

The Machine Learning Problem

Risman Adnan
Telkom University
rismanadnan@telyu



Outline

- The Machine Learning Problem
- Type of Machine Learning
- Simple Polynomial Curve Fitting
- Tools and Frameworks
- Homework #1 :

DL adalah ML dengan neural network.

After course you will be able to work with ML and DL frameworks as Data Scientist or AI engineers.

Prerequisites: pelajari python programming language.

Belajar dengan FCP. Kalau fully understand pasti bisa explain. Di ML, harus tahu mathematical model, ..., practice.

Pelajari Caltech Pak Abu Yaseer part 2 (feasible)

The Learning Problem

- Automatic discovery of **regularities** in data through computer algorithms and the use of those regularities to take actions such as *classifying* the data into categories.
- **The Essence of Machine Learning:** Salah satu dari 3 poin di bawah ini tidak dipenuhi, maka bukan ML problem.
 - A pattern exists
 - We cannot pin it down mathematically (no analytical solution)
 - We have data on it We don't need big data. But, enough data.
- **Initial:** Problems that hard for human – but easy for computer.
- **Current:** Problems that hard for computer– but easy for human.
Membuat komputer sepintar manusia.

Kalau kita tahu mathematical solution/rule base, maka itu bukan ML problem.

ML masih mencari cara supaya dengan data yang lebih sedikit dari sekarang, bisa memberikan keluaran yang baik dalam model.

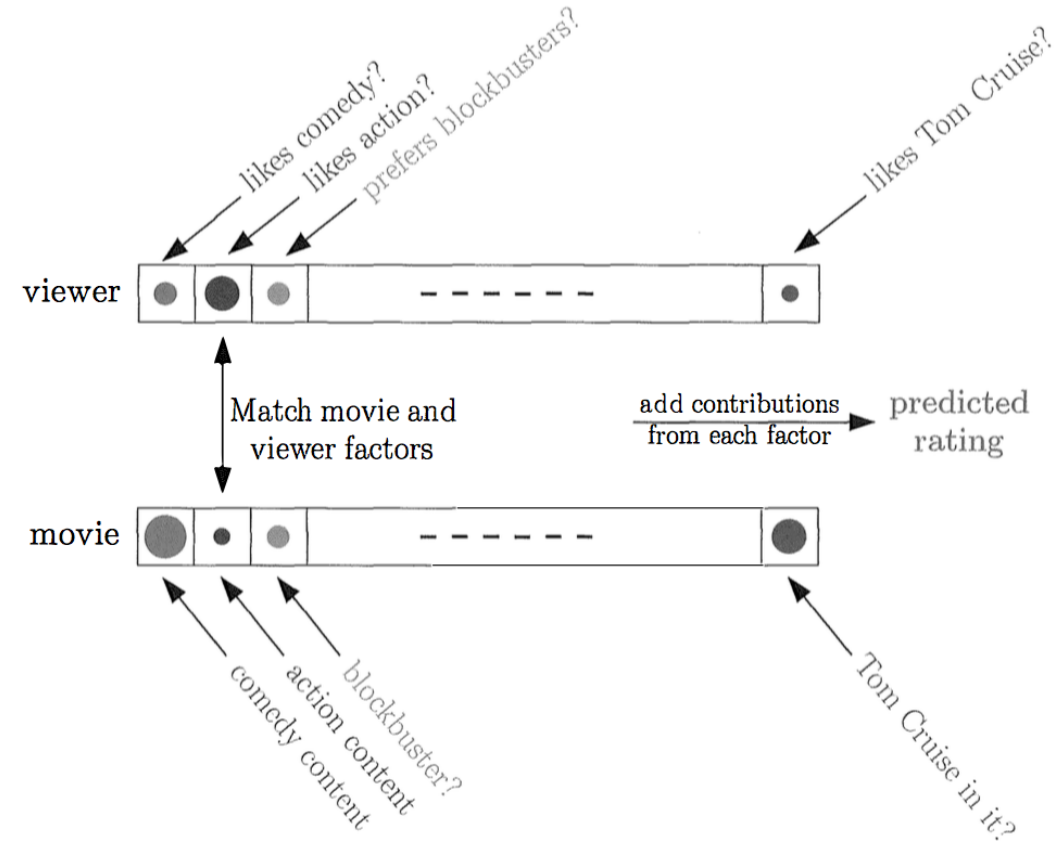
Example: Netflix

- **Problem**

- Predict users rating
- No information about users and movies

- **Dataset**

- 100,480,507 ratings, 480,189 users, 17,770 movies
- Training Data: $\langle \text{user}, \text{movie}, \text{data of grade}, \text{grade} \rangle$
- Test Data 2,817,131: $\langle \text{user}, \text{movie}, \text{data of grade}, ? \rangle$

