



Supervised Learning, Neural Networks and Applied Computer Vision- Day 1

Professor Eyad Elyan

July 26, 2022

① Overview

The Right Questions

Why ML and Algorithms?

Think Data

Recommended Readings

② Supervised Learning

Problem / Data collection

Solution

Model/s

Evaluation / Testing

③ Conclusion

① Overview

The Right Questions

Why ML and Algorithms?

Think Data

Recommended Readings

② Supervised Learning

Problem / Data collection

Solution

Model/s

Evaluation / Testing

③ Conclusion

The Right Questions



The Right Questions

- Oren Etzioni bought a plane ticket months before his departure date
- The **assumption** is that the earlier you buy your plane tickets, the cheapest the price is

The Right Questions

- Oren Etzioni bought a plane ticket months before his departure date
- The **assumption** is that the earlier you buy your plane tickets, the cheapest the price is
- However, on the flight date, he discovered that many passengers who bought their tickets recently (few days before the flight) actually paid considerably less than him

The Right Questions

- Oren Etzioni is a computer scientist
- The previous story raised a big question for him: He wanted to figure out a way for people to know if a ticket price they see online is a good deal or not?

The Right Questions

- Remember that the tickets prices vary wildly, and perhaps only airlines know the factors that influence the prices of the tickets

The Right Questions

- He decided that he doesn't need to know what factors cause an increase or decrease in the prices of the airline tickets. All he needed to know is **if a ticket price they see online is a good deal or not?**

The Right Questions

- The question raised by Etzioni: *if a ticket price you see online is a good deal or not?* is technically called a supervised binary classification problem
- Classification problems are well known and common in AI and machine Learning

The Right Questions

- Notice that the problem has been narrowed down to simple question (buy or not buy)
- Very well defined problem

The Right Questions

- Oren Etzioni managed to scrape data from the web (travel agents) for 12,000 price observations over a period of 41 days
- A predictive model was then built using the above dataset
- The results showed good savings for the 'simulated passengers'

The Right Questions

- Notice that the model built by Etzioni had no understanding of *why* the prices are up or down
- And it didn't care about the various input parameters (variables), e.g. unsold seats, seasonality
- All the model is doing is inform passengers to *'buy or not to buy'*

The Right Questions

- This simple project evolved into a successful startup and was called *Farecast*
- The solution was developed into a system capable of making prediction based on every seat on every flight for most routes in the US
- The amount of data used by Farecast now uses more than 200 billion flight-price records to make its prediction

The Right Questions

- *Farecast* was acquired by Microsoft in 2008 for \$115 million
- An exemplary system:

The Right Questions

- *Farecast* was acquired by Microsoft in 2008 for \$115 million
- An exemplary system:
 - ① Well-defined problem

The Right Questions

- *Farecast* was acquired by Microsoft in 2008 for \$115 million
- An exemplary system:
 - ① Well-defined problem
 - ② Replaced a tradition or assumption with a fully data-driven solution using machine learning

The Right Questions

- *Farecast* was acquired by Microsoft in 2008 for \$115 million
- An exemplary system:
 - ① Well-defined problem
 - ② Replaced a tradition or assumption with a fully data-driven solution using machine learning
 - ③ Simple solution that requires basic tools, limited data and inexpensive resources

The Right Questions

- *Farecast* was acquired by Microsoft in 2008 for \$115 million
- An exemplary system:
 - ① Well-defined problem
 - ② Replaced a tradition or assumption with a fully data-driven solution using machine learning
 - ③ Simple solution that requires basic tools, limited data and inexpensive resources
 - ④ Quick gains

The Right Questions

- *Farecast* was acquired by Microsoft in 2008 for \$115 million
- An exemplary system:
 - ① Well-defined problem
 - ② Replaced a tradition or assumption with a fully data-driven solution using machine learning
 - ③ Simple solution that requires basic tools, limited data and inexpensive resources
 - ④ Quick gains
 - ⑤ Evolved into more sophisticated system

The Right Questions

- *Farecast* was acquired by Microsoft in 2008 for \$115 million
- An exemplary system:
 - ① Well-defined problem
 - ② Replaced a tradition or assumption with a fully data-driven solution using machine learning
 - ③ Simple solution that requires basic tools, limited data and inexpensive resources
 - ④ Quick gains
 - ⑤ Evolved into more sophisticated system
- The question is what was the most critical **factor/s** for this success? (data, problem, tools, methods, machine learning, etc.)

Overview

① Overview

The Right Questions

Why ML and Algorithms?

Think Data

Recommended Readings

② Supervised Learning

Problem / Data collection

Solution

Model/s

Evaluation / Testing

③ Conclusion

① Overview

The Right Questions

Why ML and Algorithms?

Think Data

Recommended Readings

② Supervised Learning

Problem / Data collection

Solution

Model/s

Evaluation / Testing

③ Conclusion

Algorithms vs Humans

A bat and a ball cost **\$1.10**.

Algorithms vs Humans

A bat and a ball cost **\$1.10**.



costs one dollar more than the
How much does the ball cost?¹



¹Thinking Fast and Slow by Daniel Kahneman

Algorithms vs Humans

A bat and a ball cost **\$1.10**.



costs one dollar more than the
How much does the ball cost?¹



Answer

The ball cost **10 cents**

¹Thinking Fast and Slow by Daniel Kahneman

Algorithms vs Humans

A bat and a ball cost **\$1.10**.



costs one dollar more than the
How much does the ball cost?¹



Answer

The ball cost **10 cents** **X**

¹Thinking Fast and Slow by Daniel Kahneman

Algorithms vs Humans


A bat and a ball cost \$1.10.



costs one dollar more than the
How much does the ball cost?¹



Answer

The ball cost **10 cents** 

ball cost **5 cents** 

¹Thinking Fast and Slow by Daniel Kahneman

Why Machine Learning

Three main related reasons that drive the urgent need for AI and ML-driven solutions:

- ① Exponential increase in data
- ② Significant progress in AI, Machine Learning and Deep Learning
- ③ Technology readiness (hardware, sensors, IoT/ software, etc)

Recommended Reading

Jonny Holmström, From AI to digital transformation: [The AI readiness framework](#), Business Horizons, Volume 65, Issue 3, 2022, Pages 329-339, ISSN 0007-6813, <https://doi.org/10.1016/j.bushor.2021.03.006>

Why Machine Learning

Exponential Increase in Data

	No	Metric	Abbr	Value
1	1	kilobyte	K	1024
2	1	megabyte	M	1048576
3	1	gigabyte	G	1073741824
4	1	terabyte	T	1099511627776
5	1	Petabyte	P	1125899906842624
6	1	Exabyte	E	1152921504606846976
7	1	Zettabyte	Z	1180591620717411303424

