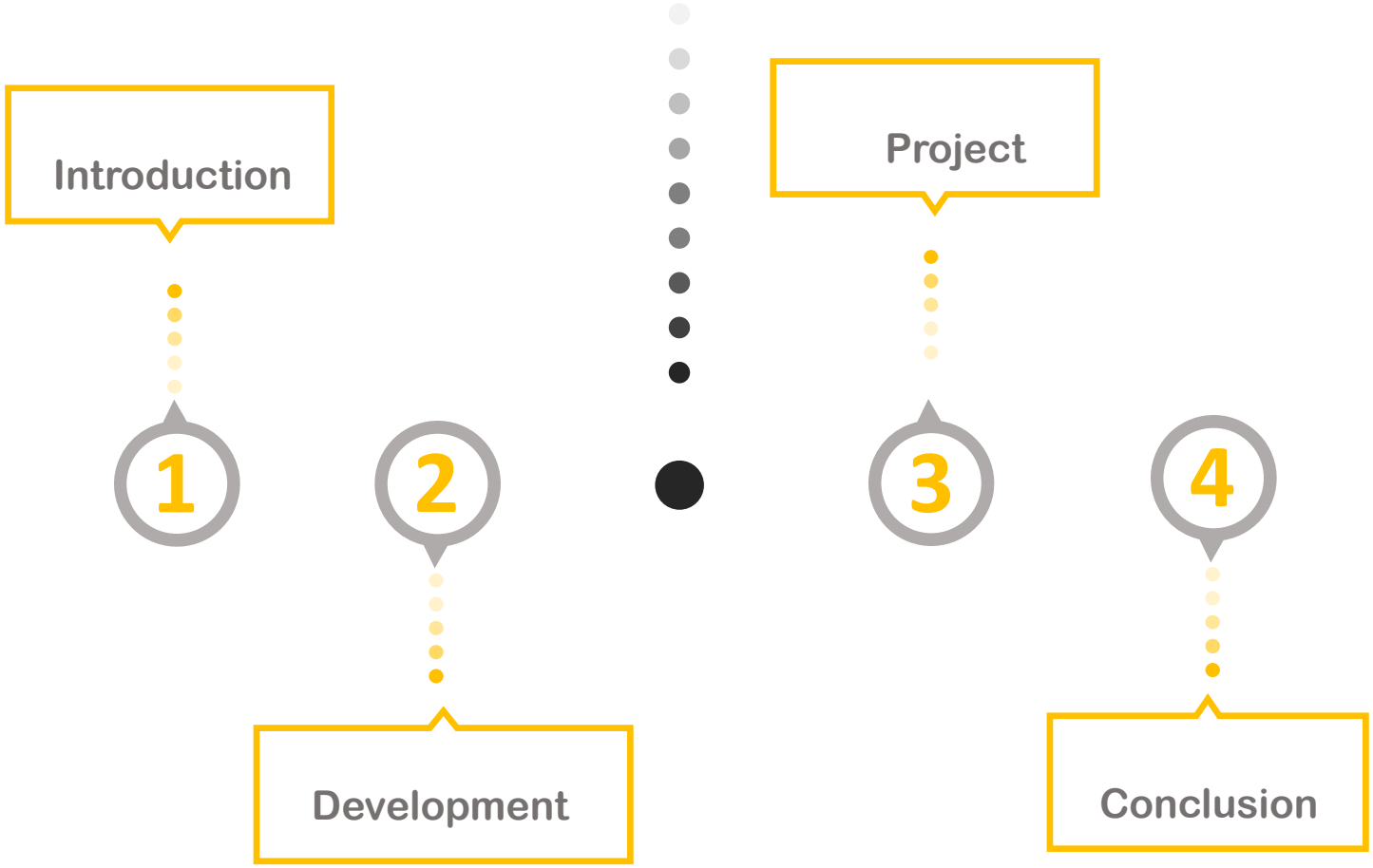
Machine Learning Foundations: A Case Study Approach