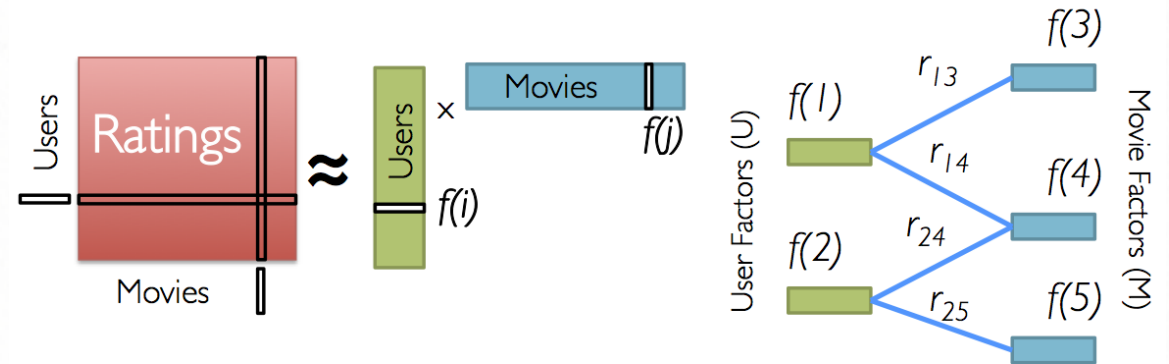


Example: Netflix

Common Techniques:

- Collaborative Filtering
- Matrix Factorization
- Random Initialization
- Iteration to Minimize Error
- Alternating Least Squares

Low-Rank Matrix Factorization:



Iterate:

$$f[i] = \arg \min_{w \in \mathbb{R}^d} \sum_{j \in \text{Nbrs}(i)} (r_{ij} - w^T f[j])^2 + \lambda ||w||_2^2$$

Example: MNIST



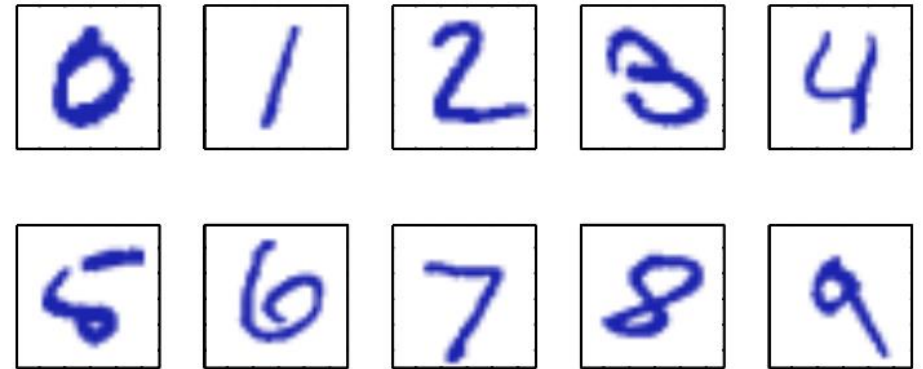
Task

Classify handwritten digits
Build a machine that take x
input and produce digit identity
0,..9.



