Applied Machine Learning

Course number: W207

Prof. Alexander I. Iliev, Ph.D.

Applied Machine Learning

Lecture 1 …

Introduction

Applied Machine Learning

- Weekly schedule: *(may change slightly)*
  - Week 1: Kick-off course
    Student introductions. Projects, assignments, requirements, breakout rooms, Introduction to Machine Learning.
  - Week 2: Nearest Neighbors
    Tutorial notebook. Review NN lecture. NN notebook in breakout rooms.
  - Week 3: Naive Bayes and text classification
    Discuss NB applications—spam (Graham), spelling (Norvig) in breakout rooms. NB Notebook in breakout rooms.
  - Week 4: Decision Tress and Ensembles
    Review DT lecture. DT Notebook in breakout rooms.
  - Week 5: Deep breadth
    Compare NN, NB, DT. Discuss AUC measure. Ensembles notebook in breakout room. Lecture review time permitting.
  - Week 6: Gradient Descent
    Review GD. Final Project discussion. Regression Notebook in breakout rooms.
  - Week 7: Neural Networks
    Review NN lecture. NN notebook part 1 in breakout rooms.

Applied Machine Learning

- Weekly schedule: *(may change slightly)*
  - Week 8: Applied SVMs and wrap-up of supervised learning

    Discussion of SVM libraries and their evolution. Comparison of algorithms learned so far. NN notebook part 2.
  - Week 9: Deep Learning

    Introduce convolutional nets (CNNs). K-means review. NN notebook part 3.
  - Week 10: Unsupervised learning.

    K-means (cont.). GMM review. Means notebook in breakout room.
  - Week 11: PCA and Case Study

    PCA review. Kaggle case study.
  - Week 12: Network Science

    Group discussion of network science, algorithms, visualizations, and different tools.
  - Week 13: Recommendations and Personalization

    Group discussions, paper review, share experience.
  - Week 14: Student Presentations

Course Content Outline

- **<u>Methods of Instruction</u>**

  Lectures, presentations and in-class discussions will be the main tools of instruction. Students should read the asynch material provided prior to the live class sessions.

- **<u>Course Grade Weighting</u>**

  Final grades will be based on
  - 3 Projects: 60% - *individual*
  - Final project: 35% - *group*
  - Participation: 5% - *individual*

- Students can meet up in Slack

Course Content Outline

Programming projects:

- This course includes 3 guided programming projects. They will be distributed at the beginning of the course and should be submitted (via github) by the beginning of your live session in the week specified below. They will involve filling in relatively short pieces of code in a python notebook and sometimes brief analysis of results.

- Late submissions will be accepted up to 1 week past the deadline with a 10% penalty, but you need to let your instructor know if you'll be submitting late.

- You may work alone or in groups but you need to write your own code. Discussion, especially about programming issues, on the wall is encouraged.
  - Project 1 Due: Week 5
  - Project 2 Due: Week 9
  - Project 3 Due: Week 13

Course Content Outline

Final project:

- At the midway point in the course, your instructor will share details about the final project. You'll choose from a list of relevant Kaggle competitions, run experiments, write up a notebook summarizing your work and key results, and give a short presentation in the final live session. You are strongly encouraged to work in groups.

- Baseline submission: Week 10
- Check-in with instructor: Week 12
- Notebook due and in-class presentation: Week 14

- The code for all problem set is posted here: https://github.com/smiletodaywithme/W207-Applied-Machine-Learning/tree/master/Notebooks

Course Content Outline

Programming environment:

- All the projects should run fine on your personal computer. Install python, ipython notebook, numpy, matplotlib, and scikits.learn. A number of other useful packages will be introduced during the semester.

- Both Enthought and Anaconda are free python distributions that include all the relevant packages.