²Source: PwC's Global Artificial Intelligence Study

Why Machine Learning

Exponential Increase in Data

No	Metric	Abbr	Value
1	kilobyte	K	1024
2	megabyte	M	1048576
3	gigabyte	G	1073741824
4	terabyte	T	1099511627776
5	Petabyte	P	1125899906842624
6	Exabyte	E	1152921504606846976
7	Zettabyte	Z	1180591620717411303424

Global Datasphere will grow from **33 Zettabytes** (ZB) in 2018 to 175 ZB by 2025 (IDC White Paper), (ZB = Billion TB) ²

²Source: PwC's Global Artificial Intelligence Study

Why Machine Learning

Significant progress in AI Algorithms

- Regression (Linear/ Logistic)
- Random Forest
- Kernel Methods (SVM)
- Gradient Boosting
- ...
- 179 ML algorithm ³

- **Deep Learning**

Common tasks in Industry

- Regression
- Classification
- Prediction / Forecasting

³Fernandez-Delgado, M., Cernadas, E., Barro, S., & Amorim, D. (2014). [Do we need hundreds of classifiers to solve real world classification problems?](#). The Journal of Machine Learning Research, 15(1), 3133-3181.

Why Machine Learning?

- The cloud (Google Colab)
- Cloud (AWS)
- GPU enabled machines
- Cheap sensors
- Super Computers (GPUs)



DGX 1: 1.3 Billion image per day

① Overview

The Right Questions

Why ML and Algorithms?

Think Data

Recommended Readings

② Supervised Learning

Problem / Data collection

Solution

Model/s

Evaluation / Testing

③ Conclusion

Supervised Machine Learning

- Main branch of Machine Learning (most ML problems are supervised learning problems)
- In simple terms it means giving the computers the ability to learn from experience (past data observations, historical data, etc.)

Supervised Machine Learning

- Main branch of Machine Learning (most ML problems are supervised learning problems)
- In simple terms it means giving the computers the ability to learn from experience (past data observations, historical data, etc.)
- The easiest way to think Supervised ML, is to think in terms of **input data** and **corresponding output or outcome**

Supervised Machine Learning

- Main branch of Machine Learning (most ML problems are supervised learning problems)
- In simple terms it means giving the computers the ability to learn from experience (past data observations, historical data, etc.)
- The easiest way to think Supervised ML, is to think in terms of **input data** and **corresponding output or outcome**
- Can we think of an example related to Students Performance at Birzeit University?

Supervised Machine Learning

- Main branch of Machine Learning (most ML problems are supervised learning problems)
- In simple terms it means giving the computers the ability to learn from experience (past data observations, historical data, etc.)
- The easiest way to think Supervised ML, is to think in terms of **input data** and **corresponding output or outcome**
- Can we think of an example related to Students Performance at Birzeit University? Predict student's final grade in Physics. All is needed a collection of data related to previous students (e.g. age, gender, residence, number of checkpoints crossed, . . . , **final grade**)

Think Data

Supervised Learning from different types of data:

- Structured data
- Text
- 2D Images
- Videos
- 3D Images
- Multi-modal data (2D & 3D, etc..)

Think Data

Supervised Learning from different types of data:

- Structured data
- Text
- 2D Images
- Videos
- 3D Images
- Multi-modal data (2D & 3D, etc..)

Motivation

- To achieve faster, more accurate, safer practices,
- Overcome humans cognitive bias by relying on data-driven and AI-based solutions

Think Data

- **Classification:** assigning a label (outcome) to each instance (data instance) depending on the values of a set of attributes (*Predicting the patient's diagnosis based on attributes related to symptoms, medical history, age, etc.*)

Think Data

- **Classification:** assigning a label (outcome) to each instance (data instance) depending on the values of a set of attributes (*Predicting the patient's diagnosis based on attributes related to symptoms, medical history, age, etc.*)

Think Data

- **Regression:** estimating the numerical value of a measurement (attribute), based on the history of values assigned to this attribute and other affecting measurements (*Estimating the house price based on the values of prices recorded in the previous month, along with the features of each house e.g., area, location, and number of bedrooms*)

① Overview

The Right Questions

Why ML and Algorithms?

Think Data

Recommended Readings

② Supervised Learning

Problem / Data collection

Solution

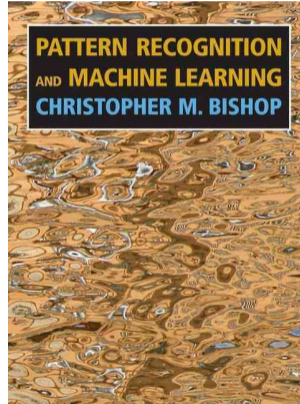
Model/s

Evaluation / Testing

③ Conclusion

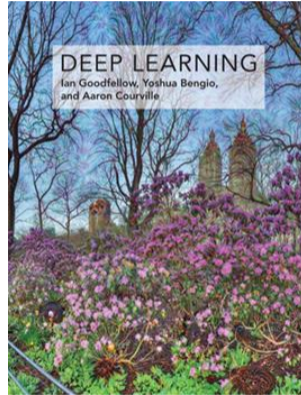
Recommended Readings (Theory)

- Excellent resource for Machine Learning



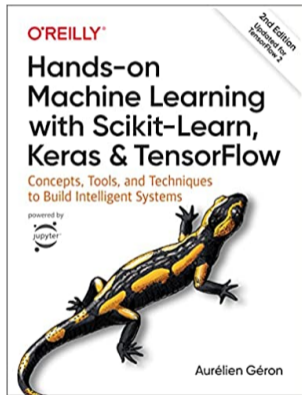
Recommended Readings (Theory)

- Excellent resource for Deep Learning



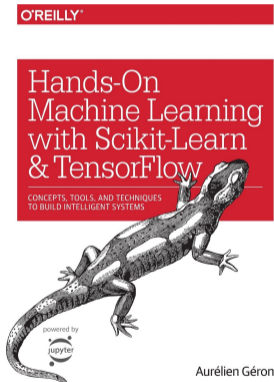
Recommended Readings (Applied)

- Very practical and applied resource for hands-on and applied machine learning and Deep Learning



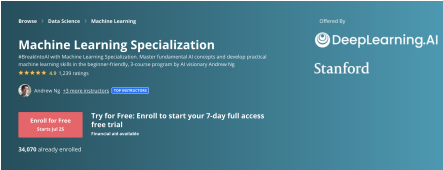
Recommended Readings (Applied)

- Available online



Recommended Courses

- Machine Learning Specialization (Andrew Ng)⁴
- Excellent balance between theory and practice

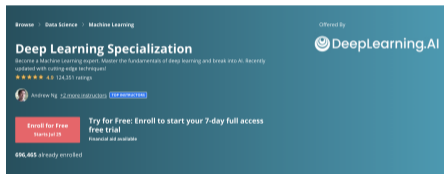


The screenshot shows the course page for 'Machine Learning Specialization' on Coursera. The page is teal and white. At the top, it says 'Browse > Data Science > Machine Learning'. The course title is 'Machine Learning Specialization' and it is offered by 'DeepLearning.AI' and 'Stanford'. The description reads: 'Get introduced to Machine Learning Specialization. Master fundamental AI concepts and develop practical machine learning skills in the beginner-friendly, 3-course program by AI visionary Andrew Ng'. It has a 4.8 rating from 1,239 ratings. The instructor is Andrew Ng, with 15,036,153 students. There are two enrollment options: 'Enroll for Free starts Jul 25' and 'Try for Free: Enroll to start your 7-day full access free trial Financial aid available'. At the bottom, it says '34,670 already enrolled'.

