Python Machine Learning: The 0th Book Circle

Big Picture of Machine Learning vs Signal Processing, Mathematics, Python, and Book

Jianan Liu

Gothenburg

Oct, 3, 2016

Disclaimer

All opinions and statements in this and following presentations in the series of python machine learning book circle are mine and do not in any way represent the company. As an engineer who has statistical signal processing/optimization for telecom academic background, most of knowledge in machine learning learned are from self-studying. I do believe there are errors somewhere in the presentations of python machine learning book circle series, so please feel free to contact me regarding any comment or correction of error by following mailbox: chisyliu@hotmail.com

Acknowledgement

Acknowledgement update in Dec 2017

I would like to thank following persons who helped/inspired me during the year long machine learning journey, for the fruitful discussion, happy lunch time and wonderful afterwork/fika: Mengbai Tao, Andreas Hultberg, Hui Wen, Ziqi Peng, Caj Zell, Yi Zhang, Xingyu Zheng from Ericsson(For Hui Wen left for Amazon and Ziqi Peng left for Goldman Sachs, I wish you guys good luck in U.K.); Dapeng Liu and Tiancong Zheng from Zenuity; Lei Chen from Viktoria ICT; Kaiye Chang from Volvo Car. Thank you all for making me finishing up all presentations, by some way consumption of your time and patience, especially when I asked silly questions or kept talking something you actually didn't want to discuss.

Right before We Dive into Machine Learning

In this presentation

- We will try to go through general background of machine learning, in order to give us some basic feeling regarding machine learning
- We will try to show some key words and important knowledge(mathematics) in machine learning, in order to give us some hints on 'If I want to know more regrading machine learning, what words or names I should search on google'
- We will figure out similarity between machine learning and signal processing for wireless communication

Right before We Dive into Machine Learning

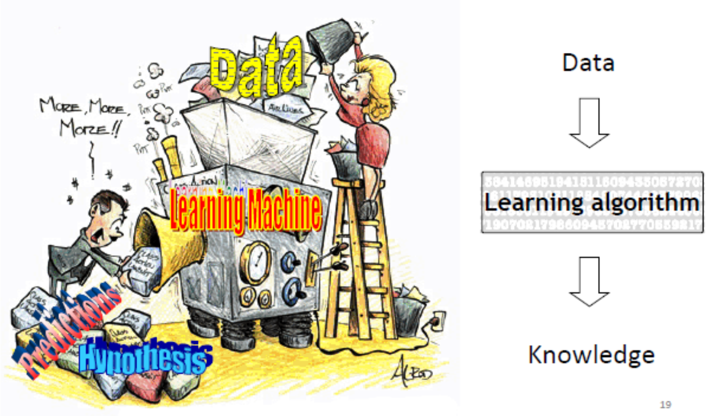
In the following series of Python Machine Learning presentations

- We will try to go through details of different algorithms(mainly **classical data mining/machine learning algorithms**), methodologies, and application listed in python machine learning book
- We will also go through lots of **statistical/adaptive signal processing algorithms**, **classical and deep neural network techniques**, **Bayesian filters(Kalman filter, etc.) algorithms** which are NOT covered by python machine learning book but widely used in different digital signal systems, e.g. wireless communication, object recognition, autonomous tracking system, since they are highly relative to machine learning algorithm
- We will try to show all the mathematics behind the algorithms in "engineering sense", which means usually we use mathematics conclusion to explain why and how for algorithms but without showing the exact derivation

Overview

Outline for Section 1

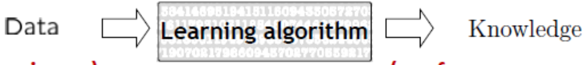
What is Machine Learning?



Remark

Design the **learning algorithms** that improve the **performance** of task based on **experience**.

What is Machine Learning?



(experience) (performance on task)

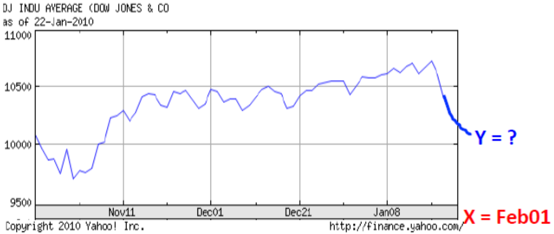
Remark

The "learning algorithms" includes

- **Model**, which is the mathematical approximation of features of a physical world task(problem)
- **Algorithm**, which is method to "solve" the model(select the exact model with concrete parameters among all the hypothesis)

Purpose/Application of Using Machine Learning?