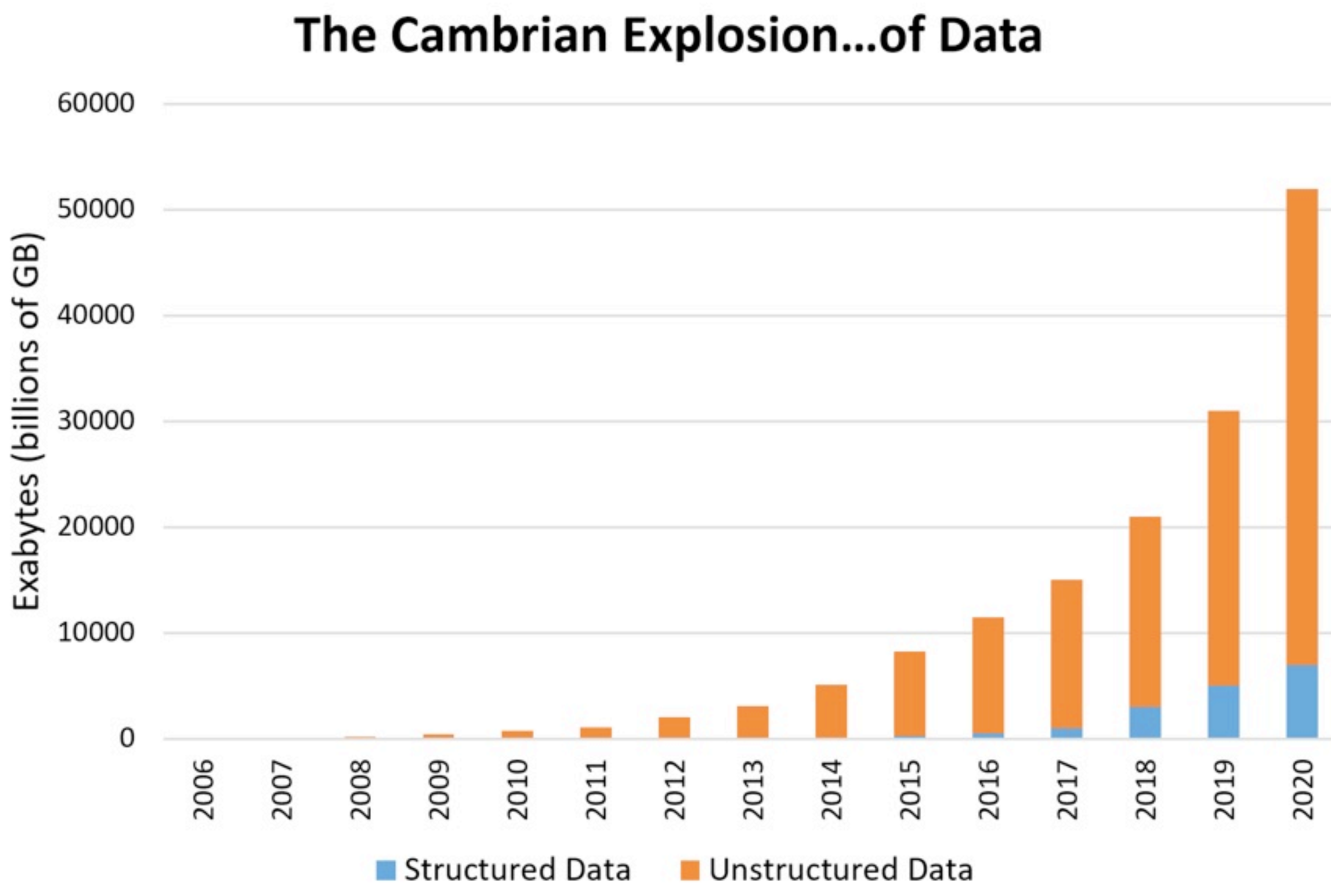
Applied Machine Learning

- Lecture 1 outline:
  - Projects, assignments and requirements
  - Data Mining vs Machine Learning
  - What is Machine Learning?
  - Supervised vs. Unsupervised Learning
  - Supervised learning methods: (mostly used in this class)
    - kNN – one of the earliest algorithms, using distance between observations
    - Naive Bayes – using Bayes theorem, estimating the probability of a class label, used in text analysis
    - Decision trees – random forests, gradient boosted trees, - one of the most useful methods
    - Regression (linear and logistic) – used in finance and in the business world
    - Regularization – estimating sparse models (more attributes than observations)
    - Optimization using gradient descent – how to estimate parameters in models like NN, just like backpropagation
    - Neural Networks with backpropagation – introducing nonlinearity for solving more complex problems
  - Unsupervised learning methods:
    - Clustering – k-means, clustering
    - PCA – dimensionality reduction (ex. healthcare: disease prediction)
    - EM (Expectation Maximization) algorithms and Mixture Models – discover the mean and covariance of each cluster assuming Gaussian distribution
  - Overfitting – making the model too complex to be applied in the real world
  - Data source – Kaggle.com

Machine Learning

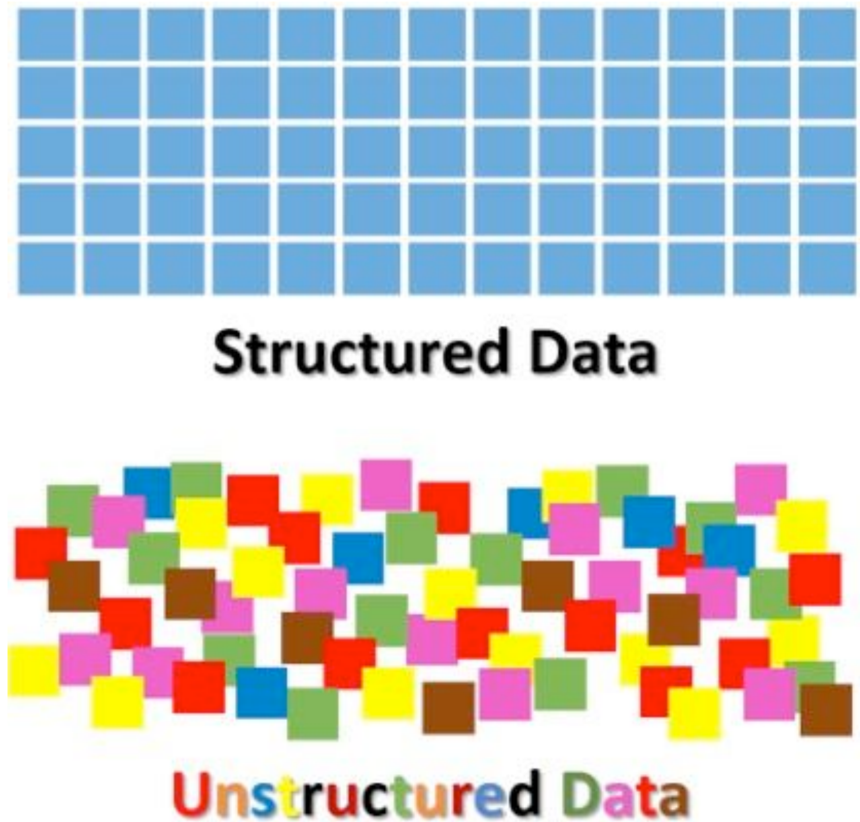
- Data:
  - We are overwhelmed with data. The amount of data in the world is ever more increasing and there is no end to it
  - According to IDC Research:
    - digital data will grow at a compound annual growth rate (CAGR) of 42% through 2020
    - in the 2010-2020 decade, the world's data will grow by 50X; i.e., from about 1ZB in 2010 to about 50ZB in 2020

Machine Learning



Source: https://www.eetimes.com/

Machine Learning



Graphical representations illustrate the difference between structured and unstructured data

Source: https://www.eetimes.com/ and Sherpa Software

Machine Learning

Multiplying Factor	SI Prefix	Scientific Notation	Name
1 000 000 000 000 000 000 000 000	Yotta (Y)	10^{24}	1 septillion
1 000 000 000 000 000 000 000	Zetta (Z)	10^{21}	1 sextillion
1 000 000 000 000 000 000	Exa (E)	10^{18}	1 quintillion
1 000 000 000 000 000	Peta (P)	10^{15}	1 quadrillion
1 000 000 000 000	Tera (T)	10^{12}	1 trillion
1 000 000 000	Giga (G)	10^{9}	1 billion
1 000 000	Mega (M)	10^{6}	1 million
1 000	kilo (k)	10^{3}	1 thousand
0 001	milli (m)	10^{-3}	1 thousandth
0 000 001	micro (u)	10^{-6}	1 millionth
0 000 000 001	nano (n)	10^{-9}	1 billionth
0 000 000 000 001	pico (p)	10^{-12}	1 trillionth
0 000 000 000 000 001	femto (f)	10^{-15}	1 quadrillionth
0 000 000 000 000 000 001	atto (a)	10^{-18}	1 quintillionth
0 000 000 000 000 000 000 001	zepto (z)	10^{-21}	1 sextillionth
0 000 000 000 000 000 000 000 001	yocto (y)	10^{-24}	1 septillionth