⁴<https://www.coursera.org/specializations/machine-learning-introduction>

Recommended Courses

- Deep Learning Specialization (Andrew Ng)⁵
- A great six courses on DL with excellent balance between theory and practice

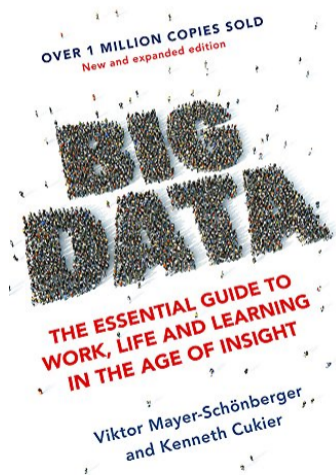


The screenshot shows the course page for 'Deep Learning Specialization' on Coursera. The page has a teal header with navigation links for 'Browse', 'Data Science', and 'Machine Learning'. The course title 'Deep Learning Specialization' is prominently displayed, along with a brief description: 'Become a Machine Learning expert. Master the fundamentals of deep learning and break into AI. Recently updated with cutting-edge techniques.' Below the title, there is a star rating of 4.9 and a note that 124,351 students have rated the course. The instructor's name, Andrew Ng, is listed with a 'VIEW INSTRUCTOR' link. Two enrollment options are shown: 'Enroll for Free' (starting Jan 28) and 'Try for Free: Enroll to start your 7-day full access free trial' (financial aid available). At the bottom, it states that 696,465 students are already enrolled. The DeepLearning.AI logo is visible in the top right corner.

⁵<https://www.coursera.org/specializations/deep-learning>

Recommended Readings - Books

- Highly recommended read
- Gentle and non-technical introduction to the power of using data in the decision making process and how machine learning was applied to successfully to various real-world scenarios



Plan

① Overview

The Right Questions

Why ML and Algorithms?

Think Data

Recommended Readings

② Supervised Learning

Problem / Data collection

Solution

Model/s

Evaluation / Testing

③ Conclusion

① Overview

The Right Questions

Why ML and Algorithms?

Think Data

Recommended Readings

② Supervised Learning

Problem / Data collection

Solution

Model/s

Evaluation / Testing

③ Conclusion

Problem Definition

A company is investing in advertising in *TV*, *Newspaper* and *Radio*. The company, is trying to understand the relation between the investments in these areas and the total sales, ...

Data

TV	Radio	Newspaper	Sales
191.10	28.70	18.20	17.30
286.00	13.90	3.70	15.90
18.70	12.10	23.40	6.70
39.50	41.10	5.80	10.80
75.50	10.80	6.00	9.90
166.80	42.00	3.60	19.60
38.20	3.70	13.80	7.60
94.20	4.90	8.10	9.70
177.00	9.30	6.40	12.80
283.60	42.00	66.20	25.50
232.10	8.60	8.70	13.40

Data

TV	Radio	Newspaper	Sales
191.10	28.70	18.20	17.30
286.00	13.90	3.70	15.90
18.70	12.10	23.40	6.70
39.50	41.10	5.80	10.80
75.50	10.80	6.00	9.90
166.80	42.00	3.60	19.60
38.20	3.70	13.80	7.60
94.20	4.90	8.10	9.70
177.00	9.30	6.40	12.80
283.60	42.00	66.20	25.50
232.10	8.60	8.70	13.40

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & \vdots & \dots & x_{mn} \end{bmatrix}, Y = \begin{bmatrix} y_1 \\ \dots \\ y_m \end{bmatrix}$$

How much increase in Sales units if spending a certain amount of money in TV, Radio, and Newspaper ads?

① Overview

The Right Questions

Why ML and Algorithms?

Think Data

Recommended Readings

② Supervised Learning

Problem / Data collection

Solution

Model/s

Evaluation / Testing

③ Conclusion



Adevrtising-Sales

This is first streamlit app. It will be used to explore the Advertising dataset and perform predictions of sales based on simple linear model. It must be noted that **linear** regression models are not best choice for this problem

Use the sidebar controls to change the spending values on TV, Radio, and Newspaper Advertising, and also to control how many rows of the dataframe you want to show.

Explore the Dataset

List of **2** Records from Data Frame **df**

	TV	Radio	Newspaper	Sales
0	230.1000	37.8000	69.2000	22.1000
1	44.5000	39.3000	45.1000	10.4000

The Data Frame has **200** Rows, and **4** Columns

2 Rows from the Data Frame are visible

Make Prediction

Spending 219 units on **TV**, 120 units on **Radio**, and 161 on **Newspaper** Advertising, will generate increase in sales by 35.418 units





Adevrtising-Sales

This is first streamlit app. It will be used to explore the Advertising dataset and perform predictions of sales based on simple linear model. It must be noted that **linear** regression models are not best choice for this problem

Use the sidebar controls to change the spending values on TV, Radio, and Newspaper Advertising, and also to control how many rows of the dataframe you want to show.

Explore the Dataset

List of **2** Records from Data Frame **df**

	TV	Radio	Newspaper	Sales
0	230.1000	37.8000	69.2000	22.1000
1	44.5000	39.3000	45.1000	10.4000

The Data Frame has **200** Rows, and **4** Columns

2 Rows from the Data Frame are visible

Make Prediction

Spending 125 units on **TV**, 411 units on **Radio**, and 390 on **Newspaper** Advertising, will generate increase in sales by 85.741 units



① Overview

The Right Questions

Why ML and Algorithms?

Think Data

Recommended Readings

② Supervised Learning

Problem / Data collection

Solution

Model/s

Evaluation / Testing

③ Conclusion

Chose Your Model

- Linear/ Logistic Regression
- K Nearest Neighbor (KNN)
- Ensemble Methods (i.e Random Forest, Gradient Boosting, Extreme Gradient Boosting, etc. . .)
- Kernel Methods (i.e. SVM)
- Neural Network
- . . .

Regression

Regression analysis is one of the most important fields in statistics. There are many regression methods available. [Linear Regression](#) is one of the simplest models.

