LEC 1: What is AI/ML

Feb 5, 2020

Introduction and data gathering

Name, Major (encode as quant/not quant), Gender, Year, Cats or Dogs?

Enrollment logistics

1. Sign up link (shorturl.at/arJTZ)
2. If you sign up and attend the first two lectures, you're eligible to be enrolled
3. We might need to do lottery if > 30 people
4. We will share enrollment codes next lecture
5. Pre-req: Stat 20 or similar

Grading policy

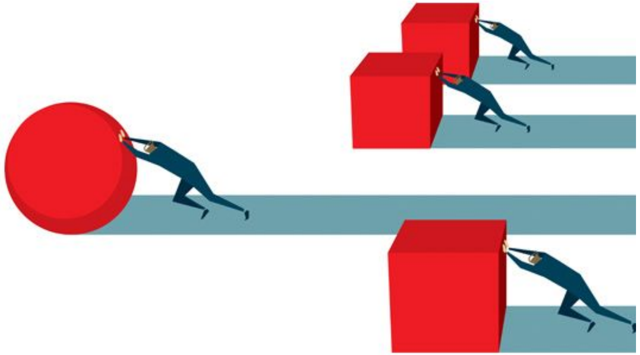
1. 2 excused absences, complete mini-quizzes, say 1 thing per class
2. Pass the open-note final exam (not intended to be difficult/math-heavy)

https://tinyurl.com/ueu6b8b

Course outline

1. What is AL/ML
2. Math and statistics review
3. Fundamentals of ML (Bias-variance, various regressions models, PCA)
4. Classification models (k-nearest neighbors, decision trees, SVM)
5. Introduction to Neural Network (various different NN models)
6. Optimization methods (LP, GD, SGD and etc...)
7. Case studies

Motivation for this course



Why learn ML at this depth?

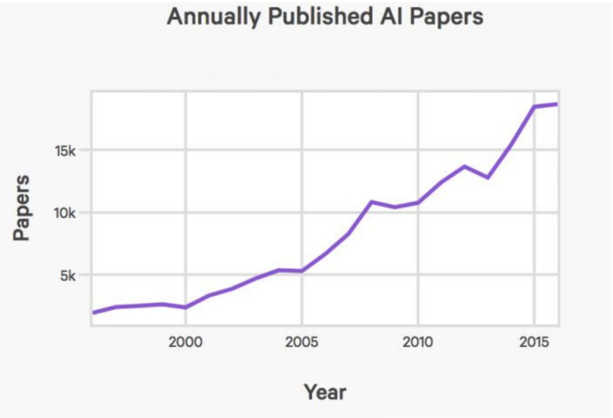
Your competitive advantage as management

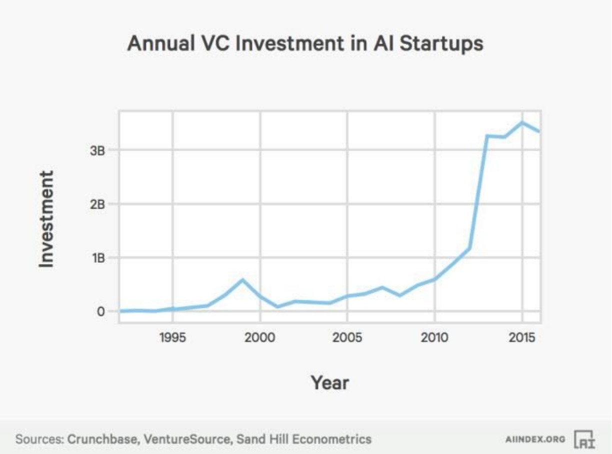
Don’t piss off the engineers

1. Pre-empt bias, assess success and failure, make decisions off data
2. How to articulate problems to your data scientists and assess feasibility
3. Know what you don’t know

Why you should understand ML

- ML has exploded in popularity in recent years, even though many ML models have existed for decades
    - Increased computing capacity
    - Cheaper, accessible, and more plentiful data
- Companies across all industries use ML across various business functions

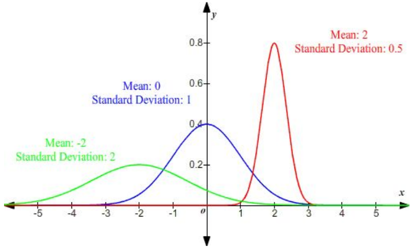




What is ML and how it can be used

Tom Mitchell in his book Machine Learning provides a definition in the opening line of the preface:

“The field of machine learning is concerned with the question of how to construct computer programs that automatically improve with experience.”