- Stock Market Prediction

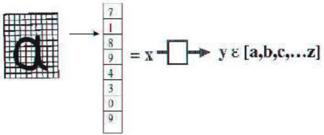


Question

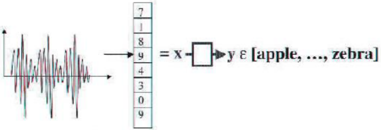
Is this a machine learning task? Is it possible to predict?

Purpose/Application of Using Machine Learning?

- **Email filtering** $x \in [\text{a-z}]^{+} \rightarrow \boxed{} \rightarrow y \in [\textbf{important, spam}]$
- **Character recognition**



- **Speech recognition**

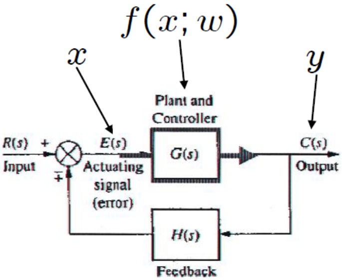


Question

Are they machine learning task?

Purpose/Application of Using Machine Learning?

- Automatic control of a vehicle



Question

Design of algorithm for **equalization**, **MIMO(spatial multiplexing)** precoding in wireless communication system, or echo cancellation, adaptive enhancement for audio system. **Are they machine learning problem?**

Purpose/Application of Using Machine Learning?

- **Already everywhere**
  - Speech recognition (e.g. Siri)
  - Computer vision (e.g. face detection)
  - Hand-written character recognition (e.g. letter delivery)
  - Information retrieval (e.g. image & video indexing)
  - Operation systems (e.g. caching)
  - Fraud detection (e.g. credit cards)
  - Text filtering (e.g. email spam filters)
  - Game playing (e.g. strategy prediction)
  - Robotics (everywhere)

Remark

(Upon **known or unknown**)Data + (select)Model(**among hypothesis to present features**) + (by **designing**)Algorithm = (for **maximizing**)Performance

Outline for Section 2

1. What is Machine Learning? Purpose of Using Machine Learning?
2. Category of Machine Learning
   - Supervised Learning vs Unsupervised Learning
3. Common Used Machine Learning Algorithms and the Mathematic Idea Behind
   - The Main Generic Mathematics Ideas Behind Machine Learning
   - Some Regression Algorithm in Supervised Learning
   - Some Classification Algorithm in Supervised Learning
   - Some Clustering Algorithm in Unsupervised Learning
   - Some Algorithm to boost Machine Learning Performance
4. What Do We Do to Organize Our Machine Learning Study
   - How Should We Study as Software Guy?
   - Which Book?
   - Which Packages to Be Installed?
   - When and How Should We Meet?
   - After This Book

Category of Machine Learning

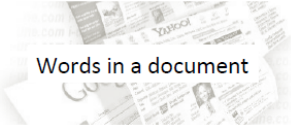
Generally speaking, machine learning could be divided into several categories depends on from which angle you look at this "categorization problem"

- Depends on if we already have some known data could be used for training the model, we have supervised learning(regression problem or classification), unsupervised learning(density estimating or clustering problem), and reinforcement learning.
- Depends on how the mathematics model looks like, we have either generativemaximize $p(x|y)p(y)$) model or discriminative(maximize $p(y|x)$) model.
- And many... As the book follows the first way of category(as most of machine learning books), we will focus on the first method of category

Supervised Learning vs Unsupervised Learning

Supervised Learning

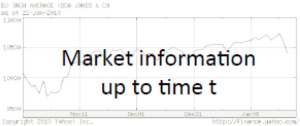
Feature Space $\mathcal{X}$



Label Space $\mathcal{Y}$

"Sports"
"News"
"Science"
…

Discrete Labels
Classification



Share Price
"$ 24.50"

Continuous Labels
Regression

Task: Given $X \in \mathcal{X}$, predict $Y \in \mathcal{Y}$.

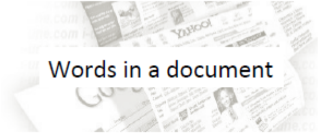
36

Supervised Learning vs Unsupervised Learning

Unsupervised Learning

Aka "learning without a teacher"

Feature Space $\mathcal{X}$



Word distribution
(Probability of a word)

Task: Given $X \in \mathcal{X}$, learn $f(X)$.

