Artificial Intelligence Foundations: Machine Learning with Doug Rose

(LinkedIn Learning)

What it means to learn:

- The term **machine learning was coined in 1959** by with computer pioneer, **Arthur Samuel**, who wondered if computers could learn behavior rather than programmed for specific tasks
- Most programs are series of very specific instructions but in machine learning, you are **giving the computer the data and tools it needs** to study the problem and solve it as well as remember what it did, so it can adapt, evolve and learn
- Similar to how humans learn:
  - observe problem →
  - create rule based on experience →
  - apply rule toward larger action →
  - await feedback →
  - adjust rule as necessary and add newly found data to memory
- Machines start with a small part of the data and use **statistical algorithms** to see how the data fits together. It uses the input data to find patterns. It then tests it output against the training data to see how correct it was.
- Every time the machine learns something new, it stores it in its database, i.e. long term memory, so it can **improve and adapt**.

Work with data:

- Most of machine learning is still about working with specific data
- Typical programs accept input and produce **output based on an algorithm; input is command and output is predetermined response**
- **Programming model** ⇨ allows the machine to learn and respond to feedback
  - i.e. **word-filter program** to filter out spam messages
- With machine learning, instead of inputting instructions, you input data, and instead of a predetermined response, you work with machine learning algorithms to help the machine learn how to respond
- Data is **split into test data and training data**. Training data is used to find patterns.
- The model used helps the machine make sense of the data, find patterns, and make accurate predictions.
- **BINARY CLASSIFICATION** ⇨ ex. use to split emails into two groups between spam and good
  - Machine comes up with a score to show the likelihood that a message is spam
- **Find best classification algorithm to use, tweak the hyperparameters** of the algorithm until the machine does a good job at classifying
- Once the model is effective, it becomes the active data model

Apply Machine Learning:

- **Weather apps, search engines**, organizations with a lot of data ⇨ all of these are good candidates for using machine learning, ex: **suggested videos and customized searches** based on the data from past interests and what a viewer has watched
- Google, Apple, Netflix, Amazon, YouTube, etc. all use machine learning **to better understand us**, the customers
- Automatic Translation via natural language processing → translating speech into text and translating from one language to another.
- In machine learning, **you are using artificial intelligence to help your program find patterns in massive datasets**, often finding ones that humans could never see
- It is not just an advanced form of human learning, but a way to find patterns, make decisions, and find better insights ⇨ **Think about how the machine learns**
- Before starting a machine learning project, you MUST consider your data: is it high enough quality, is there enough to learn something new? This is how the machine learning program will view the world
- The broader the view, the more likely you will find something interesting ⇨ You want high quality and diverse datasets

Different Types of Machine Learning:

- Machine learning is not just statistics or data science. The key word is **LEARNING**
- Analogy to learning chess: tutor plus observing for hours and hours as pros play
- **SUPERVISED LEARNING**: data scientist is the tutor for the machine
  - Need a knowledgeable tutor
- **UNSUPERVISED LEARNING**: machine makes observations, given just names and labels, and finds patterns on their own
  - Need access to A LOT of data
- **SEMI-SUPERVISED LEARNING**: train the machine just a little for a high level overview, and learning comes from observing patterns it finds
  - Need both a good tutor and good observations

Chapter Questions:

⇨ Drithi has formal badminton lessons every Tuesday afternoon, and plays with whoever is available on Saturdays. Which type of learning is she using?

Semi-supervised

Drithi learns rules by instruction on Tuesdays, and can then explore them on Saturdays.

⇨ In the learning process, what would you typically do after implementing a rule?

Observe feedback

this corresponds to testing in a machine-learning process.

⇨ Which sort of problem is most explicitly amenable to a solution by machine learning?

analyzing menu selection preferences

Looking for patterns in large datasets is the main strength of machine learning.

⇨ Instead of explicit, stepwise instructions, what are the fundamental inputs to a machine-learning process?

Data

A machine learns from data, information, or experience.