Metric prefixes defined at the 19th General Conference on Weights and Measures in 1991

Source: https://www.eetimes.com/

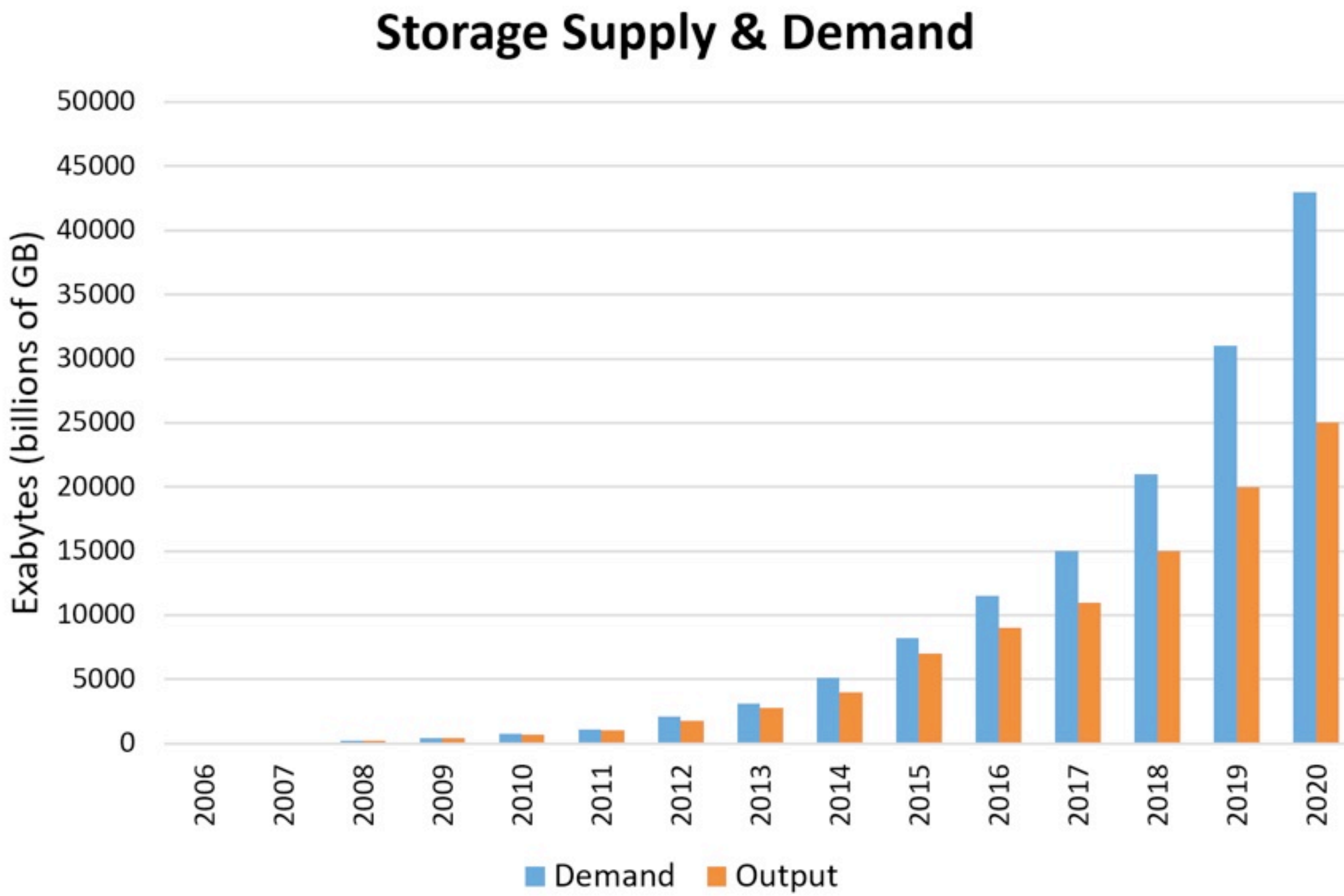
Machine Learning

Multiplying Factor	SI Prefix	Scientific Notation	Name
1 208 925 819 614 629 174 706 176	Yottabytes	2^{80}	1 septillion
1 180 591 620 717 411 303 424	Zettabytes	2^{70}	1 sextillion
1 152 921 504 606 846 976	Exabytes	2^{60}	1 quintillion
1 125 899 906 842 624	Petabytes	2^{50}	1 quadrillion
1 099 511 627 776	Terabytes	2^{40}	1 trillion
1 073 741 824	Gigabytes	2^{30}	1 billion
1 048 576	Megabytes	2^{20}	1 million
1 024	kilobytes	2^{10}	1 thousand