Linear Regression

Linear regression is a supervised learning technique, where we construct a linear model to capture the relation between a response variable and a set of independent variables (predictors). In our example about the Advertising company:

- Y dependent (response) variable is the *Sales*
- X_j independent (predictor, explanatory) variables (spendings on *TV*, *Radio* and *Newspaper Ads*)

Linear Regression

Simple linear regression can be applied when we want to model the response variable, Y , in terms of a single predictor variable, X , when we have n pairs of observations:

$$(X_1, Y_1), \dots, (X_n, Y_n)$$

Questions to answer

- Is X associated with Y ?
- What is the strength of any linear association between X and Y ?
- What is the nature of any relationship between X and Y ?
- How precise is the estimate of any relationship?
- What is the predicted value of Y for a new observed value of X ?
- How precise are predicted values of Y ?
- Is a linear model appropriate?

Definition

$\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 X$ is an estimate of the regression equation which best describes the relationship between X and Y in the population.

$\hat{\beta}_0$ and $\hat{\beta}_1$ are estimates of the unknown population parameters in the theoretical model

$$Y = \beta_0 + \beta_1 X + \epsilon$$

where, $\beta_0 + \beta_1 X$ is the straight line that best describes the relationship between X and Y in the population;

ϵ is a random error term which accounts for the fact that the points are scattered about the straight line.

Training

Back to our advertising example. Suppose we want only to consider TV , in other words we want to predict how much sales units increase, if we spent X units of money on TV

Our model here, $\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 X$.

$\hat{\beta}_0$ and $\hat{\beta}_1$ are called the coefficients, or sometimes (weights), and we want to **train** our model to learn these coefficients

To train our model, we use the data collected / provided by the company as we will be seen in the Lab.

Multiple Linear Regression

- We extend our analysis of the advertising data in order to accommodate the remaining two additional predictors (Radio, Newspaper)?

$$\hat{Y} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_k + \epsilon$$

- In our example, the model will become

$$\hat{Sales} = \beta_0 + \beta_1 TV + \beta_2 Radio + \beta_3 Newspaper + \epsilon$$

- Notice, that when we **train** the model, we can always evaluate its quality by computing the error or the difference between the predicted sales values (\hat{Sales}), and the ground truth (actual sales value in the dataset)

① Overview

The Right Questions

Why ML and Algorithms?

Think Data

Recommended Readings

② Supervised Learning

Problem / Data collection

Solution

Model/s

Evaluation / Testing

③ Conclusion

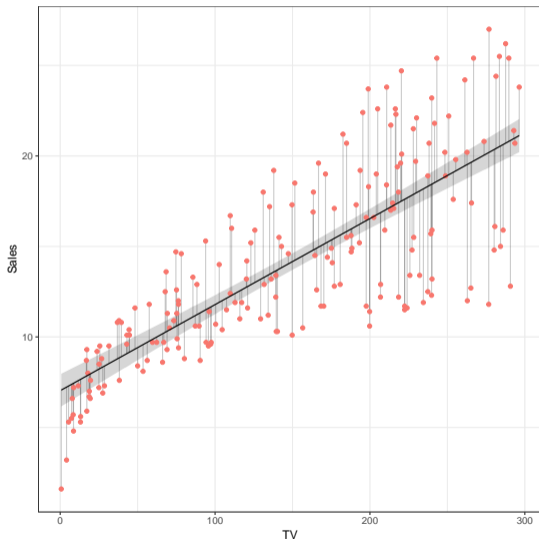
Evaluating the Model

- $\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 X$ is an estimate of the regression equation which best describes the relationship between X and Y in the dataset
- $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$ be the prediction for Y based on the i^{th} value of X
- Then $e_i = y_i - \hat{y}_i$ is the i^{th} residual
- Residual Sum of Squares (RSS) is defined as

$$RSS = e_1^2 + e_2^2 + \dots + e_n^2 \quad (1)$$

- Least square approach we chose β_0, β_1 to minimise RSS

Evaluating the Model (RSS)



$$RSS = \sum_{i=1}^n (\hat{y}_i - y_i)^2$$

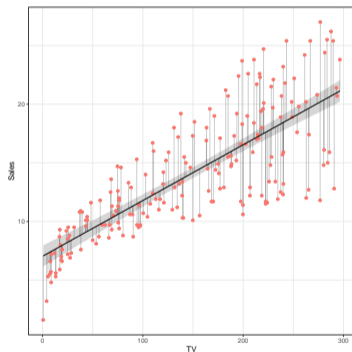
Code and More Details

- A detailed notebook with code is available is available here⁶
- This includes full explanation on how to prepare and split the dataset, evaluation of model's results and hypothesis testing

⁶<https://github.com/heyad/Teaching/tree/master/Python-Intro>

Linear Regression

- $\hat{Y} = \theta_0 + \theta_1 X_1 + \theta_2 X_2 + \dots + \theta_p X_k + \epsilon$, or

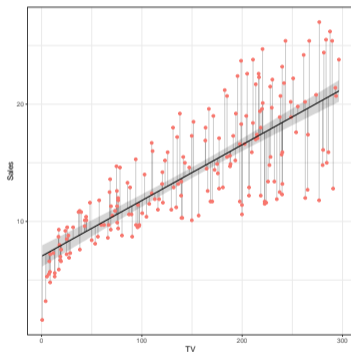


- $h_{\theta}(x) = (\theta^T x)$

Classification

Linear Regression

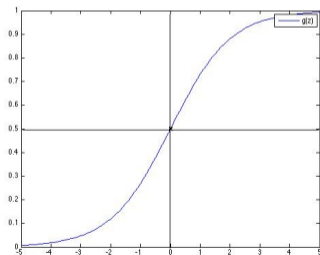
- $\hat{Y} = \theta_0 + \theta_1 X_1 + \theta_2 X_2 + \dots + \theta_p X_k + \epsilon$, or



- $h_{\theta}(x) = (\theta^T x)$

Logistic Regression

- $h_{\theta}(x) = \mathbf{g}(\theta^T x)$, where
- $g(z) = \frac{1}{1+e^{-z}}$ - sigmoid function
- $h_{\theta}(x) = \frac{1}{1+e^{-\theta^T x}}$



Classification - Structured Data

- Predict heart failures in people with cardiovascular disease given a set of input features (e.g. age, sex, type of chest pain, etc...)