Par **Kai HUANG**

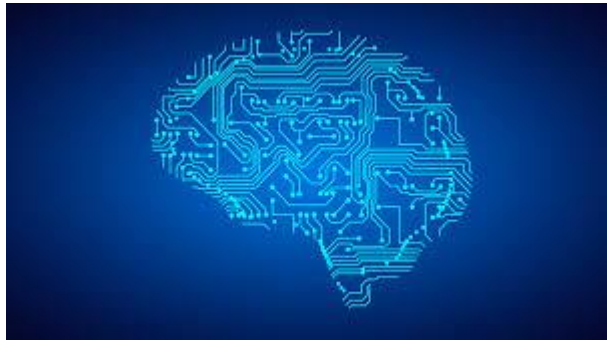
ensiie
Paris-Évry | Strasbourg

CONTENTS



ensiie
Paris-Évry | Strasbourg

Introduction



- A course in University of Washington
- Machine Learning is a subject which is combined by statistic and computer science

Introduction

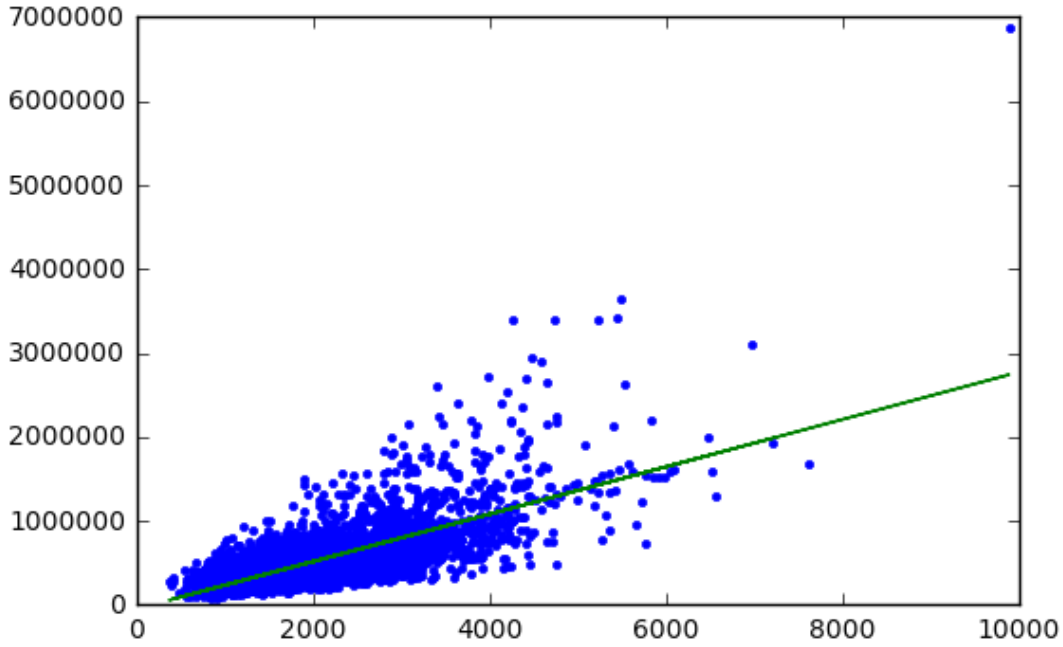
Objective

- Identify potential applications of machine learning in practice.
- Describe the core differences in analyses enabled by regression, classification, and clustering.
- Select the appropriate machine learning task for a potential application.
- Apply regression, classification, clustering, retrieval, recommender systems, and deep learning.
- Represent data as features to serve as input to machine learning models.
- Assess the model quality in terms of relevant error metrics for each task.
- Utilize a dataset to fit a model to analyse new data.
- Implement these techniques in Python.

Development

Regression: Predicting house prices.

- Regression is a statistical process for estimating the relationships among variables.
- How to predict house prices ?
- Linear regression ? Quadratic function ? Polynomial regression?
- Multiple linear regression ?



Development

Regression: Predicting house prices.