Examples of prefixes used to measure digital data with a binary system

Source: https://www.eetimes.com/

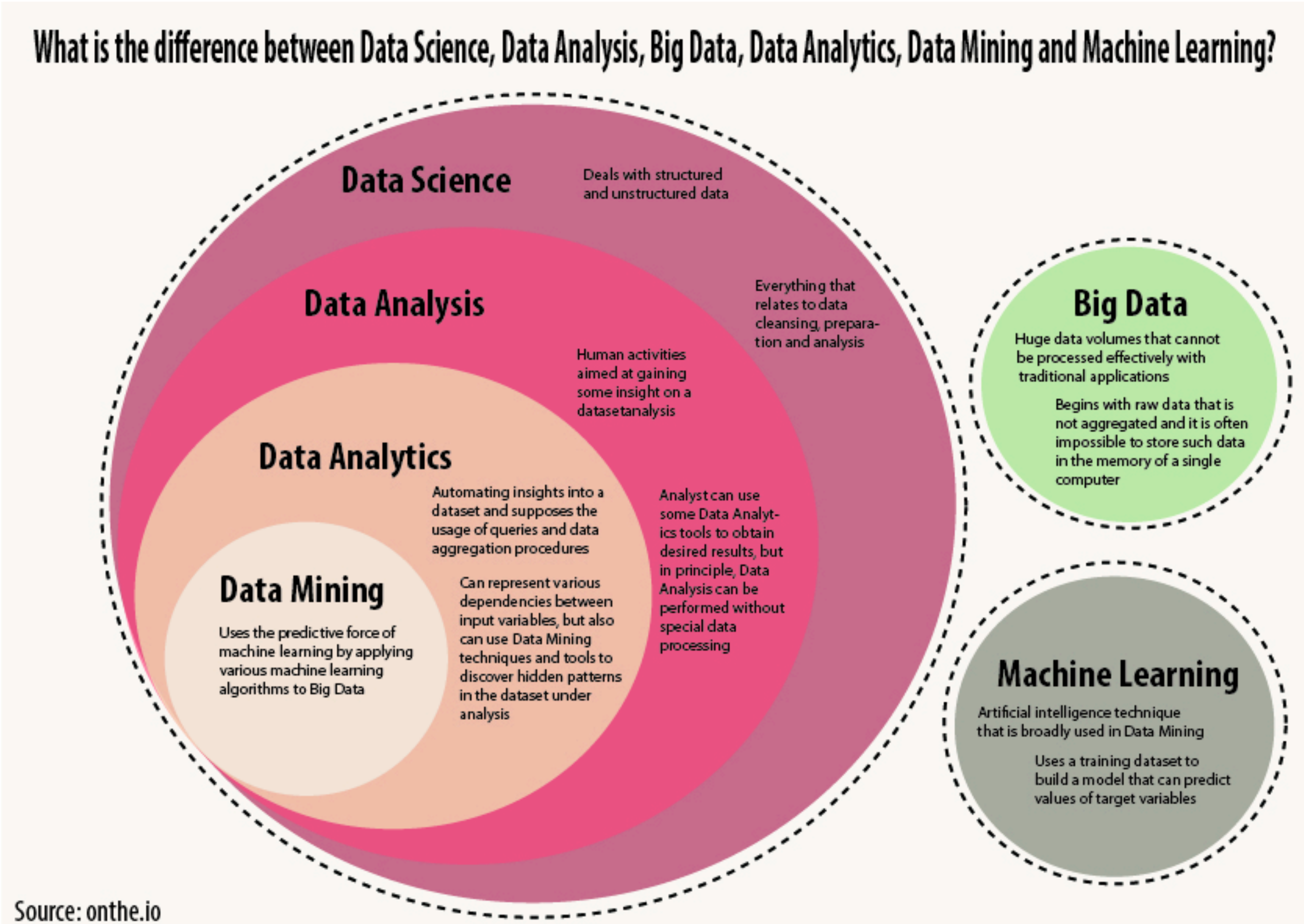
Machine Learning



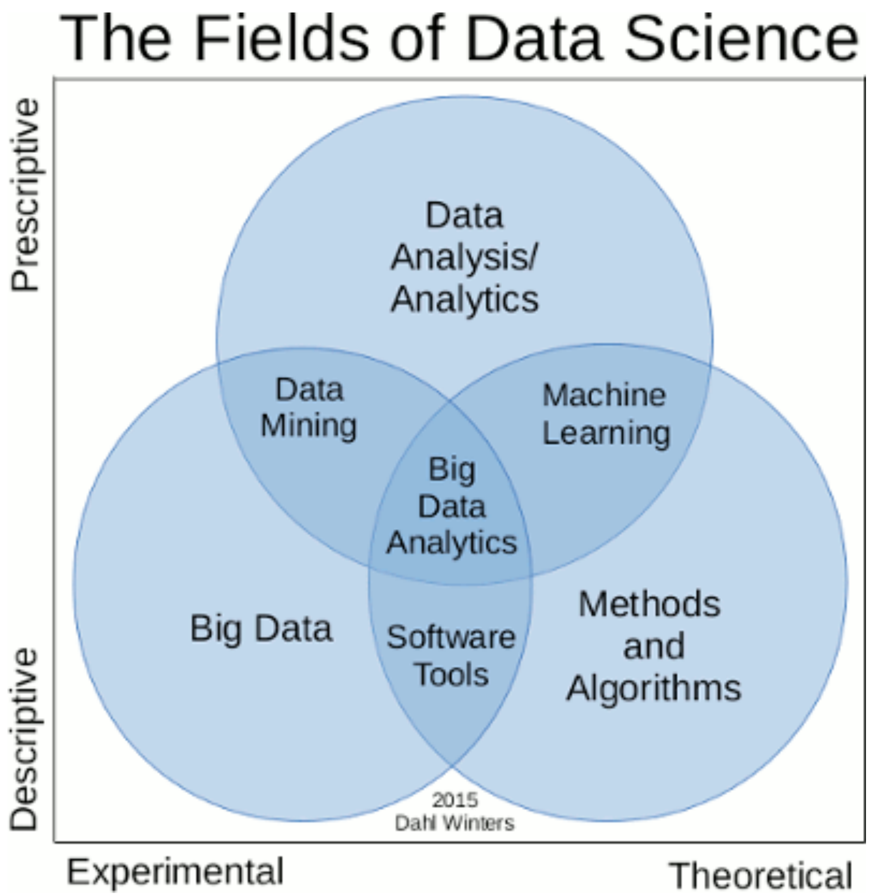
Storage supply and demand growth over two decades