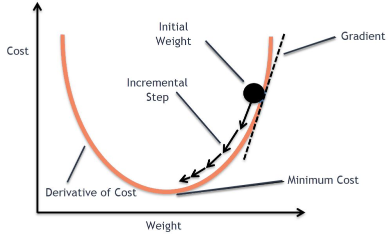


More concretely:

1. Statistically model the situation from data

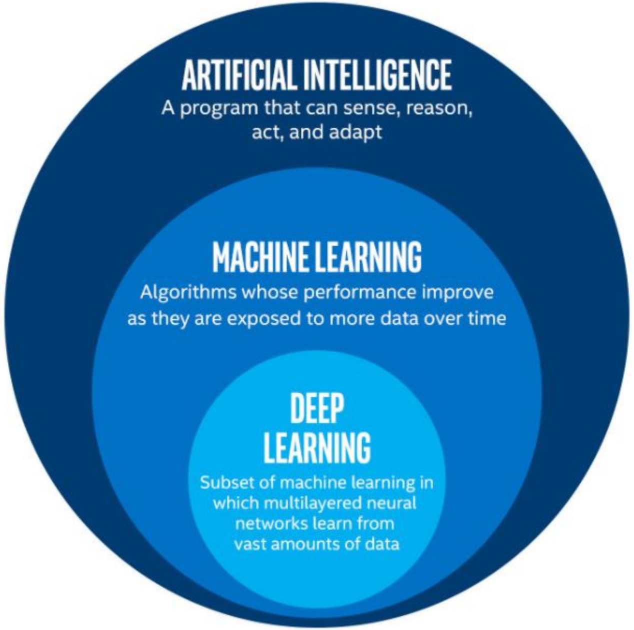
$$L2LossFunction = \sum_{i=1}^{n}(y_{true} - y_{predicted})^2$$

2. Derive a cost/error/loss function with unknown variables aka “weights”



3. Optimize the cost function and determine best model wrt. weights

What is ML and how it can be used



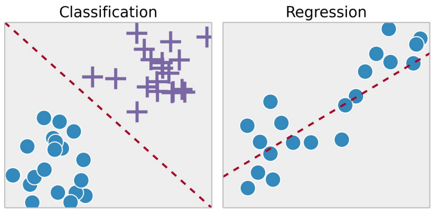
Characteristics of AI:

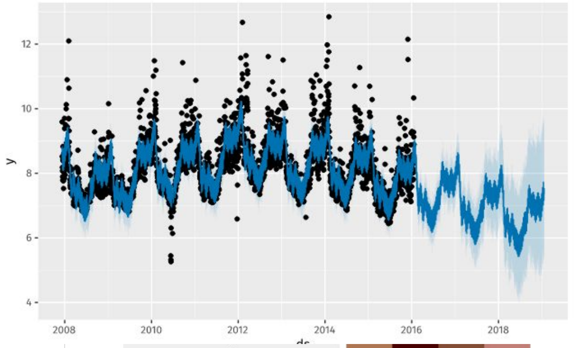
Defined solution space: Classification (Yes/No) or regression (given inputs, output?) or generation (10x10 image)

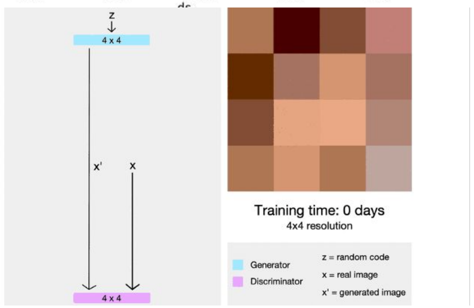
Exploratory: you don't need a testable hypothesis. E.g. you don't need to say feature X determines Y