Supervised Learning:

- ➔ In supervised learning, you show the machine the **connection between different variables and known outcomes**, known as labeled, sampled data and the correct output
  - Labeled data - because it is already tagged with identifying information
- ➔ Start by creating a **set of labeled data**:
  - Example: app to determine how long a drive home will take using labeled inputs about weather, time of day, and holiday status. The output would be the amount of time it took to drive home on a particular day
  - **Independent variables = input**
  - **Dependent variable = output**
  - Use different machine learning algorithms to make connections between the input data
  - **Statistical regression** - determine how the independent variables affect the dependent variable - Machines have to rely on data and statistics
- ➔ **FIRST: Create a training set**
  - Machine might see a **direct relationship** between rain and drive home
  - Might also see a connection between the time you leave work and time to get home
  - Machine finds SOME of the relationships in your labeled data. This is the start of your **data model.**
  - Machine applies the info learned from the training set and applies it to the test data and sees if the connections hold true
  - Give machine feedback on how accurate it was on its prediction
  - Machine will learn and adapt its model, making adjustments based on the new data
  - **KEY**: In supervised **you know a lot more about the training data** - you can feed labeled data directly into the machine that is easily classified
  - **Labeled data is the key difference** between supervised learning and other forms

Unsupervised Learning:

- ➔ Learning through study and observation, **learning and improving by trial and error**
- ➔ **NOT working with labeled data, not showing machine the correct answer**
- ➔ Different algorithms to let the **machine create connections by studying and observing**
- ➔ **Multiclass Classification** ⇨ data is sorted into several different groups
- ➔ Machine studies and observes and learns to **cluster data into groups**
- ➔ **Better chance of finding or learning something new** by trying unsupervised learning, **creating its own clusters** based on observations on the data
- ➔ **KEY: Access to MASSIVE amounts of data!** The more data, the easier for the machine to find trends and clusters

Semi-supervised Learning:

- ➔ Cross-over, taking advantage of both supervised and unsupervised learning

- ➔ Starts with a **smaller training set** for the machine to learn and perform some basic classification, like the training set in supervised learning, but here the set is fed to the machine, which is allowed to learn and observe based on the data, like unsupervised
- ➔ **Allows the machine to expand its understanding based on the input's classifications and labels**
- ➔ Labeled data allows the computer to use **INDUCTIVE REASONING**
- ➔ Each time the machine **induces a new connection**, it does its best to **improve the model**
- ➔ **TRANSDUCTIVE REASONING** - allows you to narrow down unlabeled data by thinking about what you already know about the collection. **Must think about the data in a larger context**
    - Tries to improve the model by **making better guesses about what is in the unlabeled data using context**
- ➔ **Semi-supervised learning is not so common**, but it works well in certain areas:
    - When it is not difficult for the machine to create useful groups
    - Using smaller data sets
    - Classifying web pages, group photos of a certain type ⇨ only makes sense when the other two approaches have had trouble with your data
    - Inductive and transductive reasoning can lead to greater errors and mislabeled data
- ➔ Semi-supervised is good for **particular situations**, **NOT necessarily the best place to start**

Reinforcement Learning:

- ➔ Unlike the previous three methods, where you are trying to create the best model for your machine to classify different datasets or find meaningful clusters then allowing the machine to work with the rest of the data, **reinforcement learning has the machine iterate over data and over time, the machine zeros in on high quality output**
- ➔ You are reinforcing the ways you want the machine to behave
- ➔ Instead of relying on observation, **you are giving the machine a very clear goal**.
- ➔ **Q-learning** ⇨ **states** or set environments and possible **actions or responses** that respond to these states ⇨ **Q = the QUALITY of the outcome**, better rewards to get a better outcome
    - Start out with a quality of zero and have the machine learn which actions improve the quality or conditions
    - The **quality goes up based on the states and actions**
    - Allows the machine to go through **endless simulations of actions and states and find the best strategy**
- ➔ **One of the most promising areas** of machine learning
- ➔ Reinforcement and Q-learning allows machines to quickly grow beyond our understanding and helps skip steps of unsupervised learning
- ➔ Does NOT require as much time or as much data

Chapter Questions:

⇨ What distinguishes supervised machine learning from other types of machine learning?

using labeled data for training

One has a lot of prior knowledge about the training data in supervised learning.

⇨ How are induction and transduction different?

Transduction uses more information and produces more specific rules.
The results of transduction are not as general.

⇨ A company has a variety of efforts that it uses to find new antiviral medications. Which effort best exemplifies unsupervised learning?

the screening of a large collection of botanical extracts
This process is open to massive amounts of uncategorized data.

⇨ What does Q-learning use instead of training or testing data?

simulation or experience
The quality is optimized by iteration.

⇨ Why would you choose to not use semi-supervised learning all the time?

It can lead to larger errors or confusion.
Transduction and induction can both be very misleading.