Source: https://www.eetimes.com/

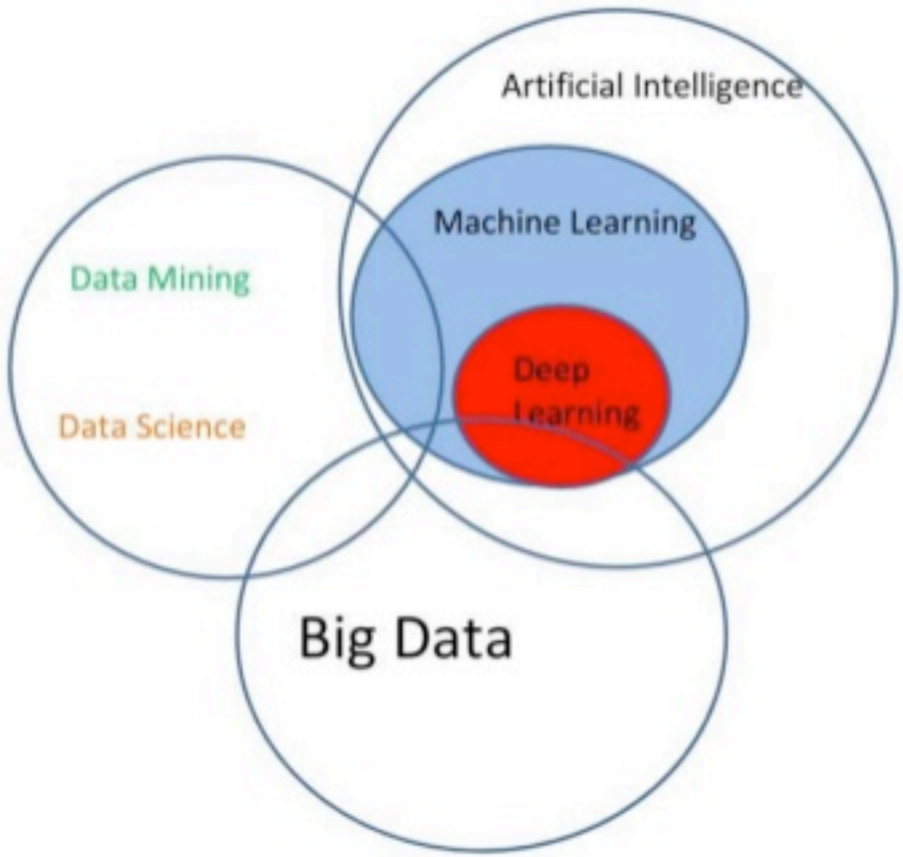
Machine Learning



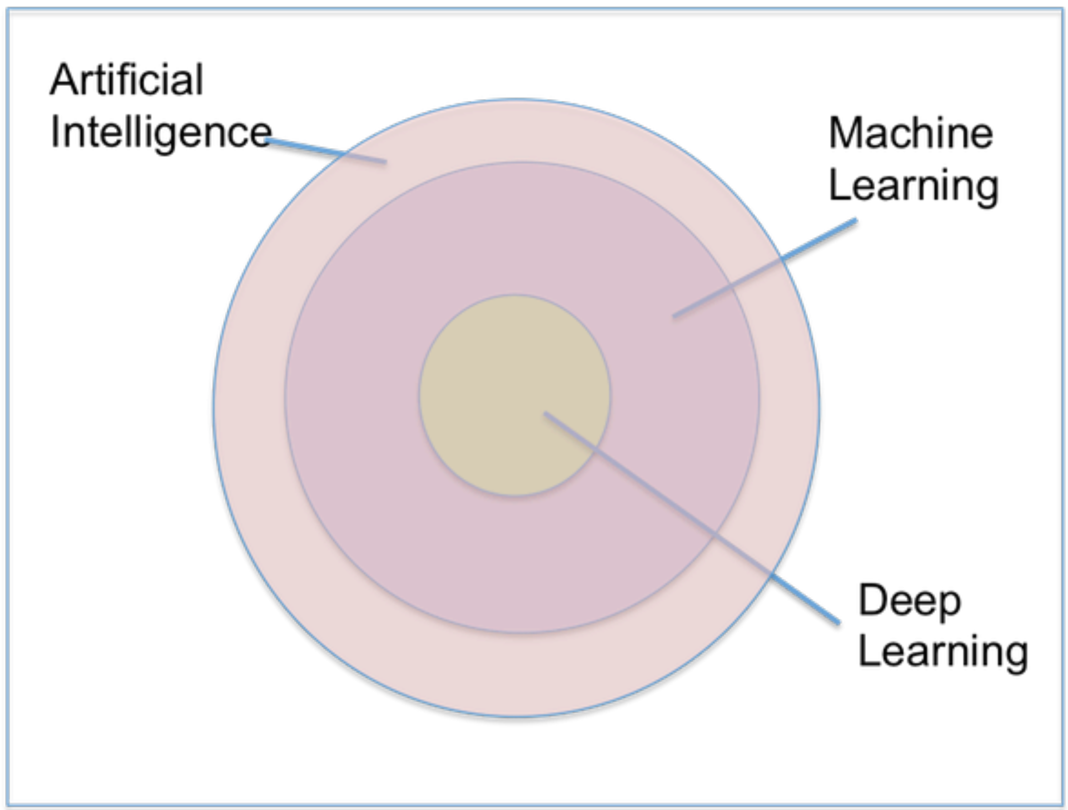
Machine Learning



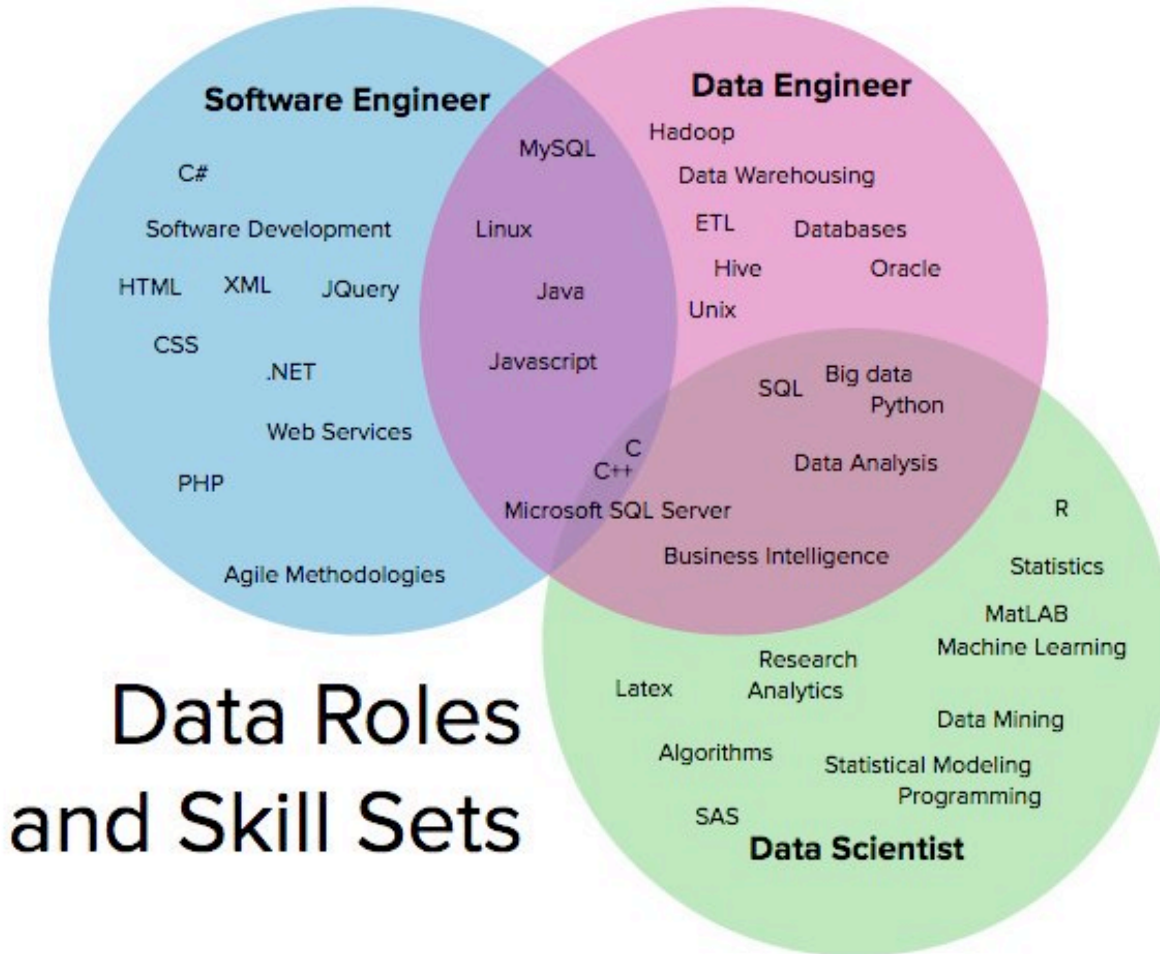
Machine Learning



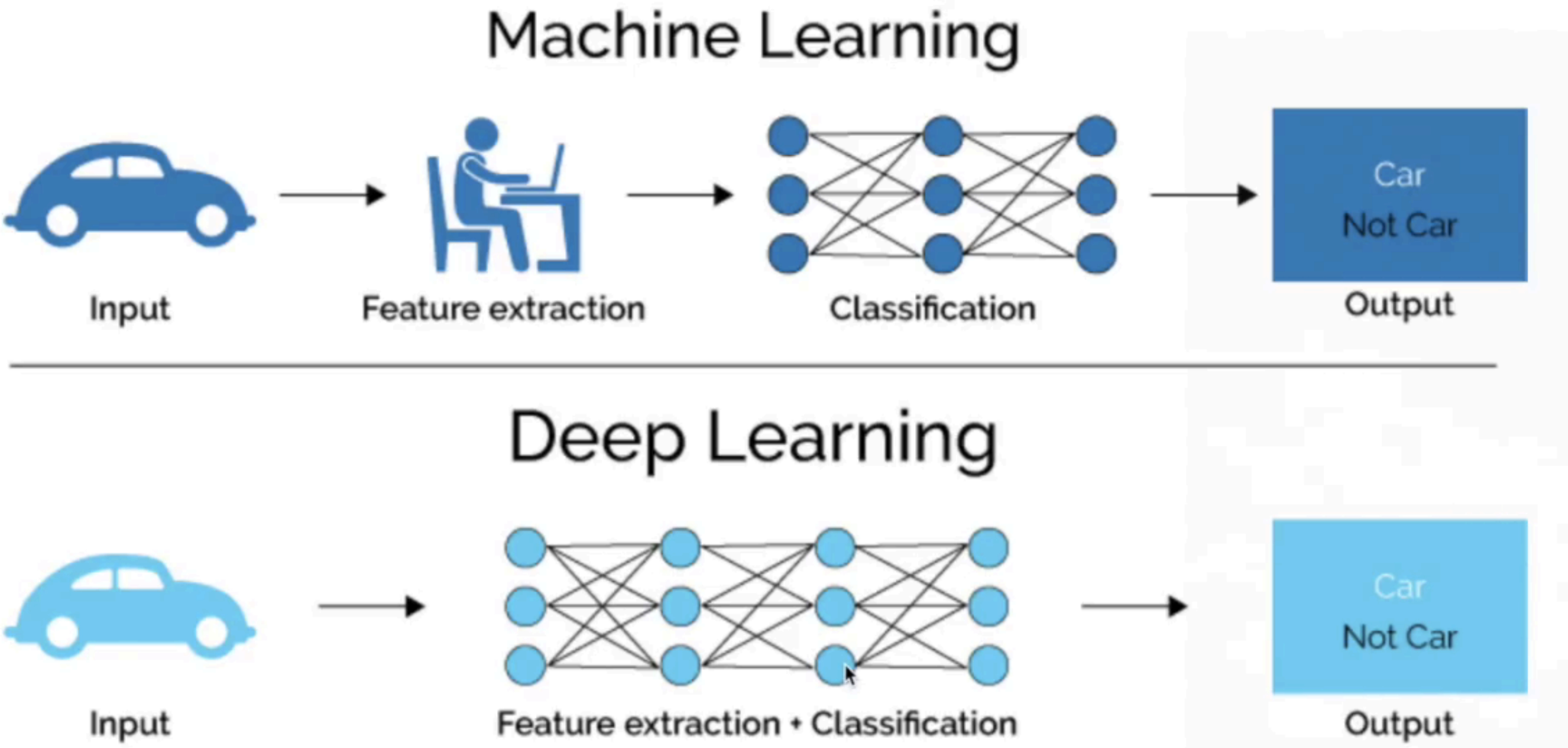
Machine Learning



Machine Learning



Machine Learning



Data Mining

- What is *Data Mining*?
  - Data mining is defined as the process of discovering patterns in data
  - Data mining is about solving problems by analyzing data already present in datasets / databases
  - In data mining, the data is stored electronically and the search is automated
  - It has been estimated that the amount of data stored in the world's databases doubles every 20 months

Machine Learning

- What is *machine learning*…? *(recall)*

- The dictionary defines "to learn" as:

  - To get knowledge of something by study, experience, or being taught
  - To become aware by information or from observation
  - To commit to memory
  - To be informed of or to ascertain
  - To receive instruction

Machine Learning

- What is *machine learning*?