37

Supervised Learning vs Unsupervised Learning

Remark

For unsupervised learning, we **estimate the probability density function for continuous target** data without training data. We also try to **cluster the discrete target data** without training data.

Supervised Learning vs Unsupervised Learning

Remark

We also have other types of machine learning, e,g, semi-supervised learning, reinforcement learning, etc. But we will focus on the **regression** and **classification** problems in supervised learning, and **clustering** problem in unsupervised learning in this book. We will also see lots of classical models and algorithms to solve these tasks, together with a little bit modern way(deep neural network)

Outline for Section 3

The Main Generic Mathematics Ideas Behind Machine Learning

- If we go back to 40 years ago, the machine learning was called as statistical machine learning. Cause most of the methods and models used are relative to **statistics**, especially **bayes rule**.
- From new century the algorithms and models based on beautiful (convex or stochastic) **optimization** problems started to be the dominate technologies and hot topics in machine learning.
- As the big data(massive amount of data) could be captured from the connected world nowadays, the old nonlinear model **neural network** has been extended a lot to have the ability to process the massive data with much better performance.
- Besides, **matrix theory** is also important cause most of the mathematical model and operation are represented in matrix and vector.

Some Common Used Machine Learning Algorithms

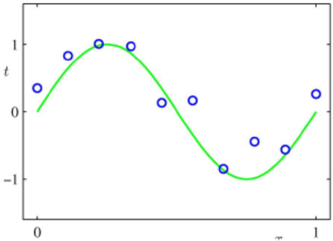
- Now we start to go through some typical machine learning algorithms
- For each of four categories, regression, classification, clustering, boost performance of machine learning, we talk about one machine learning algorithm in bit detail and mentioned the names of others
- We will see what is the relationship between mathematics used in those algorithms and mathematics for signal processing

Regression Algorithm in Supervised Learning

Least Square Linear Regression

For regression problem, or so called 'curve fitting'. There are L known data pairs y_l, $\mathbf{x}_l$ where $\mathbf{x}_l$ is N by 1 vector, by using these limited number of data we have Least Square Linear Regression to predict the unknow y for new data $\mathbf{x}$

Consider the polynomial curve fitting. The model is linear in the parameters $\mathbf{w}$ and non-linear in inputs $\mathbf{x}$



$$y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M = \sum^{M} w_j x^j$$

Regression Algorithm in Supervised Learning

Generally

$$y(\mathbf{x}, \mathbf{w}) = \sum_{j=0}^{M-1} w_j \phi_j(\mathbf{x}) = \mathbf{w}^{\mathrm{T}} \boldsymbol{\phi}(\mathbf{x})$$

where $\phi_j(x)$are known as *basis functions*.

Polynomial basis functions:

$$\phi_j(x) = x^j.$$

Gaussian basis functions:

$$\phi_j(x) = \exp\left\{-\frac{(x-\mu_j)^2}{2s^2}\right\}$$

Note

x is NOT the variable(unknown parameter we want to solve), but **w** is.
We call it linear regression is due that the model is linear model to **w**

Regression Algorithm in Supervised Learning

Assume observation from this model with added gaussian noise

$$t = y(\mathbf{x}, \mathbf{w}) + \epsilon \qquad \text{where} \qquad p(\epsilon|\beta) = \mathcal{N}(\epsilon|0, \beta^{-1})$$

What we want to do is trying to find the exact values of **w** by minimizing the noise to let the model fit the known(training) data pairs y_l, $\mathbf{x}_l$ for all l in L.

Question

How to do it? We try to minimize the sum of squared errors by finding the suitable **w**

The *least squares* objective tells us to pick the w that minimizes the sum of squared errors

$$w_{\mathrm{LS}} = \arg\min_{w} \sum_{i=1}^{n} (y_i - f(x_i; w))^2 \equiv \arg\min_{w} \mathcal{L}.$$

Regression Algorithm in Supervised Learning

- Due to the **convexity** of objective function, it has a global minimum value of $\mathbf{w}$.
- When the number of training data is larger than the dimension of $\mathbf{x}$, we can solve this optimization(minimizing the objective function) problem **analytically** by setting the gradient to zero.
- When assume the noise is gaussion, we can model this problem from **statistics** point of view to **maximize the likelihood(ML)**. The result will be exact same as LS from optimization point of view.