Dataset

MNIST:
<http://yann.lecun.com/exdb/mnist/>

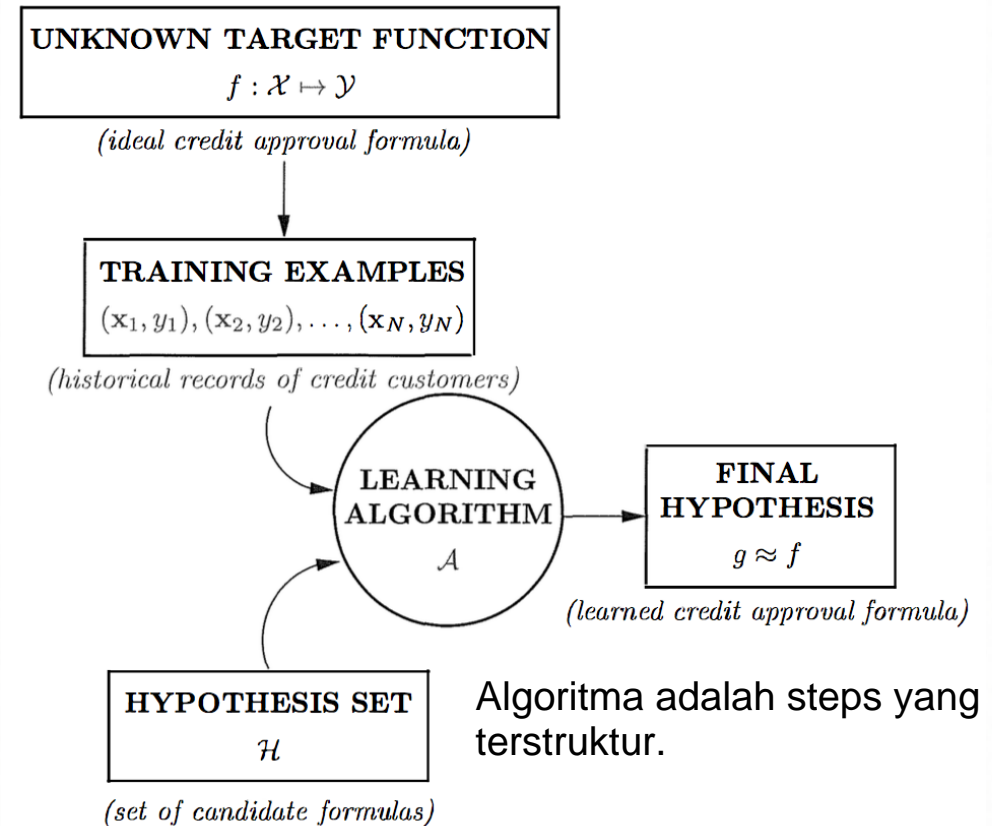


Component of Learning

Metaphor: Credit Card Approval

- Input: \mathbf{x} (*customer application*)
- Output: y (*good/bad customer*)
- Unknown Target Function f
- Data (*historical record of credit customers*)
- Hypothesis Set & Final Hypothesis

Learning Model = Hypothesis Set +
Learning Algorithm



Deep learning ada generative model (tanpa label, bisa belajar).

Machine Learning Paradigm



Supervised Learning: Learning by labeled example

E.g. An email spam detector
We have (**input**, **correct output**), and we can predict (**new input**, **predicted output**)

Amazingly effective if you have lots of data



Unsupervised Learning: Discovering Patterns

E.g. Data clustering
Instead of (**input**, **correct output**), we get (**input**, ?)
Difficult in practices but useful if we lack labeled data



Reinforcement Learning: Feedback & Error

E.g. Learning to play chess
Instead of (**input**, **correct output**), we get (**input**, **only some output**, **grade of this output**)

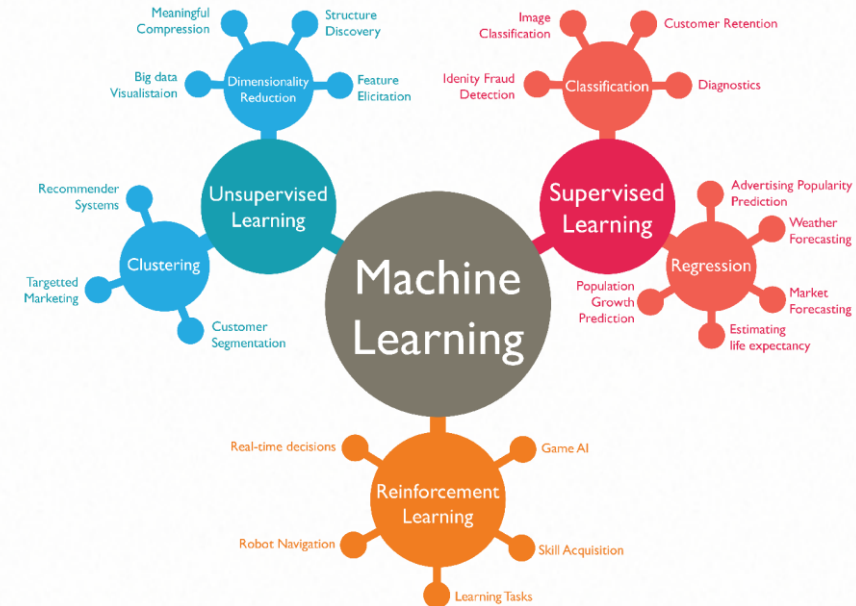
Works well in some domains, becoming more important

RL belajar dari feedback untuk memaksimalkan reward.