- An operational definition can be formulated in the same way for learning:
  - Things learn when they change their behavior in a way that makes them perform better in the future

- This ties learning to *performance* rather than *knowledge*

- You can test learning by observing present behavior and comparing it with past behavior

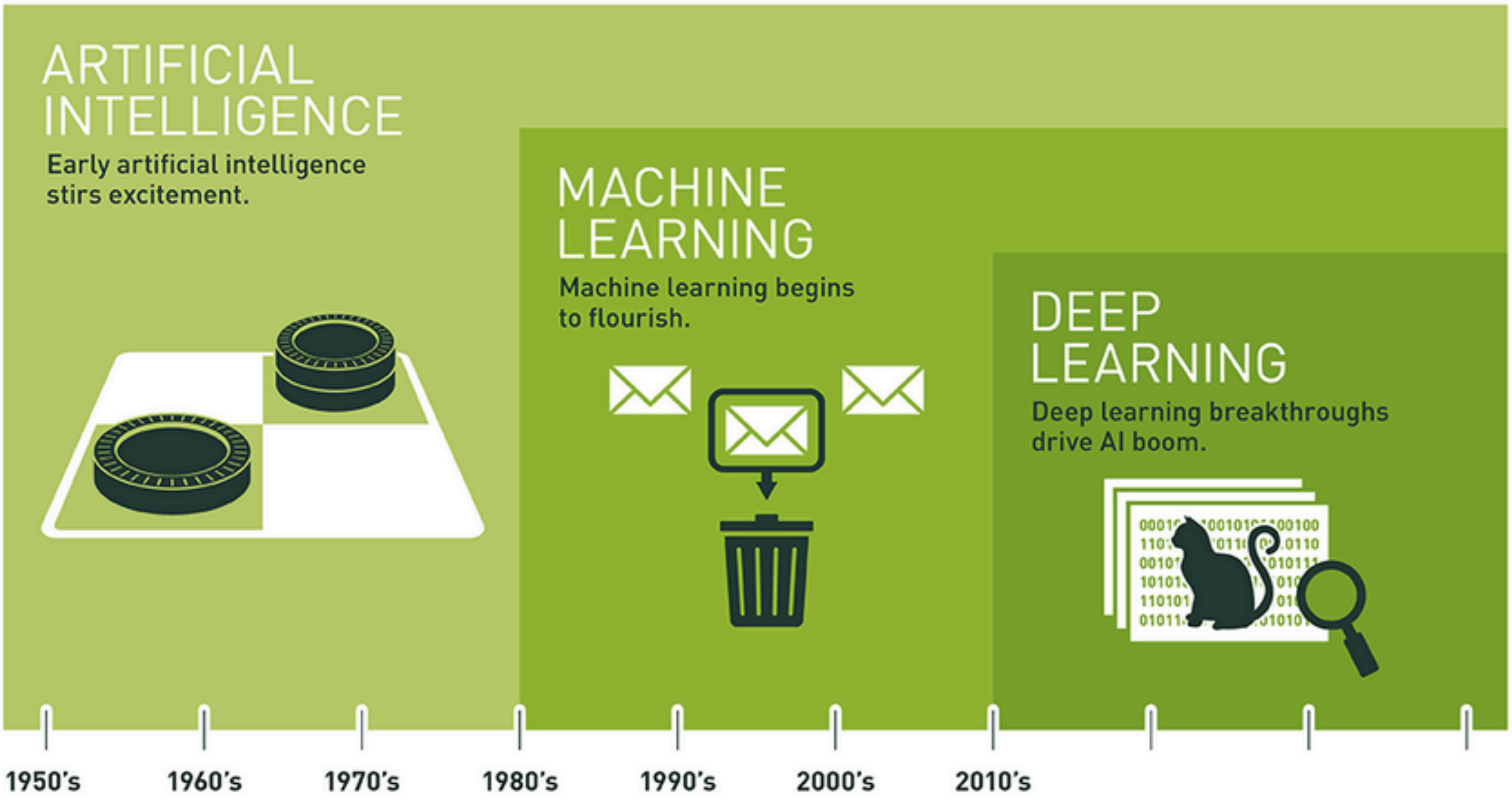
Machine Learning

- What is *machine learning*?

- … We therefore *choose* the word training to denote a mindless kind of learning
- We train animals and even plants, although it would be stretching the word a bit to talk of training objects such as slippers, which are not in any sense alive
- … But learning is different. Learning implies thinking and purpose.
- Something that learns has to do so intentionally
- That is why we wouldn't say that a vine has learned to grow around a trellis in a vineyard—we'd say it has been trained
- Learning without purpose is merely training

Machine Learning

- Deep Learning concept



source: nvidia.com

Machine Learning

- Describing Structural Patterns:
  - What are *structural patterns?*
  - How do we describe *structural patterns*?
  - What do they look like?