	age	sex	chestPain	bloodpressureRest	serumchole	fastingSugar	restingECC	heartRateMax	exercise	oldPeak	slope	Maxwals	stat	label
1	67	0	3	115	564	0	2	160	0	1.6	2	0	7	1
2	57	1	2	124	261	0	0	141	0	0.3	1	0	7	2
3	64	1	4	128	263	0	0	195	1	0.2	2	1	7	1
4	74	0	2	120	269	0	2	121	1	0.2	1	1	3	1
5	65	1	4	120	177	0	0	140	0	0.4	1	0	7	1
6	56	1	3	130	256	1	2	142	1	0.6	2	1	6	2
7	59	1	4	110	239	0	2	142	1	1.2	2	1	7	2
8	60	1	4	140	293	0	2	170	0	1.2	2	2	7	2
9	63	0	4	150	407	0	2	154	0	4.0	2	3	7	2
10	59	1	4	135	234	0	0	161	0	0.5	2	0	7	1
11	53	1	4	142	226	0	2	111	1	0.9	1	0	7	1
12	44	1	3	140	335	0	2	180	0	0.9	1	0	3	1
13	61	1	1	134	234	0	0	145	0	2.6	2	2	3	2
14	57	0	4	128	303	0	2	159	0	0.0	1	1	3	1
15	71	0	4	112	149	0	0	125	0	1.6	2	0	3	1
16	46	1	4	140	311	0	0	120	1	1.8	2	2	7	2
17	53	1	4	140	203	1	2	155	1	3.1	3	0	7	2
18	64	1	1	110	211	0	2	144	1	1.8	2	0	3	1
19	49	1	1	140	189	0	0	178	1	1.4	1	0	7	1
20	67	1	4	120	229	0	2	129	1	2.6	2	2	7	2
21	48	1	2	130	245	0	2	180	0	0.2	2	0	3	1
22	43	1	4	115	303	0	0	181	0	1.2	2	0	3	1
23	47	1	4	112	204	0	0	143	0	0.1	1	0	3	1
24	54	0	2	132	286	1	2	159	1	0.9	1	1	3	1
25	48	0	3	130	275	0	0	139	0	0.2	1	0	3	1
26	47	0	4	136	343	0	0	133	0	0.2	1	0	3	1

Classification - Structured Data

- Predict heart failures in people with cardiovascular disease given a set of input features (e.g. age, sex, type of chest pain, etc...)

	age	sex	chestPain	bloodpressureRest	serumchole	fastingSugar	restingECC	heartRateMax	exercise	oldPeak	slope	Maxwals	stat	label
1	67	0	3	115	564	0	2	160	0	1.6	2	0	7	1
2	57	1	2	124	261	0	0	141	0	0.3	1	0	7	2
3	64	1	4	128	263	0	0	195	1	0.2	2	1	7	1
4	74	0	2	120	269	0	2	121	1	0.2	1	1	3	1
5	65	1	4	120	177	0	0	140	0	0.4	1	0	7	1
6	56	1	3	130	256	1	2	142	1	0.6	2	1	6	2
7	59	1	4	110	239	0	2	142	1	1.2	2	1	7	2
8	60	1	4	140	293	0	2	170	0	1.2	2	2	7	2
9	63	0	4	150	407	0	2	154	0	4.0	2	3	7	2
10	59	1	4	135	234	0	0	161	0	0.5	2	0	7	1
11	53	1	4	142	226	0	2	111	1	0.9	1	0	7	1
12	44	1	3	140	335	0	2	180	0	0.9	1	0	3	1
13	61	1	1	134	234	0	0	145	0	2.6	2	2	3	2
14	57	0	4	128	303	0	2	159	0	0.0	1	1	3	1
15	71	0	4	112	149	0	0	125	0	1.6	2	0	3	1
16	46	1	4	140	311	0	0	120	1	1.8	2	2	7	2
17	53	1	4	140	203	1	2	155	1	3.1	3	0	7	2
18	64	1	1	110	211	0	2	144	1	1.8	2	0	3	1
19	49	1	1	140	199	0	0	178	1	1.4	1	0	7	1
20	67	1	4	120	229	0	2	129	1	2.6	2	2	7	2
21	48	1	2	130	245	0	2	180	0	0.2	2	0	3	1
22	43	1	4	115	303	0	0	181	0	1.2	2	0	3	1
23	47	1	4	112	204	0	0	143	0	0.1	1	0	3	1
24	54	0	2	132	286	1	2	159	1	0.9	1	1	3	1
25	48	0	3	130	275	0	0	139	0	0.2	1	0	3	1
26	47	0	4	136	343	0	0	133	0	0.2	1	0	3	1

Classification - Structured Data

- Predict heart failures in people with cardiovascular disease given a set of input features (e.g. age, sex, type of chest pain, etc...)

age	sex	chestPain	bloodpressureMax	serumchole	fastingSugar	restingECC	heartRateMax	exercise	oldPeak	slope	Misseds	stat	label	
1	67	0	3	115	564	0	2	160	0	1.6	2	0	2	
2	57	1	2	124	261	0	0	141	0	0.3	1	0	7	2
3	64	1	4	128	263	0	0	195	1	0.2	2	1	7	1
4	74	0	2	120	269	0	2	121	1	0.2	1	1	3	1
5	65	1	4	120	177	0	0	140	0	0.4	1	0	7	1
6	56	1	3	130	256	1	2	142	1	0.6	2	1	6	2
7	59	1	4	110	239	0	2	142	1	1.2	2	1	7	2
8	60	1	4	140	293	0	2	170	0	1.2	2	2	7	2
9	63	0	4	150	407	0	2	154	0	4.0	2	3	7	2
10	59	1	4	135	234	0	0	161	0	0.5	2	0	7	1
11	53	1	4	142	226	0	2	111	1	0.9	1	0	7	1
12	44	1	3	140	335	0	2	180	0	0.9	1	0	3	1
13	61	1	1	134	234	0	0	145	0	2.6	2	2	3	2
14	57	0	4	128	303	0	2	159	0	0.0	1	1	3	1
15	71	0	4	112	149	0	0	125	0	1.6	2	0	3	1
16	46	1	4	140	311	0	0	120	1	1.8	2	2	7	2
17	53	1	4	140	203	1	2	155	1	3.1	3	0	7	2
18	64	1	1	110	211	0	2	144	1	1.8	2	0	3	1
19	49	1	1	140	199	0	0	178	1	1.4	1	0	7	1
20	67	1	4	120	229	0	2	129	1	2.6	2	2	7	2
21	48	1	2	130	245	0	2	180	0	0.2	2	0	3	1
22	43	1	4	115	303	0	0	181	0	1.2	2	0	3	1
23	47	1	4	112	204	0	0	143	0	0.1	1	0	3	1
24	54	0	2	132	286	1	2	159	1	0.9	1	1	3	1
25	48	0	3	130	275	0	0	139	0	0.2	1	0	3	1
26	47	0	4	136	311	0	0	131	1	0.6	1	0	3	1

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & \vdots & \dots & x_{mp} \end{bmatrix}, Y = \begin{bmatrix} y_1 \\ \dots \\ y_m \end{bmatrix}$$

- Classification:** If Y is a set of discrete values

Logistic Regression

- Logistic Function or simply sigmoid is often expressed as

$$\phi(z) = \frac{1}{1+e^{-z}},$$

- Here, \mathbf{z} is the input, which is the combination of **weights** (coefficients) and the input features and is expressed as

$$z = \mathbf{w}^T \mathbf{x} = \mathbf{w}_0 + \mathbf{w}_1 x_1 + \mathbf{w}_2 x_2 + \dots + \mathbf{w}_m x_m,$$