Problems That Use Machine Learning:

- **Binary classification**: yes or no → generally only **two possible outcomes**, predefined right or wrong answer → always uses **supervised learning**, labeled data, must have a developer at the beginning to set the criteria
- **Multiclass**: there can be limitless classifications, classifies based on a number of predefined categories
- **Regression**: have a **continuous solution**, you look for trends rather than yes or no, range of possibilities that are more or less likely, uses much **statistics**, uses **supervised learning**
  - **Linear Regression:** shows a trend like for prediction

Decision Trees:

- Can be used for **binary classification** challenges with **supervised machine learning**, set up **predictors and connect them to some outcome**
- Create a graphical tree for presentation
- Multiple classes of predictors, two outcomes, binary
- Based on different combinations of predictors
- **Root node predictor** ⇨ the very bottom trunk of the tree
- **Decision node** ⇨ contains all the options from the root or previous node
- **Leaf nodes** ⇨ next step of breaking down predictors
- Too much **Entropy** ⇨ tree has gotten too messy and taking too long to reach a yes or no
  - When you reach a node or point where you are **not really gaining much information** about how to predict the result
  - Can be remedied by **splitting with a different predictor**

K-Nearest Neighbors:

- **SUPERVISED → Classifying data with what you already know** is the best way to learn more about your data and grouping by certain characteristics
- Very often used for **multiclass classification**

- **Instance based / lazy learning** type of algorithm, where the bulk of the computation takes place right before classifying data. **Learning is NOT continuous**
- Computation is all run in one big instance
- Immediate reward for the size and quality of training data
- Downside ⇨ **requires high computational power**, difficult with extremely large data sets.
- Trying to **minimize the distance** between a known and an unknown for classifying the unknown based on the known
- The **closer you are to your nearest neighbors**, the more likely you are going to be accurate
- Uses **Euclidean distance** between data points
- **Predictors** ⇨ key characteristics that are common between two classes
- Often used in **finance** for example to find best stocks and predict future performance

K-Means Clustering:

- **UNSUPERVISED → used to create clusters based on what machine sees in the data**
- The **K refers to how many clusters** or classes the data will be divided up into
- **Centroid data** ⇨ the data randomly chosen to represent the cluster to which they have been assigned.
- Other data is clustered based on the average distance it is from the centroid data
- Machine will **iteratively try to get good centroid choices** for data.
- **High overlap of data** ⇨ when data repeatedly ends up in different clusters
- Challenge ⇨ can be very **sensitive to outliers**
- Used for things like retailers deciding who which customers promotions and create strategies based on the categories created

Regression:

- **K-means and KNN ⇨ both INSTANCE based, or LAZY LEARNING**, meaning that you get all the answers in one big splash, which means **sensitive to any changes in data**
- Regression is the opposite ⇨ it is **CONTINUOUS**
- This looks at the **relationship between your predictors and outcome**
- Predictors can also be called **input variables, independent variables, or regressors**
- Try many different predictors, compare to outcome, then keep reiterating and **try to find the most accurate predictors of the outcome**
- When you find a good model (result), you try it out on your testing data
- **LINEAR REGRESSION** ⇨ one of the most common types of machine learning algorithms
  - Create a straight line showing the relationship between your predictors and outcome
  - **Line (hyperplane, trend line)** should ideally split the data into two classes
  - Having a lot of outliers makes linear regression very difficult
  - **The more data ⇨ more accurate trend line**

Naive Bayes:

- Supervised ⇨ classify data, unsupervised ⇨ cluster data
- Naive bayes is based on **conditional probability → how likely is one thing given another**

- ➔ Based on **Bayesian statistics, called naive because it assumes all predictors are independent from one another, considers each variable independently**
- ➔ Mostly used for **binary or multiclass classification**
- ➔ Starts with **class predictor probability** ⇨ looks at each independent predictor and tries to come up with the **probability that the data in question belongs to each individual class**
- ➔ **Weighted multiplication function** ⇨ create weights based on which predictor is the most accurate

Chapter Questions:

⇨ Why is k-nearest neighbor also called lazy learning?

It uses a lot of computation for every instance.

It is a brute force matching method.

⇨ What characterizes an outlier in cluster analysis?

not being close to any centroid

The outlier is forced into the nearest cluster, even though it is not a good fit.

⇨ Why are dog weight and height not necessarily good choices for predictors in naïve Bayes methods?

They are correlated.

Naïve Bayes assumes that the predictors are independent or uncorrelated.

⇨ Why are regression methods not considered good examples of machine learning?

They are based on statistical predictions.

Linear regression is an established statistical method.

⇨ What should you do if your decision tree has too much entropy?