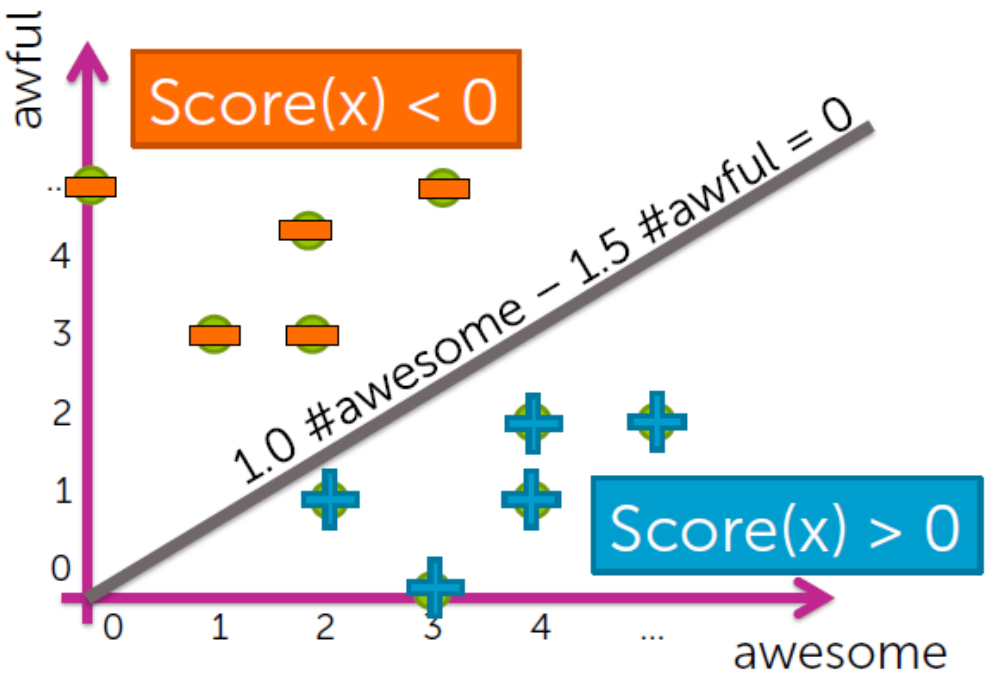
- How to evaluate our model ?
- Residual Squared Error
- Root Mean Squared Error
- Overfitting

	RMSE
With 1 feature:	255191
With 6 features:	179542

Development

Classification: Analysing sentiment

- In machine learning, classification is the problem of identifying to which of a set of categories (sub-populations) a new observation belongs, on the basis of a training set of data containing observations (or instances) whose category membership is known.



Development

Classification: Analysing sentiment

- Binary classification
- Count words
- Give every word a weight
- Calculate the score
- Evaluate the classification model with error and accuracy

Final results

Accuracy: 0.92

Error: 0.08

Development

Clustering and similarity: Retrieving documents

- Term frequency – inverse document frequency(TF-IDF)
- Term frequency (count of the word in the current article)
- Inverse document frequency

  = log ((#docs) / (1 + #docs using word))
- Similarity(doc1, doc2) =

  TF-IDF1 .* TF-IDF2

- Apply 1-nearest neighbour algorithm:
- Search over each article in corpus
- Compute s = similarity (doc1, Query_article)
- If s > Best_s, record Most_similar_article = doc1 and set Best_s = s
- Return Most_similar_article

Development

Clustering and similarity: Retrieving documents

- Clustering is the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense or another) to each other than to those in other groups (clusters).

Development

Clustering and similarity: Retrieving documents

K-means algorithm

- 1.Initialize cluster centers.
- 2. Assign observations to closest cluster center.
- 3. Revise cluster centers as mean of assigned observations.
- 4. Repeat 2 + 3 until convergence

