the model fits well in the test set, predictions can be regarded as being more robust if anything. Moreover, the author notes that a different composition of the data sets, using ten-fold cross-validation, leads to very similar conclusions.

The data are divided into mutually exclusive subgroups and sorted by the smallest probability a member of each subgroup has to be a minimum-wage worker. Three bigger groups are formed from these subgroups: the high-impact group, the baseline group and the no-impact group. Wage and employment effects of the minimum wage are then analyzed on these three groups, discovering that both the high-impact and the baseline groups report a large and statistically significant effect on wages but no effects on employment.

One interesting finding is that there is a negative correlation between the probability of being a minimum wage worker and both the percent of women and the percent of people of "non-Caucasian ethnicity" in the considered subgroup.

4.2 Bail decisions

Both computer scientists and economists are used to "abstracting away" the complexity of reality by devising and studying mathematical models that allow for isolating variables of interest. Models of human behavior, however, are not some kind of abstraction that crumbles when taken out of a lab; on the contrary, they have immediate applications to issues of interest to policymakers (Kleinberg et al., 2015).

For example, every day, judges all over the world are requested to make what amounts to a very specific and very important prediction of human behavior: should they release the defendants that stand before them, will those defendants flee the country, commit other crimes, or just go on with their lives and wait for their trial, appearing in court when called upon? This is not a matter of crime and punishment; in fact, the trial has not been carried out yet and police investigations might not even be over. The judge only knows the defendant's criminal history and some personal details, so any kind of correlation is not immediately obvious and, more importantly, there is no need to establish a causal relationship - since there either is none, or it is obvious - but only to predict somebody's behavior and act accordingly.

Kleinberg, Lakkaraju, Leskovec, Ludwig, and Mullainathan (2017) tackled precisely this case. The authors obtained a dataset of 758027 arrests made in New York City between 2008 and 2013 for which judges had to decide whether to release or jail while waiting for the trial. They then used gradient-boosted decision trees to predict how high the risk is that the defendant will fail to appear in court (FTA) and analyze the results in order to devise a rule that can help judges reduce FTA rates making use of their algorithm. Decision trees' hyperparameters such as tree depth are selected using five-fold cross-validation. k-fold cross validation is another common practice in machine learning: the dataset is divided into k subsets, known as "folds", then parameters (or, in this case, hyperparameters) are estimated using $k-1$ folds as training data and the remaining fold as test data. The process is repeated k times and results are then averaged.

The authors' decision trees are then trained only using data that is strictly related to the case, such as criminal records and previous FTAs, and the only unrelated feature is age at the time of arrest. Two glaring issues are evident from these two statements: first, we can only know FTAs from released defendants, but we cannot know what a jailed defendant would have done if released; second, the devised models only use administrative data but the judge does not, which might lead to training - and therefore prediction - issues. The authors use gang tattoos as an example: if many people who have gang tattoos are young, and judges tend to always jail people with gang tattoos as they are considered high-risk, depending on how the loss function is constructed decision trees might erroneously decide to jail all young people or, conversely, to release them even though a judge would jail them, because they do not know about gang tattoos.

These two issues are related since, if we assume that judges take into account variables that are unobserved by the algorithm, we cannot assume FTA rates of released defendants to be indicative of anything concerning jailed defendants with similar observed predictor variables. This is known as the *selective labels problem* and is tackled by exploiting its one-sidedness: it is easy to know what would have happened if a released defendant had been jailed. Since what the authors are interested in is *influencing a decision*, one way judges could be instructed to use their algorithm is to jail those that they would have released but are predicted to have a high risk of FTA: the data show that high-risk defendants are released much more often that they would if judges acted "rationally" and jailed everybody with a high predicted FTA risk. These people could in principle be low-risk, and the judges might realize this and release them, but relating observed FTA rates to predicted risk shows that defendants that are predicted to be high-risk do have a high observed FTA rate.

This kind of analysis does not take into account the cost implied by jailing defendants and only influences decisions on the basis of FTA risk. However, trying to draw a correlation between judge jailing rate and defendant characteristics does not lead to any kind of consistent finding (in statistical terms, drawing a histogram of p-values of F-test statistics does not show any unusual mass at low p-vaues). Therefore, if the average judge were to use the algorithm this way, it would be possible to maintain the same FTA rate jailing only 48.2% as many people, or get FTA reductions that are 75.8% larger by jailing more people. By assuming unobservable variables have no effect, the algorithm could reduce FTAs by 24.7% without incrementing jailing rate, or reduce the detention rate by 41.9% without incrementing FTA rate.

These results seem to suggest that judges are making mistakes in their predictions, or decisions. Therefore, the authors then move on to try and explain what is causing these mispredictions. The first interesting finding is that judges with higher jailing rate are detaining more low-risk people, so a judge being "stricter" does not necessarily mean they are "better". Continuing along this line, the authors notice that judges struggle much more with high-risk cases, treating them as if they were low-risk and heavily weighing their decision based on the current offense which caused the defendant to be arrested. One possible explanation for both issues is that judges select on variables that are not observed by the algorithm, and righfully so: drawing from behavioral science literature, the author hypothesize that human judges overweigh interpersonal information, such as the degree of eye contact being made. Still, the authors do not manage to fully explain the source of judicial error: unobservable variables can only explain about a quarter of the difference between the judges' release decisions and the arguably better ones made by the algorithm.

5 Conclusions

Machine learning is not simply a faster way to analyze data or an evolution in spreadsheet technology. The fact that machine learning algorithms are able to re-discover traditional economic models means that knowledge of these algorithms is now as valuable as the models themselves in an economist's set of tools. Beyond that, machine learning can help economists combine existing models or develop new ones in order to be able to both make better predictions and understand why those predictions work.