Regression Algorithm in Supervised Learning

- The analytical solution is

Solving for **w**, we get

$$\mathbf{w}_{\mathrm{ML}} = \underbrace{\left(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^{\mathrm{T}}}\mathbf{t}$$

The Moore-Penrose pseudo-inverse, $\boldsymbol{\Phi}^{\dagger}$.

- In case the pseudo inverse can't be found(e.g. the number of training data is far smaller than the dimension of **x**), we could use adaptive algorithm, e.g. **batch/stochastic gradient descent**(BGD/SGD) to solve it. Or we can change the objective function by adding penalty factor, e.g. L2 norm, to make the autocorrelation matrix invertible
- In case we want to avoid computing of correlation matrix in expectation, simpler adaptive/recursive algorithm could be applied, e.g. **least mean squared**(LMS) or **recursive least square**(RLS)

Regression Algorithm in Supervised Learning

Let's review the conclusions from last slide, we could find out similarity between wireless baseband equalizer designing and least square linear regression.

- The linear model we use is same as the model of wireless baseband transmission.
- The LS linear regression method is actual the **Zero Forcing(ZF) equalizer** we use in receiver of wireless system, e.g. 4G. which is based on LS method.
- In ZF equalizer we also could use same recursive algorithm, e.g. recursive least square(RLS) to compute the $\mathbf{w}_l$.
- Both LS linear regression and ZF equalizer use NO statistical information(e.g. means of ensemble data) but only information regarding the L known data pairs.

Regression Algorithm in Supervised Learning

We have lots of algorithms for regression task

Bayes Linear Regression

e.g. Ride regression. It turns the ML to **maximize a posterior**(MAP) by using bayes rule with a prior information. It uses additional l2 norm penalty factor to lower the variance(lower the overfit)(although the expectation is not unbias anymore)

- From optimization view, MAP bayes linear regression is identical to **minimize mean square error(MMSE) equalizer** in wireless baseband if the same linear model is used
- For MMSE, recursive algorithm/adaptive, e.g. **batch/stochastic gradient descent**(BGD/SGD) can also be used to make complexity lower

Regression Algorithm in Supervised Learning

We have lots of algorithms for regression task

Lasso Linear Regression

It uses l1 norm penalty factor to solve linear regression task when the number of training data is far smaller than the dimension of $\mathbf{x}$.

Logistic Linear Regression

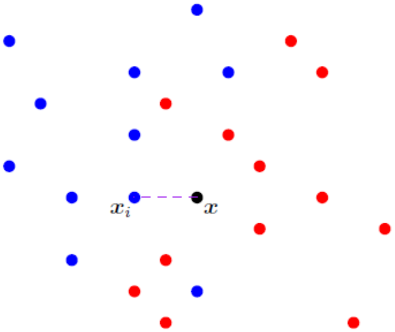
It uses logistic function to map the output from linear regression model to a binary(discrete value), it is actually for classification task.

And many more...

Classification Algorithm in Supervised Learning

classification task

Given data (x_1, y_1), ... , (x_n,y_n), $y_n \in \{red, blue\}$. Each pair belongs to either red or blue. we try to model a classifier f(x) = y for new data x(black) to judge which color it should belong to



k Nearest Neighbor

- For new data x(black), we find k points which are with closest distance to x, indexed as x_i^1, x_i^2,...,x_i^k
- We check what color most of the k points belong to, return the major color as the value y

Classification Algorithm in Supervised Learning

- Really naive algorithm
- It could be designed for regression. return y value which is the weighted average of y value all other chosen k points

Question

How to define 'closest distance'

- Euclidean distance.

$$\sum_{n=1}^{N} \sqrt{x^2 - x_i^2}$$

, n stands for number of features

- Absolute distance.

$$\sum_{n=1}^{N} |x - x_i|$$

, n stands for number of features

and many more

Classification Algorithm in Supervised Learning

We have lots of algorithms for classification task

Logistic Linear Regression

It uses logistic function $\frac{1}{1+e^{t}}$ to map the output from linear regression model to a binary(discrete value)

Naive Bayes Classification

Similar to Bayes Linear Regression, MAP to replace the ML in logistic linear regression

Supportive Vector Machine

It use technique in constraint optimization problem and convexity to make the nonlinear model linearizable in hyperplane. it could be extended into other algorithms for both regression and classification. The best one in performance before deep neural network show up.

And many more...