- $\mathbf{e} \approx 2.71828$ is a mathematical constant (Euler's number)

Logistic Regression

- The output of the logistic function is a value between **0 and 1** representing the **probability** of a sample belonging to particular **class**
- The probability is then converted easily into a particular class

$$\hat{y} = \begin{cases} 1 & \phi(z) \geq 0.5 \\ 0 & \text{Otherwise} \end{cases}$$

- For the Heart patients data, logistic regression models the probability of a patient having heart condition given his/her data as

$$P_r(\text{Heart}_{cond} = \text{Yes} | \text{age}, \text{sex}, \text{ECG}, \dots),$$

Logistic Regression - Cost Function

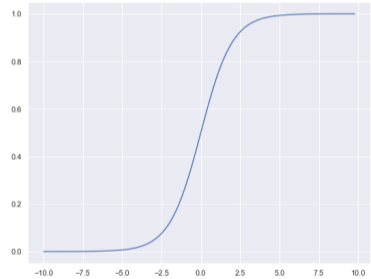
- Instead of using Mean Squared Error, as we did last week with linear regression, for logistic regression the cost function is called Cross-Entropy, and commonly known as the Log Loss

$$J(w) = \sum_{i=1}^n [-y^{(i)} \log(\phi(z^{(i)})) - (1 - y^{(i)}) \log(1 - \phi(z^{(i)}))]$$

- One thing to note here, training the Logistic Regression Model is to learn **weights** for the input features (more details in the lab)

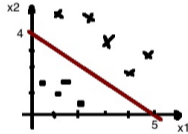
Decision boundary

- $P_r(\text{Heart}_{cond} = \text{Yes} | \text{age}, \text{sex}, \text{ECG}, \dots)$
- The decision could be that if the probability of being 1 is greater than 0.5 then we can predict 1, otherwise 0
- Looking at the graphic of the sigmoid
 $g(z) > 0.5$ when $z > 0$ or $\theta^T x > 0$ then the hypothesis predicts $y = 1$



Linear Decision boundary

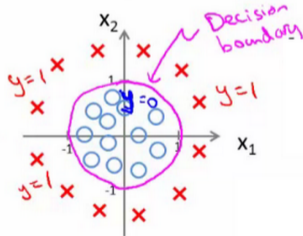
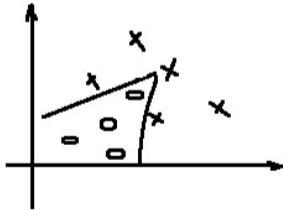
- $h_{\theta}(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$
- $\theta_0 = -4, \theta_1 = 4, \theta_2 = 5, -4 + 4x_1 + 5x_2 \geq 0$ prediction is $y = 1$
- $4x_1 + 5x_2 = 4$ is the decision boundary



- The decision boundary is a property of the hypothesis. We can create the boundary with the hypothesis and parameters without any data. We can use the data to determine the parameter values.

Non-linear Decision boundary

- We may have more complex non linear data set
- $h_{\theta}(x) = g(\theta_0 + \theta_1x_1 + \theta_2x_1^2 + \theta_3x_2^2)$
- Using higher order polynomial terms will enable more complex decision boundaries



Evaluation

- Train Logistic Regression to learn the **label** given customers data (age, sex, ..)

age	sex	chestPain	bloodpressureRest	serumcholst	...	label
0.79	0	0.67	0.20	1.00	...	1
0.58	1	0.33	0.28	0.31	...	2
0.73	1	1.00	0.32	0.31	...	1
0.94	0	0.33	0.25	0.33	...	1
0.75	1	1.00	0.25	0.12	...	1
0.56	1	0.67	0.34	0.30	...	2
0.62	1	1.00	0.15	0.26	...	2
0.65	1	1.00	0.43	0.38	...	2
0.71	0	1.00	0.53	0.64	...	2
0.62	1	1.00	0.39	0.25	...	1

- **Actual** class values (e.g label)
- Predicted class values (resulting from your model)
- Estimated probability

Evaluation Metrics- Confusion Matrix

		<u>Actual Values</u>	
		Positive (1)	Negative(0)
<u>Predicted Values</u>	Positive (1)	TP	FP
	Negative (0)	FN	TN

- True Positive (TP), True Negative(TN), False Positive(FP), False Negative(FN)
- Correct predictions fall on the diagonal of the matrix
- Off the diagonal are instances that has been misclassified
- Performance is based on the counts of the predictions on and off the diagonal of the confusion matrix

Evaluation Metrics

- Accuracy = $\frac{TP+TN}{TP+TN+FP+FN}$
- Error rate = $\frac{FP+FN}{TP+TN+FP+FN} = 1 - \text{Accuracy}$
- Sensitivity: the proportion of positive examples that were correctly classified (true positive rate) = $\frac{TP}{TP+FN}$
- Specificity: measures the proportion of negative examples that were correctly classified (true negative rate) = $\frac{TN}{TN+FP}$
- Precision = $\frac{TP}{TP+FP}$
- Recall = $\frac{TP}{TP+FN}$
- F-measure = $\frac{2 \times \text{precision} \times \text{recall}}{\text{recall} + \text{precision}}$

Bias-Variance Trade-off

- A machine learning model's performance is often defined by the **bias** and the **variance** of the model. Therefore before we release our model/ app. We need to understand its performance, and the model's error which can be defined as:

$$Model_{err} = Variance(Model) + Bias(Model) + Irreducible(Err)$$

- But, what is Bias, and Variance?

Bias-Variance Trade-off

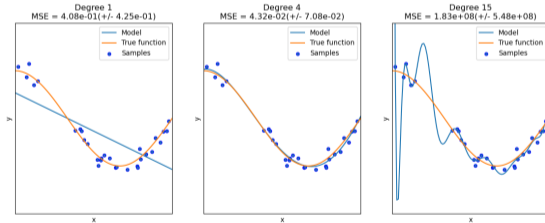
- **Train-Test:** Recall, that when building a Machine Learning Model, we first train it on a training data (i.e. subset of the original dataset), then test it using a testing dataset
- A model with **High Variance**, is simply the one that performs very well on the training data, but poorly on the testing data (overfitting)
- A model with **High Bias** is *underfitting* the data. It doesn't perform well enough on the training data (not complex enough), and won't generalise to unseen examples (testing data)
- So, what is *Overfitting*, and *Underfitting*

Underfitting vs. Overfitting

Degree 1: $\hat{Y} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_k + \epsilon$ (*Underfitting*)

Degree 4: $\hat{Y} = \beta_0 + \beta_1 X_1^4 + \beta_2 X_2^4 + \dots + \beta_p X_k^4 + \epsilon$

Degree 15: (*Overfitting*)