Moreover, machine learning is not only useful as a model but also as a way to identify causal relations, which play a fundamental role in policy applications, with more certainty thanks to samples that are more representative or novel ways to set up a study. This does not mean that the field of economics will be absorbed into that of machine learning but rather that any new research should take into account the possibilities offered by these tools which can help economists get their results not only faster but also more reliably.

References

Athey, S. (2018). The impact of machine learning on economics. In *The economics of artificial intelligence: An agenda.* University of Chicago Press.

Biau, G., & Scornet, E. (2016). A random forest guided tour. *Test*, *25*(2), 197–227.

Breiman, L. (1996). Bagging predictors. *Machine learning*, *24*(2), 123–140.

Cengiz, D., Dube, A., Lindner, A., & Zentler-Munro, D. (2021, January). *Seeing Beyond the Trees: Using Machine Learning to Estimate the Impact of Minimum Wages on Labor Market Outcomes* (Tech. Rep. No. w28399). Cambridge, MA: National Bureau of Economic Research. Retrieved from `http://www.nber.org/papers/w28399.pdf` doi: 10.3386/w28399

Efron, B., Hastie, T., Johnstone, I., Tibshirani, R., & others. (2004). Least angle regression. *The Annals of statistics*, *32*(2), 407–499. (Publisher: Institute of Mathematical Statistics)

Fudenberg, D., & Liang, A. (2019). Predicting and understanding initial play. (Publisher: PIER Working Paper)

Gill, D. (2018). Through the Minimum Wage Looking Glass: Economic Consensus Unrealized. *UCLA Anderson Review.* Retrieved from `https://anderson-review.ucla.edu/minimum-wage-primer-leamer/`

Hastie, T., Tibshirani, R., & Friedman, J. (2013). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* Springer Science & Business Media. (Google-Books-ID: yPfZBwAAQBAJ)

Haywood, O. G. (1954, November). Military Decision and Game Theory. *Journal of the Operations Research Society of America*, *2*(4), 365–385. Retrieved from `https://pubsonline.informs.org/doi/abs/10.1287/opre.2.4.365` (Publisher: INFORMS) doi: 10.1287/opre.2.4.365

Karpathy, A. (2019, April). *A Recipe for Training Neural Networks.* Retrieved from `http://karpathy.github.io/2019/04/25/recipe/`

Kim, H., & Koehler, G. J. (1995, December). Theory and practice of decision tree induction. *Omega*, *23*(6), 637–652. Retrieved from `https://www.sciencedirect.com/science/article/pii/0305048395000364` doi: 10.1016/0305-0483(95)00036-4

Kleinberg, J., Lakkaraju, H., Leskovec, J., Ludwig, J., & Mullainathan, S. (2017). Human Decisions and Machine Predictions. *The Quarterly Journal of Economics*, *133*(1), 237–293. Retrieved from `https://doi.org/10.1093/qje/qjx032` (_eprint: https://academic.oup.com/qje/article-pdf/133/1/237/30636517/qjx032.pdf) doi: 10.1093/qje/qjx032

Kleinberg, J., Ludwig, J., Mullainathan, S., & Obermeyer, Z. (2015). Prediction policy problems. *American Economic Review*, *105*(5), 491–95.

Kohavi, R. (1995). A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. In (pp. 1137–1143). Morgan Kaufmann.

Kosinski, M., Stillwell, D., & Graepel, T. (2013). Private traits and attributes are predictable from digital records of human behavior. *Proceedings of the National Academy of Sciences*, *110*(15), 5802–5805. Retrieved from `https://www.pnas.org/content/110/15/5802` (Publisher: National Academy of Sciences _eprint: https://www.pnas.org/content/110/15/5802.full.pdf) doi: 10.1073/pnas.1218772110

Mackenzie, A. (2015, August). The production of prediction: What does machine learning want? *European Journal of Cultural Studies*, *18*(4-5), 429–445. Retrieved from `https://doi.org/10.1177/1367549415577384` (Publisher: SAGE Publications Ltd) doi: 10.1177/1367549415577384

Nagel, R. (1995). Unraveling in guessing games: An experimental study. *The American Economic Review*, *85*(5), 1313–1326. (Publisher: JSTOR)

Paolacci, G., & Chandler, J. (2014). Inside the Turk: Understanding Mechanical Turk as a participant pool. *Current Directions in Psychological Science*, *23*(3), 184–188. (Publisher: Sage Publications Sage CA: Los Angeles, CA)

Peysakhovich, A., & Naecker, J. (2017). Using methods from machine learning to evaluate behavioral models of choice under risk and ambiguity. *Journal of Economic Behavior & Organization*, *133*, 373–384. (Publisher: Elsevier)

Quinlan, J. (1990, March). Decision trees and decision-making. *IEEE Transactions on Systems, Man, and Cybernetics*, *20*(2), 339–346. (Conference Name: IEEE Transactions on Systems, Man, and Cybernetics) doi: 10.1109/21.52545

Saloner, G. (1991). Modeling, game theory, and strategic management. *Strategic Management Journal*, *12*(S2), 119–136. Retrieved from `https://onlinelibrary.wiley.com/doi/abs/10.1002/smj.4250121009` (_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/smj.4250121009) doi: 10.1002/smj.4250121009

Varian, H. R. (2014). Big data: New tricks for econometrics. *Journal of Economic Perspectives*, *28*(2), 3–28.