MACHINE LEARNING WAS featured in episode 193 of Software Engineering Radio with Grant Ingersoll in 2013. But because this area has changed considerably in the past four years, it made sense to revisit it with a fresh outlook. In episode 286, Edaena Salinas talks with Katie Malone, a data scientist in the R&D department at Civis Analytics, which specializes in data science software and consulting. Katie earned a PhD in physics from Stanford University; during her studies she searched for new particles at CERN. She teaches Udacity’s Intro to Machine Learning course and hosts Linear Digressions, a podcast about machine learning (lineardigressions.com).

Here, Katie and Edaena discuss the major types of machine-learning algorithms and some examples, including supervised and unsupervised classification. Portions of the interview not included here for reasons of space include topics such as cleaning the raw data, training data versus test data, randomization, evaluation metrics, and Katie’s take on popular programming languages. To hear the full interview, visit se-radio.net or access our archives via RSS at feeds.feedburner.com/se-radio. —Robert Blumen

Edaena Salinas: Machine learning is widely used—in search engines, speech recognition, language translation, Netflix recommendations, and most recently in driverless cars. In the coming years, we’ll see it used in more fields. So, what is machine learning?

Katie Malone: My background is in science. I believe that there’s truth in the world and that science is one of the ways we get to that truth. It’s really hard to measure truth directly. Instead, we collect data on the world. If we analyze that data, sometimes we can pull out the truth. A true thing about the world might be, “I’m interested in watching this movie.” Or it might be, “There’s a good way to translate this sentence from English to French.” Machine learning is, in my view, a suite of tools that allows you to analyze data to figure out what’s going on in the world, and how that’s expressed in the data.

Usually it involves heavy computational lifting. The “machine” component implies computers. Then there’s usually a heavy dose of statistics, and often additional scientific fields. If you’re studying human behavior, you should be aware of [other fields that study humans] like behavioral psychology and economics. Those areas give you context about the thing you’re interested in.

Edaena: How does machine learning relate to AI?
Katie: I once heard that machine learning focuses more on understanding—measuring or making predictions—while AI is thinking one step further. Once we understand what’s going on, how can we make better decisions? How can we change the way we do things to take advantage of those insights? AI adds a layer of decision making on top of machine learning.

Edaena: Let’s walk through a simple example of machine learning: spam detection in email. Once I indicate that an email is spam, I’m telling the system something. What happens under the hood?

Katie: Email is an example of supervised classification. Let me break this into two parts. Supervision occurs when you have the correct answer for some of the cases. In this example, you provide the answer when you manually label the email as spam. If you don’t tell the model that it’s spam, then the model assumes it’s a legitimate email. Classification is sorting things into two buckets: spam and not spam.

Machine learning is making predictions based on the attributes of an email. We learn what spam email looks like, and then we extrapolate those patterns onto new emails to predict whether they’re spam or not.

The model is probably going to look at the words in the email, and potentially the sender’s domain. Spam emails have very particular patterns. The words in spam tend to be distinctive—usually they’re trying to sell you something with lots of superlative adjectives. Or maybe they’re trying to get you to send money to somebody in a foreign country. Very often there are grammatical mistakes.

Based on the presence of particular words like “Nigerian prince,” from the cases where you have said “this is spam,” the model can learn those patterns and apply them to new cases. Hopefully at some point you don’t have to manually label emails because the model will have figured out what spam looks like.

Edaena: By that time, is the system able to figure out those common words?

Katie: Spam is an interesting case, as presumably spammers are getting more sophisticated. The spam filters that worked five years ago probably wouldn’t work that well right now. That’s another important aspect of machine learning: it’s pretty rare to have a problem that you solve once and for all. Usually you want to revisit it periodically to see if the solutions you came up with last year or last month still apply.

Spam is a good example of that. I don’t know if people talk about Nigerian princes anymore, because that’s such a cliché at this point, but the formula of “We’re going to pretend there’s money sitting in an account and if you send a small deposit we’ll release it to you” remains popular, although the exact details change. In that scenario, you have to keep retraining your algorithm to continue to make good decisions.

Edaena: How is this information represented? Is there a specific format for the model?

Katie: The simplest thing you can do is to treat each word as its own feature. Many machine-learning algorithms assume there’s a big matrix of attributes. Imagine a matrix as a big data table, and each row of the table is an email and each column is a word. If a particular word shows up in a particular email, you’ll get a 1 in that spot in the matrix; if it doesn’t show up, you’ll get a 0. Then you can put that matrix into a standard machine-learning algorithm, and it will find the structure in the matrix that allows it to understand which words are most closely associated with the emails you’ve classified as spam.

Another important aspect of machine learning is thinking about different representations of your data. What I just described is the simplest way you might represent data in an email, but there are other algorithms that can be more compressed with respect to how the words or sentences showing up in an email are represented. How you represent your data has an intimate connection with the type of algorithm you’re going to use to do the supervised classification. The way the data is formatted can make it very easy for us to find the truth we’re seeking, or it can make it very hard. It’s worth thinking about carefully.

Edaena: What’s one way in which this data has been compressed in other data structures?

Katie: One case is Netflix movie recommendations. Imagine that each person who watches movies is a row and each possible movie they could watch is a column. Most people will only watch one percent of all the movies out there. And most movies are not going to be watched by even a significant fraction of all users. You have a big sparse matrix.