Expectation Maximization

Let's show the EM on board

Kalman Filter could be extended to EM algorithm, we see what is the relationship between them on board

Other Common Used Clustering Algorithm

- k Means Alg
- Sequential Leader
- Affinity Propagation
- etc.

Principle Component Analysis

- **principal component analysis**(PCA) is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components.
- The number of principal components is less than or equal to the smaller of the number of original variables or the number of observations.
- The number of 'real' features(amount of data need to be processed) could be reduced a lot but without losing the ability to represent the task. Lead to low computational complexity.

Algorithm to boost Machine Learning Performance

Question

How to get the principal components?

- It is based on a technique in matrix theory, Singular value decomposition(SVD) of data matrix $\mathbf{X}$(it could alsobe seen as Eigen decomposition of a $\mathbf{X}^\mathbf{T}\mathbf{X}$)
- Any m by n real or complex matrix $\mathbf{X}$ could be decomposed as $\mathbf{X} = \mathbf{UDV}^*$ where $\mathbf{U}$ is an m by m real or complex unitary matrix, $\mathbf{D}$ is a m by n rectangular diagonal matrix with non-negative real numbers on the diagonal, and $\mathbf{V}$ is other n by n real or complex unitary matrix. $\mathbf{V}^*$ transposed conjugate of $\mathbf{V}$

Algorithm to boost Machine Learning Performance

- Non-negative real numbers on this diagonal of **D** stands for the principal components in last slide.
- Note number of non-negative real numbers on this diagonal of **D** could be much smaller than the min(m, n). That is how we reduce the computational complexity.
- Note number of non-negative real numbers on this diagonal of **D** is actually the rank of **X** .

Algorithm to boost Machine Learning Performance

Let's recall the SVD(PCA), are we familiar with this technique?

- Yes, the MIMO(spatial multiplexing) for 4G and 5G are built upon SVD(PCA)
- If we use channel matrix $\mathbf{H}$ to replace $\mathbf{X}$, we could decompose channel matrix as $\mathbf{H} = \mathbf{UDV}^*$ if we know channel matrix(or when we have really good channel estimation from feedback information)
- The purpose of MIMO is trying to find more than one parallel transmission data streams which could transmit data in same frequency and same time slot

- As we figure out from last slide, if we apply U^* at receiver side and V at transmitter side, the diagonal matrix $\mathbf{D}$ has been left for parallel streaming. The rank of $\mathbf{H}$ is equal to number of non-negative diagonal value in $\mathbf{D}$, which stands for how many parallel streaming could be generated.
- The pre-coding matrix (we already used in 4/5G) is predefined, try to match(approximate) the $\mathbf{V}$ for different cases as accurate as possible.
- That is reason why the feedback of channel estimation in LTE system needs to have PMI(pre-coding matrix indicate, stands for which precoding matrix to be used), RI(rank information, stands for how many layers can be gotten in current propagation channel).

Algorithm to boost Machine Learning Performance

- Ensemble Learning
- Boosting
- Ada Boost
- etc.

There are even more algorithm

- Recommendation System, PageRank
- Reinforcement Learning, e.g. Q-learning, MCMC(Markov Chain Monte Carol), etc.
- Deep Neural Network, e.g. CNN(Convolutional Neural Network), RNN(Recurrent Neural Networks), etc.
- Bio-inspired Algorithm, Ant Colony, Swarm, etc.
- etc.

Communication Theory(Signal Processing) vs Machine Learning

Question

Go back to the question: Design of algorithm for **equalization**, **MIMO(spatial multiplexing)** precoding in wireless communication system, or echo cancellation, adaptive enhancement for audio system. **Are they machine learning problem?**

We already saw these problems are same as machine learning tasks

- Communication Theory is where Shanon(Information Theory) meets Weiner(Estimation Theory, Statistic Signal Processing)
- The mathematics behind estimation theory are mainly bayes statistics and optimization theory
- Machine Learning relies on bayes statistics and optimization theory heavily, but nowadays (deep) neural network raises as well
- Machine Learning is where Weiner meets someone stands for neural network, correct?

Outline for Section 4

How Should We Study as Software Guy?

We already noticed machine learning technologies may NOT be easy.

Problem

Why?

We just listed small amount of machine learning technologies, but we already went through lots of math

- Statistics Signal Processing
- Convex(Stochastic) Optimization
- Matrix Theory

Actually there are more math

- Information Theory
- Mixed Integer Programming
- Nonlinear System Theory
- Probabilistic Graphical Modeling

Everyday people find out new mathematics for ML

How Should We Study as Software Guy?