Add or substitute predictors.

One or more of the predictors is not acting efficiently.

⇨ Which one of the following do you use when you look for trends instead of trying to classify data into different groups?

regression problems

These problems tend to have a continuous solution thus you look for trends.

⇨ Salim wants to predict the best harvest date for his orchard based on prior weather reports and harvest histories. Which type of tool does he require?

regression analysis

There are nearly continuous ranges of variables involved.

⇨ Which of the following is instance-based (or lazy learning)?

K-Means Clustering and K-NN

K-Means Clustering and K-NN are lazy learning. You get all the answers in one big splash

Follow the Data:

- ➔ **Bias** ⇨ the difference between your predicted value and the actual outcome
- ➔ **Variance** ⇨ when predicted values are scattered, AKA noise in the data
- ➔ High bias and low variance = consistently wrong

- ➔ High bias AND high variance = consistently wrong in a very inconsistent way
- ➔ Bias and variance are ways of measuring the difference between your predictions and outcomes
- ➔ Knowing the measure of each is how you know the direction in which to tweak hyperparameters and improve predictions

Fit the Data:

- ➔ **Training set** ⇨ smaller set within the larger dataset and used to prepare for testing against the testing set
- ➔ **Underfitting** ⇨ when you create an inflexible prediction that fits the training set but misses a lot when used on the testing set, not capturing enough
- ➔ **Overfitting** ⇨ extremely flexible and too complex, fitting too closely to the training set but misses a lot on the test set
- ➔ When you add **more complexity,** you make your model more flexible, but also it becomes more difficult to manage and recognize connections between variables
- ➔ **Signal** ⇨ used to make good prediction
- ➔ **Noise** ⇨ natural variances in the data that might not offer insights
- ➔ Want to focus in on the signal and filter out the noise

Select the Best Algorithm:

- ➔ If **data is labeled** ⇨ choose supervised learning
- ➔ Labels help you understand both input and output
- ➔ If **data is unlabeled** ⇨ choose unsupervised learning
- ➔ Once you have clusters from an unsupervised learning model, you can extract some meaning
- ➔ Massive amounts of unlabeled data ⇨ maybe K-means Clustering is right
- ➔ Lots of labeled data ⇨ K-nearest neighbors, regression, or decision tree
- ➔ Possibly try a bunch of different algorithms and then look at the results
- ➔ Ensemble modeling ⇨ working to create different ensembles of machine learning algorithms
    - **Bagging** ⇨ when you create several versions of the same kind of algorithm, taking the best or averaging out the outcomes.
    - **Boosting** ⇨ using several different algorithms to boost the accuracy of your results
        - Ex: taking a decision tree's leaves and letting an unsupervised model see if there are any interesting groupings → semi-supervised learning
    - **Stacking** ⇨ using several algorithms in a stacked way to improve accuracy
        - Feature-weighted linear stacking ⇨ model used to win the Netflix award
        - Ex. K-NN on top of naive bayes
        - Improvement adds up
        - Some competitive data scientists will stack over 30 algorithms

Chapter Questions:

⇨ Hugo's modeling strategy uses k-nearest neighbor, followed by regression analysis. What does his strategy exemplify?

stacking or boosting

These are ensemble techniques using methods in series.

⇨ Lydia has produced a model with 100 predictors that is capable of almost perfectly fitting the 200 observations in the training data. How would you improve Lydia's model?

by reducing its complexity

Using 100 predictors is perhaps too complex.

⇨ Which scenario is an example of high bias and low variance?

predictions that are almost always 7.5 pounds too high

The predictions are consistently high.

Machine Learning Challenges:

- ➔ **Ask interesting questions** → embrace a more exploratory mindset
- ➔ Keep **training data separate** from testing data → never mix back together. It can help the machine cheat by giving the answers.
  - You want to keep high generalization for accuracy, and mixing can ruin that.
  - Do not overstate the effectiveness of the model → make sure that the testing data is used for presentation, so people have a realistic view of its accuracy
  - A bias toward a certain algorithm is ok if it works
- ➔ **Time investment** ⇨ don't spend TOO much time choosing the right algorithm

Chapter Questions:

⇨ Most of Atul's experience is with decision trees and regression. Which strategy should he embrace as manager of a new project?

applying decision trees and regression to the new problem

Atul can often make fast progress by exploiting available expertise.

⇨ You finished fine-tuning your model with training data, and are eager to show the results to your business team. What should you do instead?

Show predictions for the testing data.

This gives the business team realistic expectations.