In a case like this, you can use matrix factorization. Instead of having this big sparse matrix, imagine that there are two factors—two different types of attributes—that we’re trying to understand. In this case, users and
types of movies.] Say there are buckets, or segments, of users, and users in each bucket watch certain types of movies or certain mixtures of movies. Then we have types of movies: action movies or foreign documentaries. Whether a particular user likes a particular movie is a combination of the type of user and type of movie. Representing the same data differently can make it easier and more direct to figure out whether a user is going to like a movie.

**Edaena:** How does machine learning handle cases where there’s no prior data?

**Katie:** In recommendation engines, there’s the classic problem of “cold starts.” This is when a new movie will be added to Netflix next month, and they need to figure out if a lot of people are going to want to watch it. Should they give it valuable real estate on the front page to advertise it? And to which people? But they have no data on this movie yet. They don’t know who has watched it or who liked it before. This is a tricky place to be in terms of machine learning, because machine learning is usually about pattern recognition, and there’s no pattern yet.

But if you have some other contextual information, like “this is an action movie,” then you have a better place to start from. You have some idea of the people who like action movies. This is effective at the beginning, and then you can refine those estimates as you collect more data.

**Edaena:** Can supervised learning be applied to values that are continuous instead of discrete?

**Katie:** That’s usually called regression. A lot of the same algorithms can be used for classification and regression, depending on the final type of output you want.

**Edaena:** Can you explain regression a bit more? For example, what is the objective of linear regression?

**Katie:** Linear regression has a continuous output. Classification is trying to figure out if something is A or B, spam or not spam. You wouldn’t say that spam has an inherently higher value than not spam or vice versa. There isn’t a natural ordering of those two things.

We’ll use income as an example. If you’re trying to predict somebody’s income from other attributes that you have, then obviously there’s a natural ordering. There’s a natural ordering to values like $10,000 and $100,000.

Linear regression tries to use known attributes about a person to predict another attribute, like income. Do I see a relationship between a person’s attributes and their income? One example is age. The older someone is, up to a certain point, the more money they tend to make. You might observe what kind of car this person drives. I know there are patterns in income versus the type of transportation you use: richer people have nicer cars.

None of these patterns is going to hold absolutely for every single case, but statistically it’ll usually hold. And from that you can make predictions. The quality of those predictions will depend on how good your data is, and to a lesser extent, how good your algorithm is. But you’re getting a little closer than if you were to make a shot-in-the-dark guess.

**Edaena:** If we plot the income on one
axis and the car’s price on another, you could predict from income how much you spend on a car, right?

Katie: That would give you some idea. You could imagine fitting a line to the distribution you see. The slope of that line will give an estimate of someone’s income once you know how expensive their car is. For every data point in your dataset, the line gives a predicted income.

Edaena: How do we measure how good the line is compared to the data?

Katie: For a lot of machine-learning algorithms, there are standard metrics. Somebody drives an expensive car, so I think they make $100,000 a year. If you have the actual income in your dataset, then the dataset shows that the person makes $110,000 a year. The $10,000 difference is my error. Sometimes you square it for other reasons that we don’t have to get into. You sum that error for other reasons that we don’t have to get into. You sum that over your whole dataset and divide by the number of points. This is a metric for the “goodness of fit” of this line. For classification you can use accuracy, which is how many you got right divided by the total.

One of the big challenges of machine learning is figuring out if those metrics are really measuring the thing you care about, because usually they aren’t. You have to be a little smarter to figure out exactly what you care about, and modify the metrics to reflect that. You usually have good options, but the artistry is knowing when to leave those options behind.

Edaena: How is unsupervised machine learning different from supervised?

Katie: The canonical answer is that in supervised machine learning, you have correct answers that came with your dataset. With unsupervised machine learning, you don’t have that luxury. You just have data. The types of questions you can ask of that data are different and very often constrained by the fact that there’s no correct answer.

There are different types of questions you try to answer with unsupervised learning. In my experience, it’s really hard. It’s a different way of thinking about your data, because when you have the correct answer, you want to get as close as possible to it. With unsupervised machine learning, it can be very tricky to try to understand what a good answer looks like.

Edaena: What do some of those questions look like?

Katie: The biggest one in my experience has been clustering. Clustering is the idea that you have blobs or coherent clumps in the data. You want to find them. This is hard because with a lot of real-world datasets, you don’t know if there are clusters in the dataset to begin with. If you don’t find any, it’s really hard to know if it’s because you’re doing a bad job or because it doesn’t exist. This gets back to the idea of truth—what are we trying to understand here?

Principal component analysis is another unsupervised technique. You’re trying to find ways...
of compressing the data down to the aspects that make it the most variable. You’re trying to find directions in your dataset in a lower-dimensional space that maximize the variance in your data.

The point is that this is something you can do without knowing any “correct” answers for your data, which can be pretty useful when you don’t have labels to rely on.

Edaena: Is unsupervised machine learning widely used? Are there any systems that we interact with that might be using it?

Katie: Segmentation within marketing is one example. If you think your users might fall into a few main buckets, you want to segment them. For example, buyers of computers might be power users, programmers and data scientists, people who watch movies and use Facebook, and people who use it as a work machine but don’t program on it. Maybe there’s a dataset that would allow you to pick out these distinct groups.

Edaena: Unsupervised might also be more about discoverability, because you don’t necessarily know what you’re looking for.

Katie: That’s fair. Supervised methods are used when you know what you’re trying to answer. Unsupervised methods are for when you don’t have that labeled data available or when you don’t exactly know how to slice and dice the data yet.

EDAENA SALINAS is a software engineer in Microsoft Research’s Knowledge Technologies Group. She also hosts The Women in Tech Show (thewomenintechshow.com). Contact her at edaena@thewomenintechshow.com.