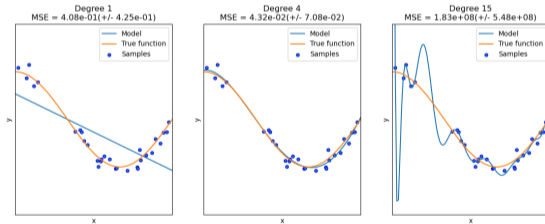
(Image source) : <https://scikit-learn.org>

Underfitting vs. Overfitting

Degree 1: $\hat{Y} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_k + \epsilon$ (*Underfitting*)

Degree 4: $\hat{Y} = \beta_0 + \beta_1 X_1^4 + \beta_2 X_2^4 + \dots + \beta_p X_k^4 + \epsilon$

Degree 15: (*Overfitting*)



(Image source) : <https://scikit-learn.org>

```
intercept = 2.938889369459412
tv_coeff = 0.04576465
radio_coeff = 0.18853082
news_coeff = 0.00183749 # negative
# make prediction
new_sales = intercept + (tv_coeff*tv) + (radio_coeff*radio)
            - (news_coeff*newspaper)
```

myApp.py(Week 2)

Regularisation

- One way to find a good bias-variance *tradeoff* is to tune the complexity of your machine learning model with regularisation
- **Regularisation** is good way to handle collinearity, noise, and prevent overfitting. The method includes adding an additional term to penalise extreme parameter weights, such as **L2 regularisation** expressed as $\frac{\lambda}{2} \sum_{j=1}^m w_j^2$, so the cost function becomes

$$J(w) = \sum_{i=1}^n [-y^{(i)} \log(\phi(z^{(i)})) - (1 - y^{(i)}) \log(1 - \phi(z^{(i)}))] + \frac{\lambda}{2} \sum_{j=1}^m w_j^2$$

- Implementation using [`sklearn.linear_model.LogisticRegression`](#)

Generalization, Over-fitting Under-fitting

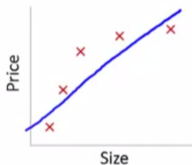
- **Generalization** - The ability to correctly classify new examples different from those used for the training.
- **Overfitting** - The learned classifier is 100% correct on the train data and only 50% correct on the test data.
- **Underfitting** - The learned classifier is too simplistic and does not capture the structure of the data.

Bias, Variance

The main objective is to find the best possible function $h(x)$ that maps X (a set of features) to a class label y (classification problem). The prediction error for any machine learning algorithm can be defined as:

- Bias error (*under-fitting*)
- Variance error (*over-fitting*)
- Irreducible error

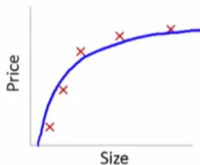
Bias, Variance



$$\theta_0 + \theta_1 x$$

High bias
(underfit)

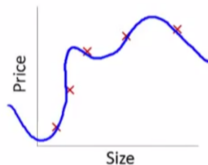
$$d=1$$



$$\theta_0 + \theta_1 x + \theta_2 x^2$$

“Just right”

$$d=2$$



$$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$$

High variance
(overfit)

$$d=4$$

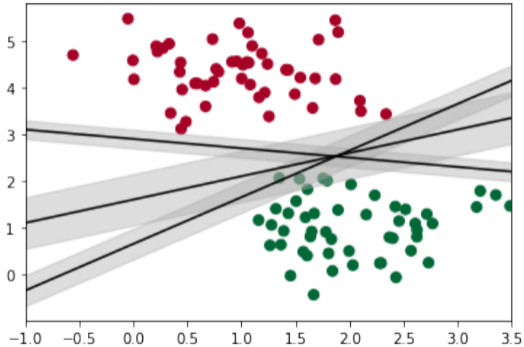
Andrew Ng⁷

⁷[http:](http://)

Other Algorithms

Support Vector Machine

- Which line (model/ classifier) best describes the data, and we should use



- In SVM, the optimisation objective is to maximise the margin (i.e. distance separating line to the support vectors)

Decision Tree

- Iris dataset: 150 rows (**instances**), each representing a flower, 4 columns (**features**), and three class **labels**. In this example, we will only use two features (for visualisation purposes)

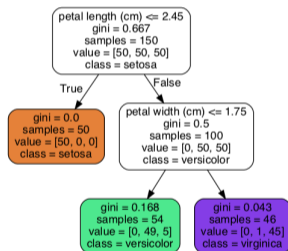
Sepal Length	Sepal Width	Petal Length	Petal Width	Species
5.10	3.50	1.40	0.20	setosa
4.90	3.00	1.40	0.20	setosa
4.70	3.20	1.30	0.20	setosa
4.60	3.10	1.50	0.20	setosa
5.00	3.60	1.40	0.20	setosa
5.40	3.90	1.70	0.40	setosa
4.60	3.40	1.40	0.30	setosa
5.00	3.40	1.50	0.20	setosa
4.40	2.90	1.40	0.20	setosa
4.90	3.10	1.50	0.10	setosa
...	setosa

- Task:** build a classifier that classifies a flower based on the four features

Decision Tree

- Iris dataset: 150 rows (**instances**), each representing a flower, 4 columns (**features**), and three class **labels**. In this example, we will only use two features (for visualisation purposes)

Sepal Length	Sepal Width	Petal Length	Petal Width	Species
5.10	3.50	1.40	0.20	setosa
4.90	3.00	1.40	0.20	setosa
4.70	3.20	1.30	0.20	setosa
4.60	3.10	1.50	0.20	setosa
5.00	3.60	1.40	0.20	setosa
5.40	3.90	1.70	0.40	setosa
4.60	3.40	1.40	0.30	setosa
5.00	3.40	1.50	0.20	setosa
4.40	2.90	1.40	0.20	setosa
4.90	3.10	1.50	0.10	setosa
...	setosa



- Task:** build a classifier that classifies a flower based on the four features

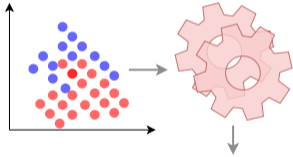
Ensemble Learning

Training Data

First, we train the model on a training set, then we test it on different data (testing set)

Model

Linear regression, logistic regression, SVM, Decision Tree



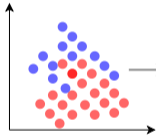
Predictions

- Predictions
- Classifications
- Forecasting
- ...

Ensemble Learning

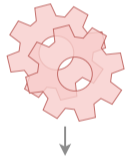
Training Data

First, we train the model on a training set, then we test it on different data (testing set)



Model

Linear regression, logistic regression, SVM, Decision Tree

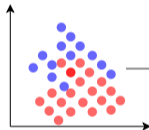


Predictions

- Predictions
- Classifications
- Forecasting
- ...