Development

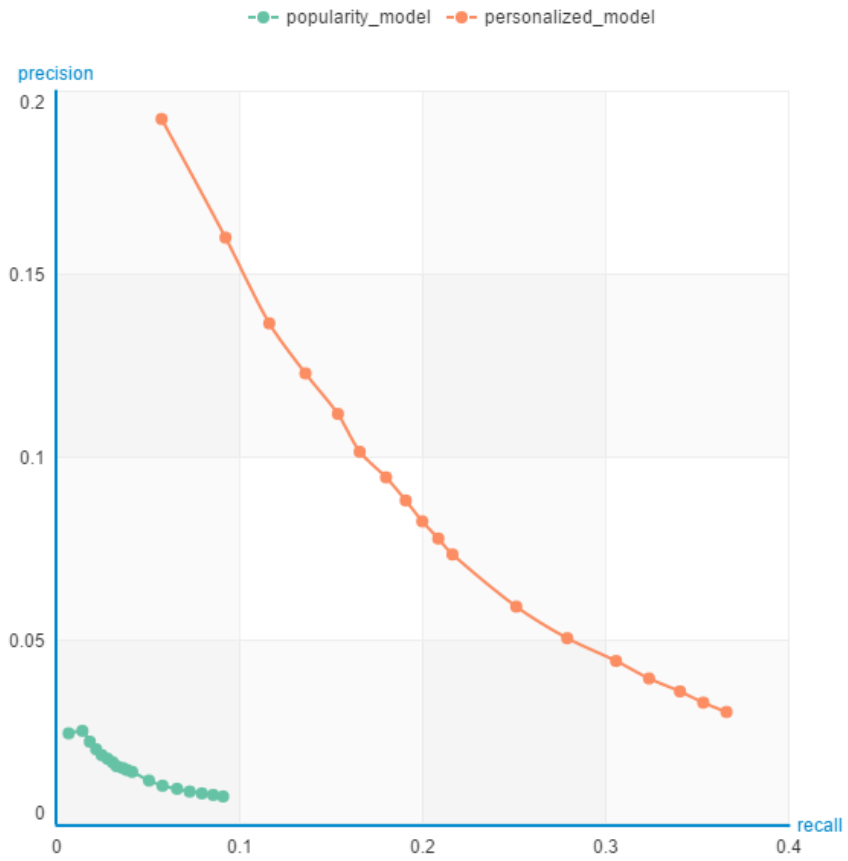
Recommending products

- Collaborative filtering is a method of making automatic predictions(filtering) about the interests of a user by collecting preferences or taste information from many users (collaborating).

- Using collaborative filtering to recommend songs for users.
- Evaluate the model with precision and recall.
- Precision = # (liked and shown) / #shown
- Recall = # (liked and shown) / #liked
- AUC (area under the curve).

Development

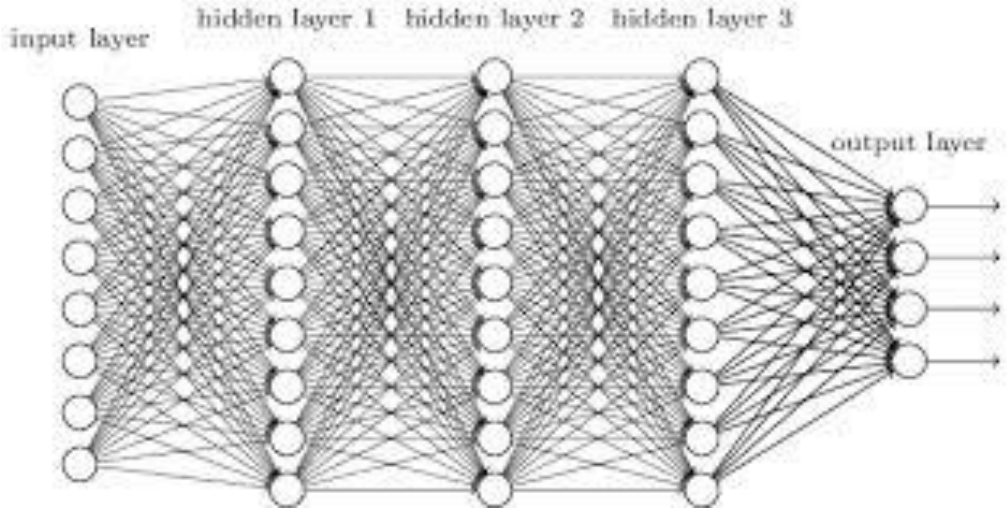
Recommending products



Development

Deep learning: Searching for images

- Deep learning is a branch of machine learning, and it attempts to use high-level abstraction algorithms that involve complex structures or multiple processing layers made up of multiple nonlinear transformations.