Structural descriptions

- Example: if-then rules

```
If tear production rate = reduced
   then recommendation = none
Otherwise, if age = young and astigmatic = no
   then recommendation = soft
```

Age	Spectacle prescription	Astigmatism	Tear production rate	Recommended lenses
Young	Myope	No	Reduced	None
Young	Hypermetrope	No	Normal	Soft
Pre-presbyopic	Hypermetrope	No	Reduced	None
Presbyopic	Myope	Yes	Normal	Hard
…	…	…	…	…

Machine Learning

- Describing Structural Patterns: *contact lens dataset*

Age	Spectacle Prescription	Astigmatism	Tear Production Rate	Recommended Lenses
young	myope	no	reduced	none
young	myope	no	normal	soft
young	myope	yes	reduced	none
young	myope	yes	normal	hard
young	hypermetrope	no	reduced	none
young	hypermetrope	no	normal	soft
young	hypermetrope	yes	reduced	none
young	hypermetrope	yes	normal	hard
pre-presbyopic	myope	no	reduced	none
pre-presbyopic	myope	no	normal	soft
pre-presbyopic	myope	yes	reduced	none
pre-presbyopic	myope	yes	normal	hard
pre-presbyopic	hypermetrope	no	reduced	none
pre-presbyopic	hypermetrope	no	normal	soft
pre-presbyopic	hypermetrope	yes	reduced	none
pre-presbyopic	hypermetrope	yes	normal	none
presbyopic	myope	no	reduced	none
presbyopic	myope	no	normal	none
presbyopic	myope	yes	reduced	none
presbyopic	myope	yes	normal	hard
presbyopic	hypermetrope	no	reduced	none
presbyopic	hypermetrope	no	normal	soft
presbyopic	hypermetrope	yes	reduced	none
presbyopic	hypermetrope	yes	normal	none

Machine Learning

- Describing Structural Patterns: *contact lens dataset*

Age	Spectacle Prescription	Astigmatism	Tear Production Rate	Recommended Lenses
young	myope	no	reduced	none
young	myope	no	normal	soft
young	myope	yes	reduced	none
young	myope	yes	normal	hard
young	hypermetrope	no	reduced	none
young	hypermetrope	no	normal	soft
young	hypermetrope	yes	reduced	none
young	hypermetrope	yes	normal	hard
pre-presbyopic	myope	no	reduced	none
pre-presbyopic	myope	no	normal	soft
pre-presbyopic	myope	yes	reduced	none
pre-presbyopic	myope	yes	normal	hard
pre-presbyopic	hypermetrope	no	reduced	none
pre-presbyopic	hypermetrope	no	normal	soft
pre-presbyopic	hypermetrope	yes	reduced	none
pre-presbyopic	hypermetrope	yes	normal	none
presbyopic	myope	no	reduced	none
presbyopic	hypermetrope	no	normal	soft
presbyopic	hypermetrope	yes	reduced	none
presbyopic	hypermetrope	yes	normal	none

```
If tear production rate = reduced then recommendation = none
Otherwise, if age = young and astigmatic = no then
recommendation = soft
```

Machine Learning

- Describing Structural Patterns:

  - ***Structural descriptions*** do not necessarily need to be expressed as rules such as these

  - ***Decision trees***, specify the sequences of decisions that need to be made along with the resulting recommendation, are another popular means of expression

Machine Learning

- Describing Structural Patterns:

  - **The rules** do not generalize from the data; they merely summarize it

  - In most learning situations, the set of examples given as input is far from complete, and part of the job is to generalize to other, new examples

  So:

  - Imagine omitting some of the rows in a table for which the tear production rate is *reduced* and still coming up with the rule ..

    .. *(see next slide)*

Machine Learning

- Describing Structural Patterns: *contact lens dataset*

Age	Spectacle Prescription	Astigmatism	Tear Production Rate	Recommended Lenses
young	myope	no	reduced	none
young	myope	no	normal	soft
young	myope	yes	reduced	none
young	myope	yes	normal	hard
young	hypermetrope	no	reduced	none
young	hypermetrope	no	normal	soft
young	hypermetrope	yes	reduced	none
young	hypermetrope	yes	normal	hard
pre-presbyopic	myope	no	reduced	none
pre-presbyopic	myope	no	normal	soft
pre-presbyopic	myope	yes	reduced	none
pre-presbyopic	myope	yes	normal	hard
pre-presbyopic	hypermetrope	no	reduced	none

```
If tear production rate = reduced then recommendation = none
```

pre-presbyopic	hypermetrope	yes	normal	none
presbyopic	myope	no	reduced	none

```
If tear production rate = reduced then recommendation = none
Otherwise, if age = young and astigmatic = no then
recommendation = soft
```

presbyopic	hypermetrope	no	normal	soft
presbyopic	hypermetrope	yes	reduced	none
presbyopic	hypermetrope	yes	normal	none

Machine Learning

- Describing Structural Patterns:
  - First, this would generalize to the missing rows
  - Second, values are specified for all the features in all the examples. Real-life datasets invariably contain values of some features, for some reason or other, are unknown—for example, measurements were not taken or were lost
  - Third, the preceding rules classify the examples correctly, whereas often, because of errors or noise in the data, misclassifications occur even on the data that is used to create the classifier

Machine Learning

attributes = features = predictors

examples = instances = variables

sunny,85,85,FALSE	no
sunny,80,90,TRUE	no
overcast,83,86,FALSE	yes
rainy,70,96,FALSE	yes
rainy,68,80,FALSE	yes
rainy,65,70,TRUE	no
overcast,64,65,TRUE	yes
sunny,72,95,FALSE	no
sunny,69,70,FALSE	yes
rainy,75,80,FALSE	yes
sunny,75,70,TRUE	yes
overcast,72,90,TRUE	yes
overcast,81,75,FALSE	yes
rainy,71,91,TRUE	no

class = outcome

Applied Machine Learning

- Lecture 1 recap:
  - Projects, assignments and requirements
  - Data Mining vs Machine Learning
  - What is Machine Learning?
  - Supervised vs. Unsupervised Learning
  - Supervised learning methods: (mostly used in this class)
    - kNN – one of the earliest algorithms, using distance between observations
    - Naive Bayes – using Bayes theorem, estimating the probability of a class label, used in text analysis
    - Decision trees – random forests, gradient boosted trees, - one of the most useful methods
    - Regression (linear and logistic) – used in finance and in the business world
    - Regularization – estimating sparse models (more attributes than observations)
    - Optimization using gradient descent – how to estimate parameters in models like NN, just like backpropagation
    - Neural Networks with backpropagation – introducing nonlinearity for solving more complex problems
  - Unsupervised learning methods:
    - Clustering – k-means - *hard clustering*, clustering
    - PCA – dimensionality reduction (ex. healthcare: disease prediction)
    - EM (Expectation Maximization) algorithms - *soft clustering* and Mixture Models – discover the mean and covariance of each cluster using probability assuming Gaussian distribution
  - Overfitting – making the model too complex to be applied in the real world
  - Data source – Kaggle.com