Categories of ML applications

1. Classification
   a. Discrete result *note the use of logistic regression for binary classification, e.g. hot dog vs not hot dog
2. Regression
   a. Continuous result, predicting one or more variables from others, e.g. house price based on features
3. Meaning extraction
   a. Natural language processing: LDA
4. Forecasting
   a. Signal decomposition (+ regression) e.g. https://facebook.github.io/prophet/ seasonality, Google Flights
5. Generation
   a. Pick features from data + add noise
6. + more



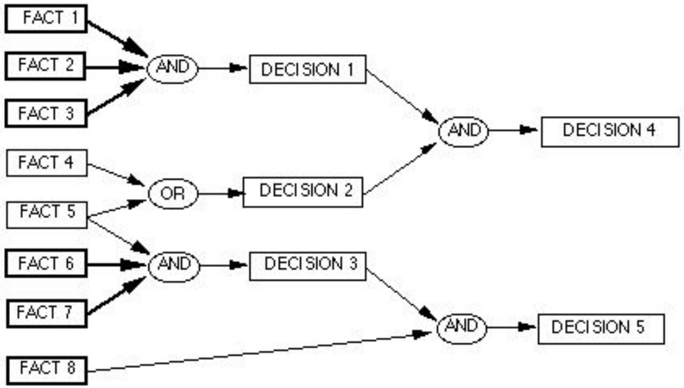




What is not ML and what are non-applications

Not all problems solved by computers is ML

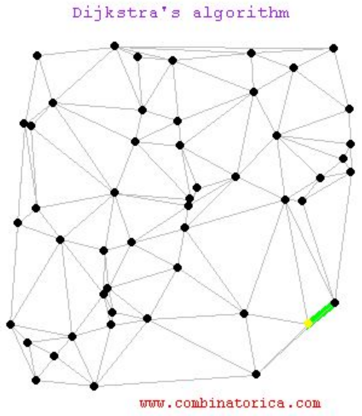
1. Rule-based systems (e.g. simple autonomous vehicles, some chat bots)



What is not ML and what are non-applications

Not all problems solved by computers is ML

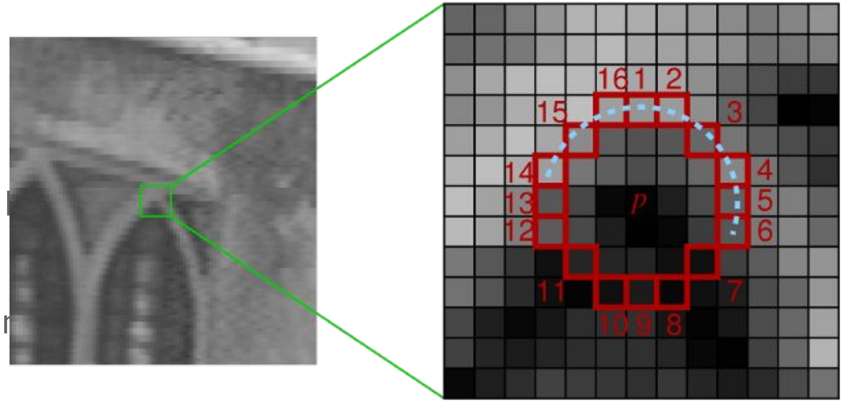
1. Rule-based systems
   a. E.g. simple autonomous vehicles, some chat bots
2. Deterministic algorithms
   a. E.g. A* or Dijkstra's (shortest path algorithms in maps) based on comparisons
   - There is no predictive power in Dijkstra's without running the algorithm on the new dataset, purely optimization



What is not ML

Not all problems solved by computers is ML

1. Rule-based systems
   a. E.g. simple autonomous vehicles, some chat
2. Deterministic algorithms
   a. E.g. A* or Dijkstras (shortest path algorithms i comparisons
   - There is no predictive power in Dijkstra's without running the algorithm on the new dataset, purely optimization
3. A/B testing - no predictive capability
4. Many aspects of Computer Vision
   a. E.g. Canny edge detection



What is not ML and discretion

Rule of thumb: ML is advantageous on tasks that don't require a good understanding of the inner mechanics of the system that created the data.

Examples?