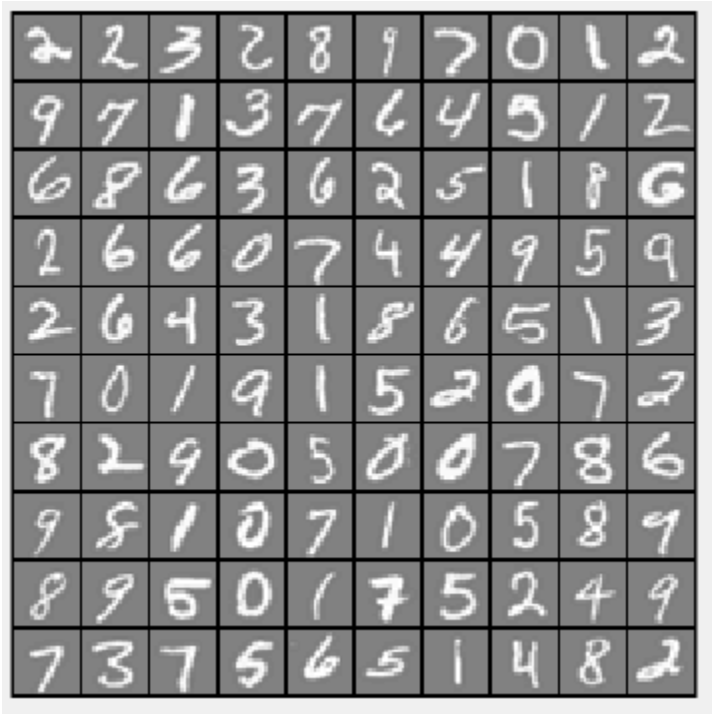
Development

Deep learning: Searching for images

- Extract deep features from images
- Apply the K nearest neighbours algorithm on building a KNN model
- Search similar images with this KNN model