The Landscape

- Theory is Required to Guide Engineering
- Model = Hypothesis Set + Algorithm
- Methods = How to make it works well
- Paradigms = The way we see ML problems

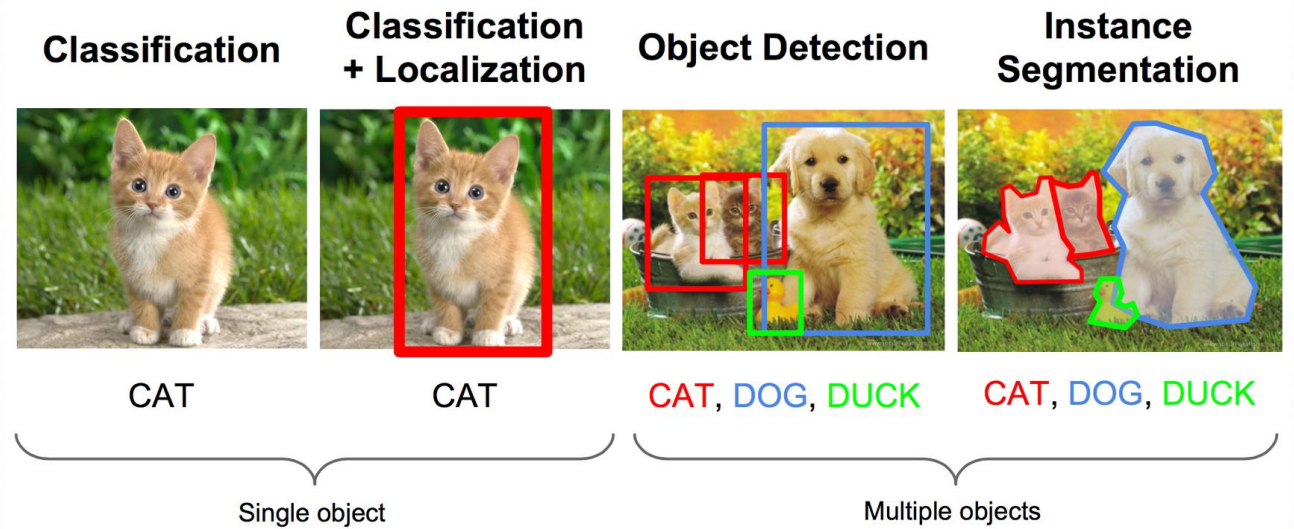
model: hypothesis set (seperti polinomial, dsb) + algorithm
 paradigm: kondisi (supervised kalau labelnya full, RL label datangnya belakangan)
 methods supaya algoritmanya berjalan dengan baik.



THEORY	TECHNIQUES		PARADIGMS
	MODELS	METHODS	
VC	Linear	Regularization	Supervised
Bias-Variance	Neural Networks	Validation	Unsupervised
Complexity	SVM	Aggregation	Reinforcement
Bayesian	Nearest Neighbors	Input Processing	Active
	RBF		Online
	Gaussian Processes		
	SVD		

