Holmes in *The Path of the Law* asked “What constitutes the law?” and answered that law is nothing more than prophecies of what the courts will do. As we discussed in Chapter 5, this is not just the trivial observation that one of the jobs of a lawyer is to predict the outcome of a client’s case: it is the insight that growth and development of the law itself—the path of the law—is constituted through predictive acts.

Holmes was preoccupied throughout his legal career with understanding the law as an evolving system. Kellogg, in a recent study of the roots of Holmes’s thinking, traces this interest to the period 1866–1870, Holmes’s first years as a practicing lawyer, and to his reading of John Stewart Mill on the philosophy of induction, and of William Whewell and John Herschel on the role of induction in scientific theory-building. Holmes’s original insight was that the development of law is a process of social induction: it is not simply logical deduction from axioms laid down in statutes and doctrine as formalists would have it; it is not simply the totality of what judges have done as the realists would have it. Instead, law develops through agents embedded in society who take actions that depend on and contribute to the accumulating body of experience, and it involves social agents who through debate are able to converge toward entrenched legal doctrine.

In the standard paradigm for machine learning, there is no counterpart to the first part of Holmes’s insight of social induction—i.e., to the role of active agents embedded in society. The standard paradigm is that there is something for the machine to learn, and this “something” is data, i.e.
given, and data does not accumulate through ongoing actions. This is why
the field is called “machine learning” rather than “machine doing”! Even
systems in which a learning agent’s actions affect its surroundings, for
example a self-driving car whose movements will make other road-users
react, the premise is that there are learnable patterns about how others
behave, and learning those patterns is the goal of training, and training
should happen in the factory rather than on the street.

There is however a subfield of machine learning, called reinforcement
learning, in which the active accumulation of data plays a major role.
“AlphaGo,” the AI created by DeepMind which in 2016 won a historic
victory against top-ranking (human) Go player Lee Seedol, is a product of
reinforcement learning. In this chapter we will describe the links between
reinforcement learning and Holmes’s insight that law develops through
the actions of agents embedded in society.

The second part of Holmes’s insight concerns the process whereby
data turns into doctrine, the “continuum of inquiry.” As case law accu-
mulates, there emerge clusters of similar cases, and legal scholars, exam-
ining these clusters, hypothesize general principles. Holmes famously
said that “general propositions do not decide concrete cases,” but he also saw
law as the repository of the “ideals of society [that] have been strong
enough to reach that final form of expression.” In other words, legal
document is like an accepted scientific theory: it provides a coherent nar-
rative, and its authority comes not from prescriptive axioms but rather
from its ability to explain empirical data. Well-settled legal doctrine arises
through a social process: it “embodies the work of many minds, and has
been tested in form as well as substance by trained critics whose practical
interest is to resist it at every step.”

There is nothing in machine learning that corresponds to this sec-
ond aspect of Holmes’s social induction, to the social dialectic whereby
explanations are generated and contested and some explanation eventu-
ally becomes entrenched. In the last part of this chapter we will discuss
the role of legal explanation, and outline some problems with explainabil-
ity in machine learning, and suggest how machine learning might learn
from Holmes.

9.1 Accumulating Experience

According to Holmes, “The growth of the law is very apt to take place in
this way: two widely different cases suggest a general distinction, which
is a clear one when stated broadly. But as new cases cluster around the
opposite poles, and begin to approach each other [...] at last a mathematical line is arrived at by the contact of contrary decisions.\(^6\)

Holmes’s metaphor, of a mathematical line drawn between cases with contrary decisions, will be very familiar to students of machine learning, since almost any introductory textbook describes machine-learning classification using illustrations such as the figure above. In the figure, each datapoint is assigned a mark according to its ground-truth label,\(^7\) and the goal of training a classifier is to discover a dividing line. DeepMind’s AlphaGo can be seen as a classifier: it is a system for classifying game-board states according to which move will give the player the highest chance of winning. During training, the system is shown many game-board states, each annotated according to which player eventually wins the game, and the goal of training is to learn dividing lines.