Project

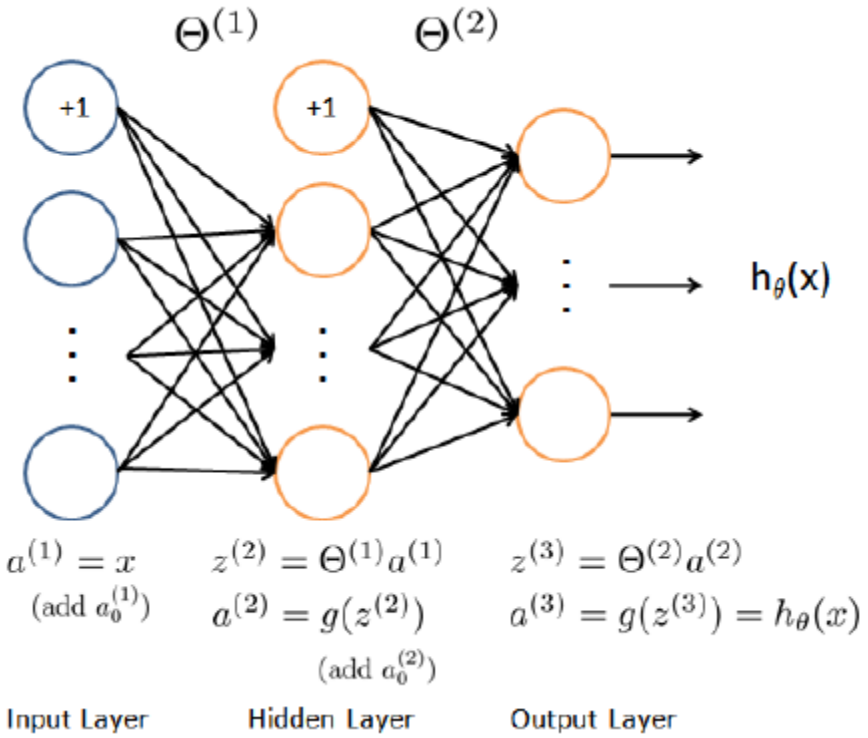
Hand-written digit recognition



- 5000 training sample
- Each : 20 pixel * 20 pixel
- Neural network

Project

Hand-written digit recognition: Forward propagation



$$\mathbf{z}^{(2)} = \Theta^{(1)} * a^{(1)}$$

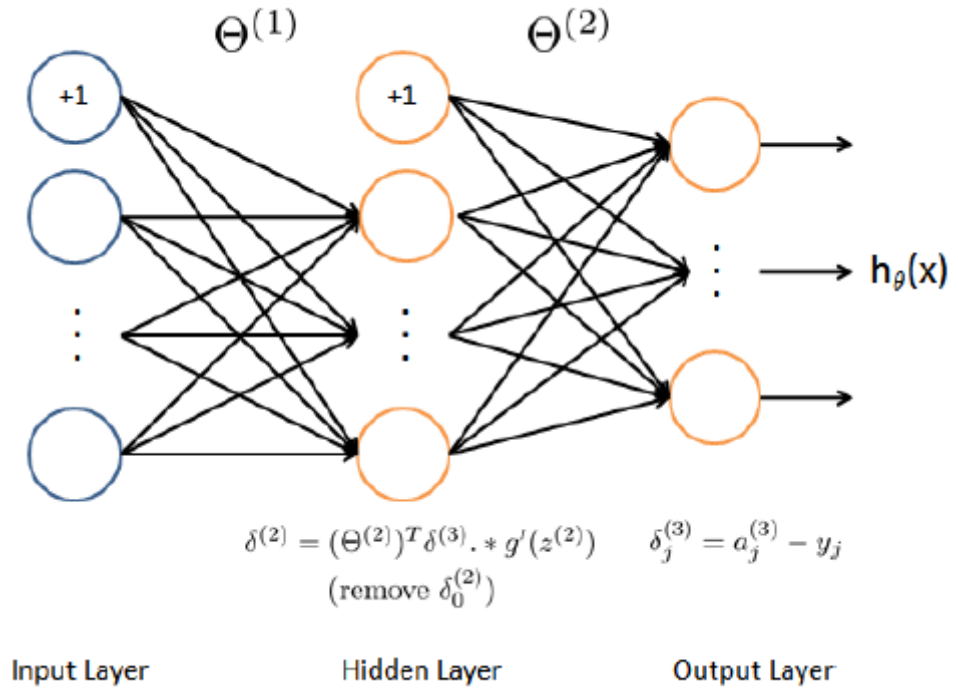
$$\mathbf{a}^{(2)} = g(z^{(2)})$$

$$\mathbf{z}^{(3)} = \Theta^{(2)} * a^{(2)}$$

$$\mathbf{h_\Theta}(x) = a^{(3)} = g(z^{(3)})$$

Project

Hand-written digit recognition: Backward propagation



Training set $(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \ldots, (x^{(m)}, y^{(m)})$

Set $\Delta_{ij}^{(l)} = 0$

for i = 1 to m:

- Set $a^{(1)} = x^{(i)}$;
- Perform forward propagation to compute $a^{(l)}$ (for l = 2, 3, …, L);
- Using $y^{(i)}$, computer $\delta^{(l)} = a^{(l)} - y^{(i)}$;
- Compute $\delta^{(L-1)}, \delta^{(L-2)}, \delta^{(2)}$
- $\Delta_{ij}^{(l)} := \Delta_{ij}^{(l)} + a_j^{(l)} \delta_i^{(l+1)}$;

$D_{ij}^{(l)} := \frac{1}{m}\Delta_{ij}^{(l)} + \lambda\Theta_{ij}^{(l)}$ if y != 0

$D_{ij}^{(l)} := \frac{1}{m}\Delta_{ij}^{(l)}$ if y = 0

The derivate of $J(\Theta)$ is : $D_{ij}^{(l)}$.

Project

Hand-written digit recognition: General

1. Randomly initialize weights.
2. Implement forward propagation to get $h_\Theta(x^{(i)})$ for any $x^{(i)}$.
3. Compute cost function $J(\Theta)$.
4. Implement back propagation to compute partial derivatives of $J(\Theta)$.
5. Use gradient checking to compare partial derivatives of $J(\Theta)$
with the one using numerical estimate of gradient of $J(\Theta)$.
And then, disable gradient checking (this is very important).
6. Use gradient descent to try to minimize $J(\Theta)$as a function of parameter Θ.

And we get an accuracy of 95%.

The accuracy is 95%.

Conclusion

- Machine Learning is a power tool
- Machine learning is widely used
- This course give me a whole impression about machine learning
- Lack of details

Thank you!

ensiie
Paris-Évry | Strasbourg