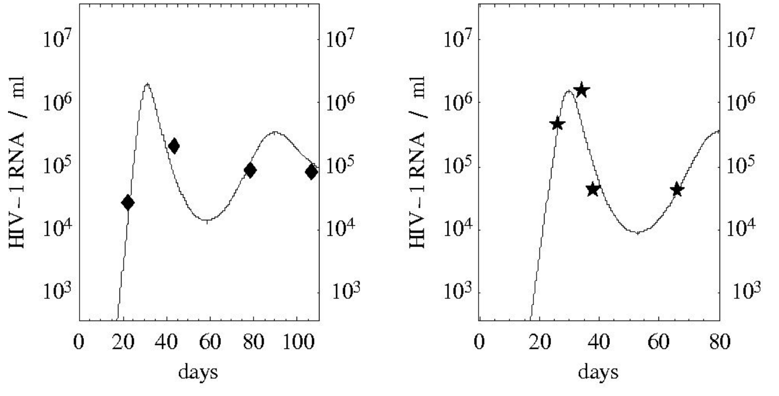
Discretion required

1. If you need certainty on the result - sometimes false positives could be damaging
    a. If you consistently get 99-100% accuracy, you can try to develop a rule-based method because it means there's some defining feature set
2. Some machine learning methods lack interpretability
    a. DotLab (https://www.dotlab.com/)
    b. NN for medical imaging - don't know if it's fitting to noise or the tumor
    c. Predicting the weather based on yesterday's rain, temp, wind speed: ignores meteorological patterns

Discretion required

3. Insufficient data

a. Heavily skewed data: detecting breast cancer markers in white women but not black women



4. When the past does not indicate the future

b. Forecasting for GDP

ML in the news

1. AI beats Wall Street analysts on financial forecasting
   http://news.mit.edu/2019/model-beats-wall-street-forecasts-business-sales-1219
   a. *57% of the time
   b. Do you think AI can do the job of analysts?
2. BlueDot "predicts" the Coronavirus outbreak
   a. https://economictimes.indiatimes.com/magazines/panache/this-tech-firm-used-ai-machine-learning-to-predict-coronavirus-outbreak-warned-people-about-danger-zones/articleshow/73697801.cms
   b. https://venturebeat.com/2020/01/31/ai-weekly-disease-coronavirus-prediction-spread/
3. Palantir mass monitoring, tracking car movements through surveillance cameras
   a. https://theintercept.com/2017/02/22/how-peter-thiels-palantir-helped-the-nsa-spy-on-the-whole-world/
   b. https://theoutline.com/post/3978/peter-thiel-knows-you-ran-that-red-light?zd=1&zi=wn6ta6si

https://developers.google.com/machine-learning/guides/rules-of-ml

Vocab

- **Instance**: The thing about which you want to make a prediction. For example, the instance might be a web page that you want to classify as either "about cats" or "not about cats".
- **Label**: An answer for a prediction task either the answer produced by a machine learning system, or the right answer supplied in training data. For example, the label for a web page might be "about cats".
- **Feature**: A property of an instance used in a prediction task. For example, a web page might have a feature "contains the word 'cat'".
- **Feature Column**: A set of related features, such as the set of all possible countries in which users might live. An example may have one or more features present in a feature column. "Feature column" is Google-specific terminology. A feature column is referred to as a "namespace" in the VW system (at Yahoo/Microsoft), or a field.
- **Example**: An instance (with its features) and a label.
- **Model**: A statistical representation of a prediction task. You train a model on examples then use the model to make predictions.
- **Metric**: A number that you care about. May or may not be directly optimized.
- **Objective**: A metric that your algorithm is trying to optimize.
- **Pipeline**: The infrastructure surrounding a machine learning algorithm. Includes gathering the data from the front end, putting it into training data files, training one or more models, and exporting the models to production.

What do you want to learn about?