Holmes was interested not just in the dividing lines but in the accumulation of new cases. Some new cases are just replays with variations in the facts, Cain killing Abel again and again through history. But Holmes had in mind new cases arising from novel situations, where legal doctrine has not yet drawn a clear line. The law grows through a succession of particular legal disputes, and in no situation would there be a meaningful legal dispute if the dividing line were clear. Actors in the legal system adapt their actions based on the body of legal decisions that has accumulated,
and this adaptation thus affects which new disputes arise. New disputes will continue to arise to fill out the space of possible cases, until eventually it becomes possible to draw a line “at the contact of contrary decisions.” Kellogg summarizes Holmes’s thinking thus: “he reconceived logical induction as a social process, a form of inference that engages adaptive action and implies social transformation.”

Machine learning also has an equivalent of adaptive action. The training dataset for AlphaGo was not given a priori: it was generated during training, by the machine playing against itself. To be precise, AlphaGo was trained in three phases. The first phase was traditional machine learning, from an a priori dataset of 29.4 million positions from 160,000 games played by human professionals. In the second phase, the machine was refined by playing against an accumulating library of earlier iterations of itself, each play adding a new iteration to the library. The final iteration of the second-phase machine was played against itself to create a new dataset of 30 million matches, and in the third phase this dataset was used as training data for a classifier (that is to say, the machine in the third phase trains on a given dataset, which, like the given dataset in the first phase, is not augmented during training). The trained classifier was the basis for the final AlphaGo system. DeepMind later created an improved version, AlphaGo Zero, which essentially only needed the second phase of training, and which outperformed AlphaGo. The key feature of reinforcement learning, seen in both versions, is that the machine is made to take actions during training, based on what it has learnt so far, and the outcomes of these actions are used to train it further—Kellogg’s “adaptive action.”

Holmes says that the mathematical line is arrived at “by the contact of contrary decisions.” Similarly, AlphaGo needed to be shown sufficient diversity of game-board states to fill out the map, so that it can learn to classify any state that it might plausibly come across during play. In law the new cases arise through fractiousness and conflict—“man’s destiny is to fight” —whereas for AlphaGo the map was filled out by artificially adding noise to the game-play dataset.

Holmes has been criticized for putting forwards a value-free model of the law—he famously defined truth “as the majority vote of that nation that can lick all the others.” Kellogg absolves Holmes of this charge: he argues that Holmes saw law as a process of social inquiry, using the mechanism of legal disputes to figure out how society works, similar to how science uses experiments to figure out how nature works. The dividing
lines that the law draws are therefore not arbitrary: “Any successful conclusions of social inquiry must, in an important respect, conform with the world at large. Social inductivism does not imply that the procedures and ends of justification are relativist products of differing conventions.”

Likewise, even though the training of AlphaGo is superficially relativist (it was trained to classify game-board states by the best next move, assuming that its opponent is AlphaGo), it is nonetheless validated by objective game mechanics: pitted against Lee Seedol, one of the top human Go players in the world, AlphaGo won.

9.2 Legal Explanations, Decisions, and Predictions

“It is the merit of the common law,” Holmes wrote, “that it decides the case first and determines the principle afterwards.” Machine learning has excelled (and outdone the ingenuity of human engineers) at making decisions, once decision-making is recast as a prediction problem as described in Chapter 5. This success, however, has come at the expense of explainability. Can we learn how to explain machine learning decisions, by studying how common law is able to determine the principle behind a legal decision?

In the law, there is a surfeit of explanation. Holmes disentangled three types: (i) the realist explanation of why a judge came to a particular decision, e.g. because of an inarticulate major premise, (ii) the formalist explanation that the judge articulates in the decision, and (iii) explanation in terms of principles. Once principles are entrenched then the three types of explanation will tend to coincide, but in the early stages of the law they often do not. Principles reflect settled legal doctrine that “embodies the work of many minds and has been tested in form as well as substance by trained critics whose practical interest is to resist it at every step.” They arise through a process of social induction, driven forwards not just by new cases (data) but also by contested explanations.

To understand where principles come from, we therefore turn to judicial decisions. (In legal terminology, decision is used loosely to refer both to the judgement and to the judge’s explanation of the judgement.)