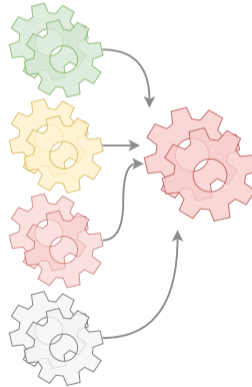
Training Data

Train more than one model on the training set



Models

SVM, Logistic Regression, Decision Trees and others, ...



Predictions

- Predictions
- Classifications
- Forecasting
- ...

Ensemble Construction

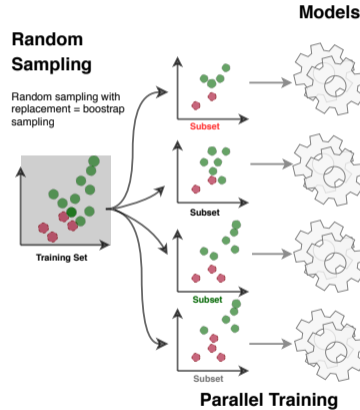
- Train more than one model (i.e SVM, KNN, Logistic Regression, etc...)
- Aggregate the predictions of each model and predict the class that gets the most votes
- A majority-vote classifier is called *hard-voting* classifier
- A good ensemble needs **models** to be **diverse** enough and **independent** from each other. So, how to ensure ensemble diversity?

Ensemble Diversity

- One way to ensure diverse ensemble is to train very *different learning algorithms* such as Support Vector Machine, Logistic Regression, K-NN, and others on the same training data
- The second approach that is also widely used is to train the same algorithms on *different subsets of the data* (training sets)

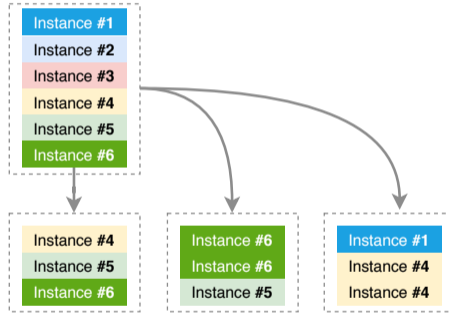
Bagging

- **Bootstrap aggregating** (Bagging) is sampling data from the training set with replacement
- With Bagging an instance can be sampled more than one time for the same model/ predictor
- Once all models are trained, the ensemble can make a prediction for a new instance by aggregating the predictions from all models



Bagging

- Some instances may be sampled several times. Others may not be sampled at all
- Each model in the ensemble will be trained on almost **63%** of the training set?
- The remaining **37%** of the data are used for **out-of-bag(oob)** evaluation of each model in the ensemble



Random Forest

- An ensemble classification and regression technique introduced by Leo Breiman
- It generates a diversified ensemble of decision trees adopting two methods:
 - A bootstrap sample is used for the construction of each tree (bagging), resulting in approximately 63.2% unique samples, and the rest are repeated
 - At each node split, only a subset of features are drawn randomly to assess the goodness of each feature/attribute (\sqrt{F} or $\log_2 F$ is used, where F is the total number of features)
- Trees are allowed to grow without pruning
- Typically 100 to 500 trees are used to form the ensemble
- It is now considered among the best performing classifiers

Random Forest

In one of the largest experiments that have been carried out in 2014, researchers used:

- 179 classifiers
- 121 datasets (the whole UCI repository at the time of the experiment)
- Random Forest was the first ranked, followed by SVM with Gaussian kernel

Recommended Reading

Fernandez-Delgado, M., Cernadas, E., Barro, S., & Amorim, D. (2014). [Do we need hundreds of classifiers to solve real world classification problems?](#). The Journal of Machine Learning Research, 15(1), 3133-3181.

Neural Networks

- Neural Networks (NN) is another algorithm that can be used to perform regression and classification tasks
- NN and CNNs to be covered in the next section

Plan

① Overview

- The Right Questions
- Why ML and Algorithms?
- Think Data
- Recommended Readings

② Supervised Learning

- Problem / Data collection
- Solution
- Model/s
- Evaluation / Testing

③ Conclusion

Conclusion

- Finding articulating a problem is the most important and crucial step for building intelligent machine vision solution
- ML and AI is very interdisciplinary field (applicable across all sectors and applications domains)

Conclusion

- Wide range of tools and learning resources are available to get into Machine Learning and AI

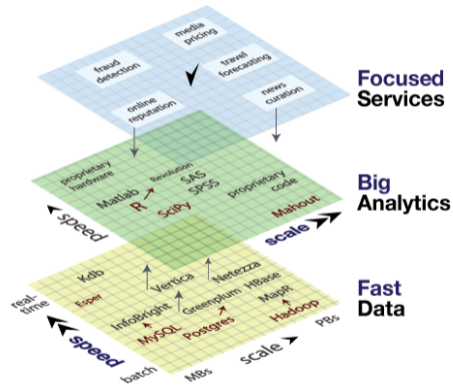
Conclusion

- Python
- R
- Tensorflow, Keras, PyTorch (Deep Learning framework)
- And others



Big Data Tools

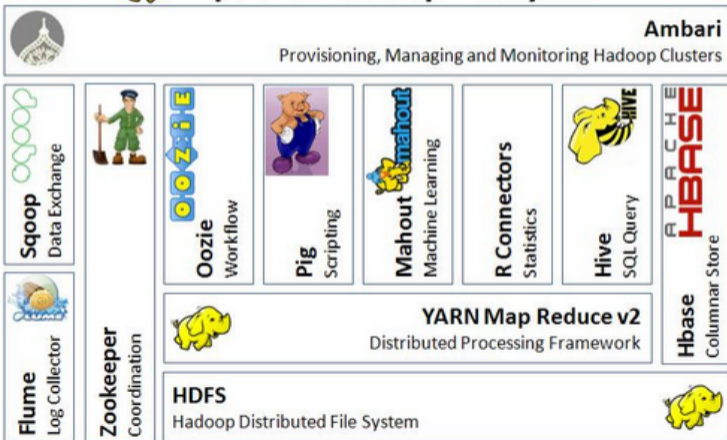
The Emerging Big Data Stack



[Source **Source:** <http://medriscoll.com/post/9062115121/the-big-data-stack-from-my-piece-building-data>]



Apache Hadoop Ecosystem



Open Source Tools / Knowledge

Learning Resources (Free)

- Books, videos, Code⁸
- Papers with Code⁹

⁸<https://github.com/academic/awesome-datascience>

⁹<https://paperswithcode.com/>

Open Source Tools / Knowledge

Learning Resources (Free)

- Books, videos, Code⁸
- Papers with Code⁹

- Read and process images and videos ([opencv](#))
- Almost all ML algorithms ([scikit-learn](#))
- [Keras](#) implements most Deep Learning methods
- ...

⁸<https://github.com/academic/awesome-datascience>

⁹<https://paperswithcode.com/>

Thank You

[@ElyanEyad](#)

<https://www3.rgu.ac.uk/dmstaff/elyan-eyad>