- Discussing applications
- Dissecting theory
- Implementation
- Data skills

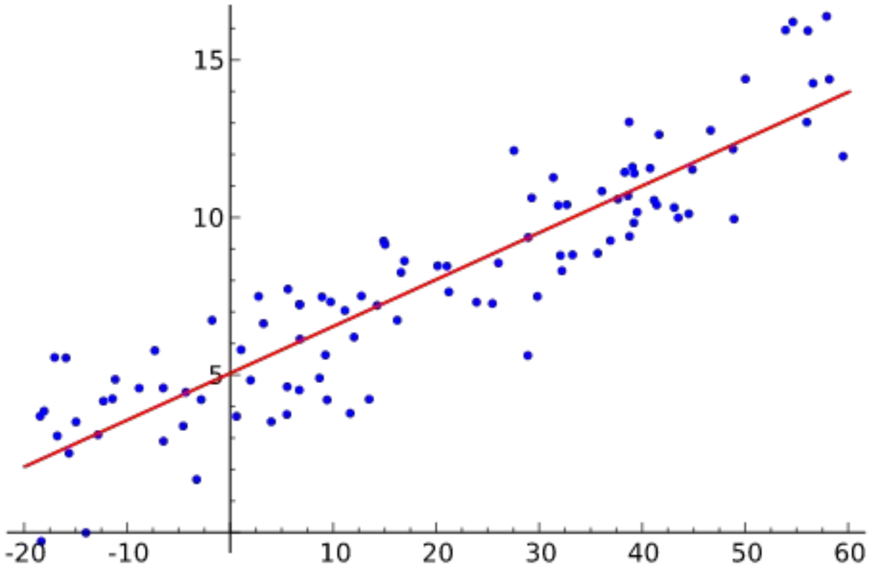
Linear Regression

Brief Intro to Linear Regression

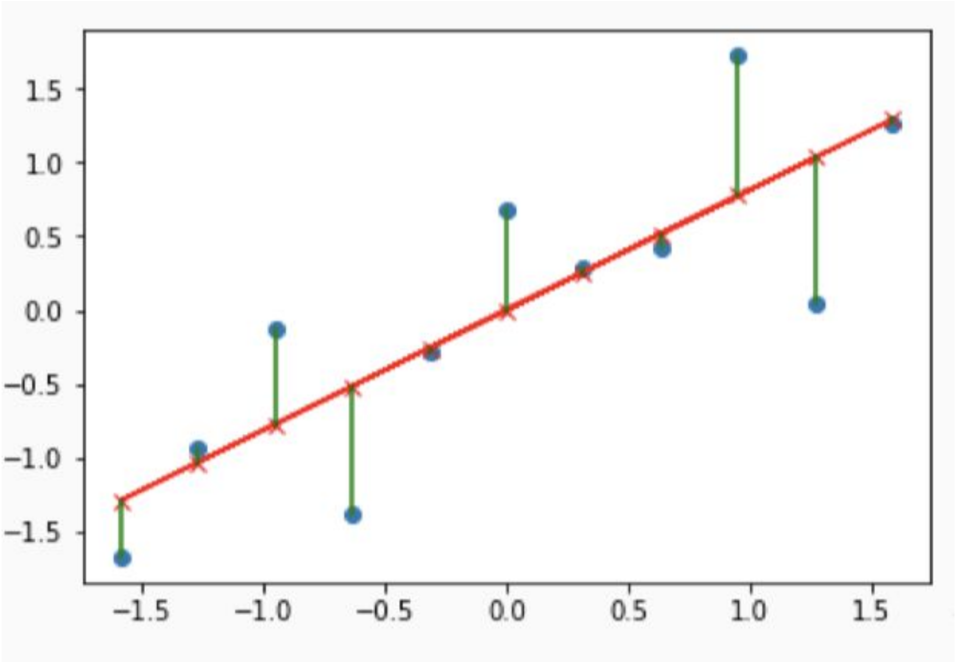
- Popular way of exploring relationship between one response variable and multiple explanatory variables ("features")
- Isn't usually used for prediction

$$Y_i = w_0 + w_1 X_i + \varepsilon_i$$

- Choose weights to minimize squared error: $\sum(Y_i - w_0 + w_1 X_i)^2$



Minimizing Squared Error



Multiple Regression

- Can actually be generalized to multiple features--very powerful if you want to study the impact of multiple features on the response variables

$$Y_i = w_0 + wX_{1i} + w_1X_{2i} + \ldots + w_kX_{ki} + \varepsilon_i$$

Home_price = 100k +

50k*(number_of_windows) +

200*(square footage) + ε