Problem

What could we do?

- Go back to school and get a PhD in machine learning
- Learn by doing! Use existing(very well developed)machine learning package to solve problem

We go for later way. Focus on understanding how to use the machine learning library functions with python, without diving into math.(But I welcome discussion on mathematic)

- Some books are written for this purpose

How Should We Study as Software Guy?

For some(or all?) of us who also want to get understand of mathematics behind machine learning, use the algorithms in the new field. It is still NOT hard thing!

Problem

Why?

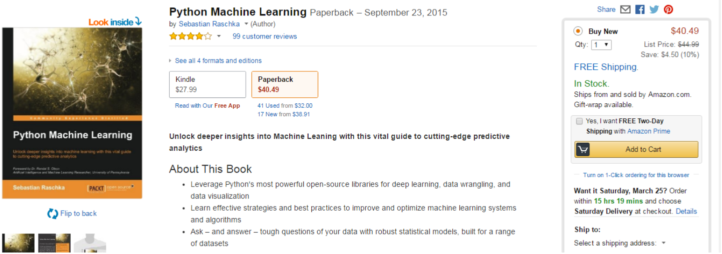
- We noticed almost all the mathematics behind machine learning are familiar
- Almost everyone who has CS/EE degree knows these mathematics, or at least has ability to study them
- We also noticed the **math behind signal processing for wireless system are almost same as ones for machine learning**

Which Book?

- We have two books to be selected, most of us voted to python machine learning

Which Book?

Python Machine Learning



- 4.2 Points
- Not well organized(from the view of systematize knowledge), jump between different topics to give more application of machine learning
- But most of reviewers point out it is the best book for now on topic 'how to write python codes to design your machine learning algorithms'. Not only use scikitlearning lib but also show the mathematics and principle.

Which Book?

Python Machine Learning

- Python Machine Learning book mainly divided contents into 3 part: **algorithm, practical methodology, application of machine learning**
- Algorithm part includes 6 chapters which are relevant to classification, dimension reduction, combination/boost of algorithm, linear regression and clustering and (deep) neural network
- Practical methodology include 2 chapters and application of machine learning include 2 chapters
- Author does NOT give all contents in the order of part, but jump between each part
- **We will read the book by following chapter from chapter, with notification of which type that chapter/circle is in the slide**

Scikit Learn

Scikit Learn: The machine learning package in Python

The user guide could be checked ▸ Link

Which packages to be installed?

If you already have python 2.7 in windows OS/Linux OS, now install pip(Ulf, he recommends python 3.0)

- Linux: pip install -U pip
- Windows: python -m pip install -U pip
- If you require pip updated, go for https://pip.pypa.io/en/stable/installing/

Which Packages to Be Installed?

When pip is ready

- Matrix operating: pip install numpy
- Optimization functionality: pip install scipy
- Data analysis: pip install pandas
- Show your result in figure: pip install matplotlib
- Machine learning algorithm: pip install -U scikit-learn / conda install scikit-learn

Which Packages to Be Installed?

You can also download:Anaconda

- https://www.continuum.io/downloads
- It contains all the packages you need(so you might not need to follow the steps in previous slide)

Which Packages to Be Installed?

Jupyter notebook(Ipython notebook). Your best python IDE for interactive programming and making note

- pip install ipython
- conda update ipython(if you get anaconda)

https://www.youtube.com/watch?v=qb7FT68tcA8

When and How Should We Meet?

- Should we start next week?
- Which day?
- Meet very two weeks?
- How many chapters to study for each circle?
- Should we try to follow all the codes?
- Should we divide into 2 or 3 groups?

After This Book

After reading this book, basic machine learning technologies should be studied. If you want, another book regarding deep neural network(deep learning) could be the next.

References

- Aarti Singh and Barnabas Poczos, Machine Learning Course, Carnegie Mellon University
- Bastian Leibe, Machine Learning Course, RWTH Aachen
- Cristian Sminchisescu, Machine Learning Course, Lund University
- Geert Leus, Alle-Jan van der Veen, Statistical Digital Signal Processing and Modeling Course, Delft University of Technology
- Ioan Tabus, Adaptive Signal Processing, Tampere University of Technology
- Sheng Chen, Advanced Wireless Communications Networks and System Course, Southampton University

Question?