Here is a simple thought experiment. Consider two judges A and B. Judge A writes decisions that are models of clear legal reasoning. She takes tangled cases, cases so thorny that hardly any lawyer can predict the outcome, and she is so wise and articulate that her judgments become widely
relied upon by other judges. Judge B on the other hand writes garbled decisions. Eventually a canny lawyer realizes that this judge finds in favor of the defendant after lunch, and in favor of the plaintiff at other times of day (her full stomach is the inarticulate major premise). Judge B is very predictable, but her judgments are rarely cited and often overturned on appeal.

If we think of law purely as a task of predicting the outcome of the next case, then judgments by A and by B are equivalent: they are grist for the learning mill, data to be mined. For this task, the quality of their reasoning is irrelevant. It is only when we look at the development of the legal system that reasoning becomes significant. Judge A has more impact on future cases, because of her clear explanations. “[T]he epoch-making ideas,” Holmes wrote, “have come not from the poets but from the philosophers, the jurists, the mathematicians, the physicists, the doctors—from the men who explain, not from the men who feel.”

Our simple thought experiment might seem to suggest that it is reasoning, not prediction, that matters for the growth of the law. What then of Holmes’s famous aphorism, that prophecy is what constitutes the law? Alex Kozinski, a U.S. Court of Appeals judge who thought the whole idea of inarticulate major premise was overblown, described how judges write their decisions in anticipation of review:

If you’re a district judge, your decisions are subject to review by three judges of the court of appeals. If you are a circuit judge, you have to persuade at least one other colleague, preferably two, to join your opinion. Even then, litigants petition for rehearing and en banc review with annoying regularity. Your shortcuts, errors and oversights are mercilessly paraded before the entire court and, often enough, someone will call for an en banc vote. If you survive that, judges who strongly disagree with your approach will file a dissent from the denial of en banc rehearing. If powerful enough, or if joined by enough judges, it will make your opinion subject to close scrutiny by the Supreme Court, vastly increasing the chances that certiorari will be granted. Even Supreme Court Justices are subject to the constraints of colleagues and the judgments of a later Court.

Thus judges, when they come to write a decision, are predicting how future judges (and academics, and agents of public power, and public opinion) will respond to their decisions. Kozinski thus brings us back to prophecy and demonstrates the link with explanations “tested in form as well as substance by trained critics.”
9.3 Gödel, Turing, and Holmes

We have argued that the decision given by a judge is written in anticipation of how it will be read and acted upon by future judges. The better the judge’s ability to predict, the more likely it is that this explanation will become part of settled legal doctrine. Thus judges play a double role in the growth of the law: they are actors who make predictions; and they are objects of prediction by other judges.

There is nothing in machine learning that is analogous, no system in which the machine is a predictor that anticipates future predictors. This self-referential property does however have an interesting link to classic algorithmic computer science. Alan Turing is well known in popular culture for his test for artificial intelligence. Among computer scientists he is better known for inventing the Turing Machine, an abstract mathematical model of a computer that can be used to reason about the nature and limits of computation. He used this model to prove in 1936 that there is a task that is impossible to solve on any computer: the task of deciding whether a given algorithm will eventually terminate or whether it will get stuck in an infinite loop. This task is called the “Halting Problem.” A key step in Turing’s proof was to take an algorithm, i.e. a set of instructions that tell a computer what to do, and represent it as a string of symbols that can be treated as data and fed as input into another algorithm. Turing here was drawing on the work of Kurt Friedrich Gödel, who in 1930 developed the equivalent tool for reasoning about statements in mathematical logic. In this way, Gödel and later Turing were able to prove fundamental results about the limits of logic and of algorithms. They analyzed mathematics and computation as self-referential systems.

In Turing’s work, an algorithm is seen as a set of instructions for processing data, and, simultaneously, as data which can itself be processed. Likewise, in the law, the judge is an agent who makes predictions, and, simultaneously, an object for prediction. Through these predictions, settled legal principles emerge; in this sense the law can be said to be constituted by prediction. Machine learning is also built upon prediction—but machine learning is not constituted by prediction in the way that law is. We might say that law is post-Turing while machine learning is still pre-Turing.
9.4 What Machine Learning Can Learn from Holmes and Turing

Our point in discussing legal explanation and self-referential systems is this:

(i) social induction in the law is able to produce settled legal principles, i.e. generally accepted explanations of judicial decision-making;
(ii) the engine for social induction in the law is prediction in a self-referential system;
(iii) machine learning has excelled (and outdone human engineering ingenuity) at predictive tasks for which there is an empirical measure of success;
(iv) if we can combine self-reference with a quantitative predictive task, we might get explainable machine learning decisions.

In the legal system, the quality of a decision can be evaluated by measuring how much it is relied on in future cases, and this quality is intrinsically linked to explanations. Explanations are evaluated not by “are you happy with what you’ve been told?”, but by empirical consequences. Perhaps this idea can be transposed to machine learning, in particular to reinforcement learning problems, to provide a metric for the quality of a prediction. This would give an empirical measure of success, so that the tools that power machine learning can be unleashed, and “explainability” will become a technical challenge rather than a vague and disputed laundry list. Perhaps, as in law, the highest quality machine learning systems will be those that can internalize the behavior of other machines. Machines that do that would all the more trace a path like that of Holmes’s law.

These are speculative directions for future machine learning research, which may or may not bear fruit. Nonetheless, it is fascinating that Holmes’s understanding of the law suggests such avenues for research in machine learning.

Notes

1. Kellogg (2018) 29.
2. Silver et al. (2016).
3. Kellogg (2018) 8.
4. Kellogg draws out Holmes’s formative exposure to philosophers of science and his program to find a legal analogy for scientific hypothesis formulation. Id. 25, 51.
5. Holmes, *Codes, and the Arrangement of the Law*, American Law Review 5 (October 1870): 1, reprinted in Kellogg (1984) 77; CW 1:212. See Kellogg (2018) 37.

6. Holmes, *The Theory of Torts*, American Law Review 7 (July 1873): 652, 654. See Kellogg (2018) 41.

7. See Chapter 3, p. 37.

8. Kellogg (2018) 17.

9. Silver et al. (2017).

10. Holmes to Learned Hand, June 24, 1918, quoted in Kellogg (2018) 186–87.

11. Id.

12. Kellogg (2018) 180.

13. *Codes, and the Arrangement of the Law*, American Law Review 5 (October 1870): 1, reprinted in Kellogg (1984) 77; CW 1:212. See Kellogg (2018) 37.

14. In Chapter 6 we drew attention to a similar confounding, in Hempel’s failure to distinguish between explanation and prediction. See p. 74.

15. Remarks at a Tavern Club Dinner (on Dr. S. Weir Mitchell) (March 4, 1900) reprinted in De Wolfe Howe (ed.) (1962) 120. Poetry was one of Holmes’s polymath father’s several avocations.

16. Kozinski, *What I Ate for Breakfast and Other Mysteries of Judicial Decision Making*, 26 Loy. L.A. L. Rev. 993 (1993). Kozinski (1950–) served on the U.S. Court of Appeals for the Ninth Circuit from 1985 to 2017.

17. See *The Imitation Game* (U.S. release date: Dec. 25, 2014), in which actor Benedict Cumberbatch plays Turing. For the test, see Alan M. Turing, *Computing Machinery and Intelligence*, 59 Mind 433–60 (1950) (and esp. the statement of the game at 433–34). Cf. Halpern, *The Trouble with the Turing Test*, The New Atlantis (Winter 2006): https://www.thenewatlantis.com/publications/the-trouble-with-the-turing-test.

18. Alan Turing, *On Computable Numbers, with an Application to the Entscheidungsproblem*, 2(42) LMS Proc. (1936). For a readable outline of Turing’s contribution and its historical context, see Neil Immerman, *Computability and Complexity*, in Zalta (ed.), The Stanford Encyclopedia of Philosophy (Winter 2018 edition): https://plato.stanford.edu/archives/win2018/entries/computability/.

19. For a further link between Turing and Holmes, see Chapter 10, p. 123.
Open Access  This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter’s Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter’s Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.