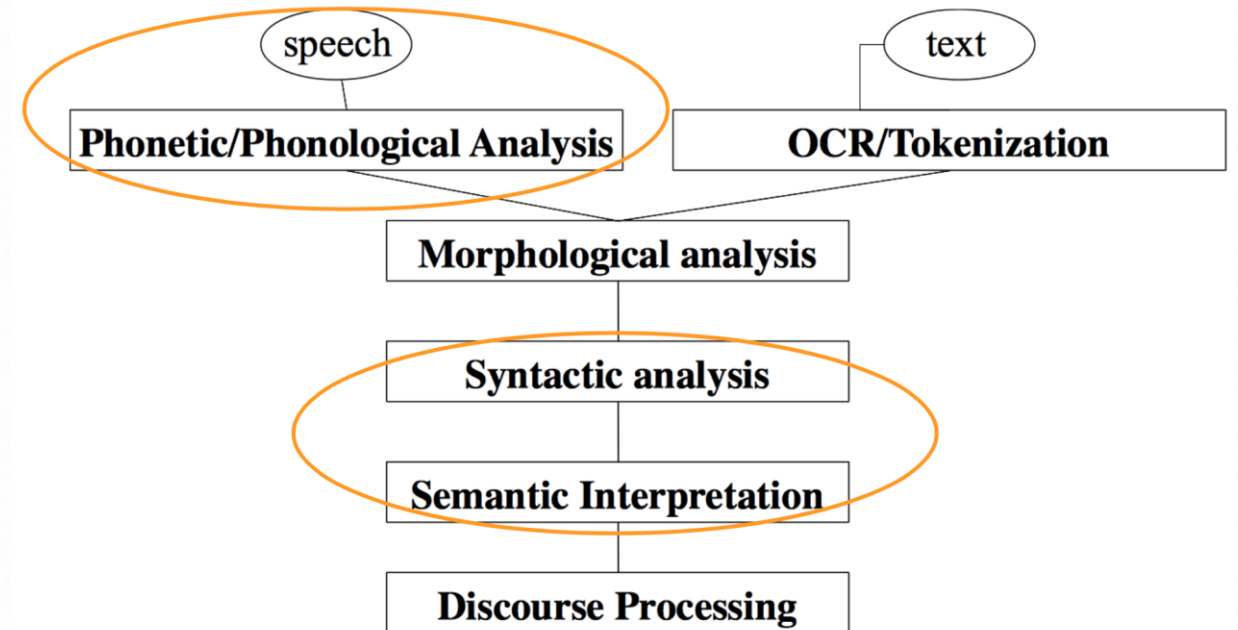
Computer Vision Problem

- Data: Images, Videos
- Pattern: Spatial
- Tasks: Many....



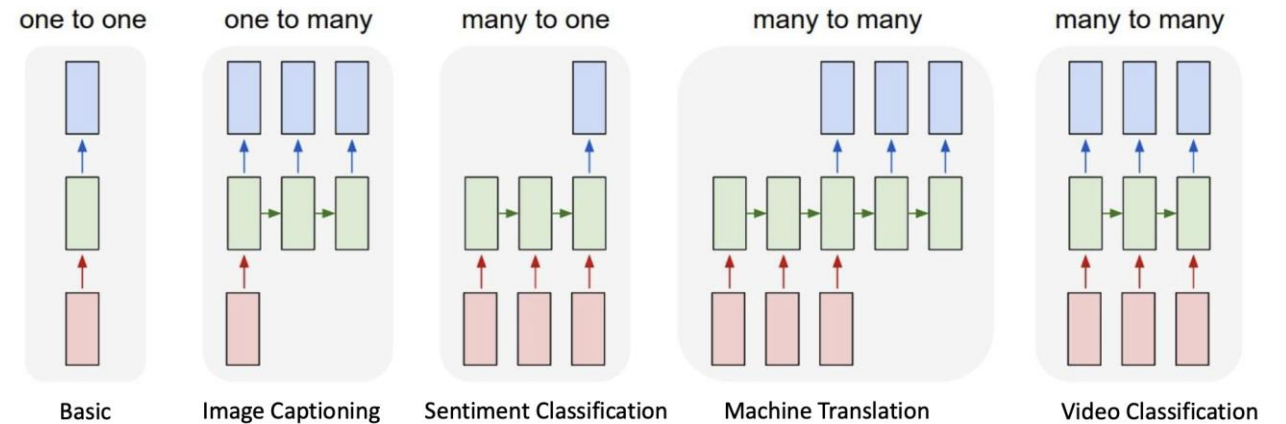
Natural Language Processing Problem

- Data: Speech, Texts
- Pattern: Sequential
- Tasks: Many ...



Sequence to Sequence Problem

- Data: Texts, Speech
- Pattern: Sequential
- Tasks: Many ...



Browse State-of-the-Art

5,407 benchmarks 2,450 tasks 54,456 papers with code

PapersWithCode

Computer Vision



Semantic Segmentation

📈 187 benchmarks
2070 papers with code



Image Classification

📈 260 benchmarks
1808 papers with code



Object Detection

📈 237 benchmarks
1563 papers with code



Image Generation

📈 158 benchmarks
693 papers with code



Denoising

📈 100 benchmarks
667 papers with code

► [See all 1128 tasks](#)

Natural Language Processing



Language Modelling

📈 25 benchmarks
1330 papers with code



Machine Translation

📈 71 benchmarks
1234 papers with code



Question Answering

📈 100 benchmarks
1183 papers with code



Sentiment Analysis

📈 62 benchmarks
750 papers with code



Text Generation

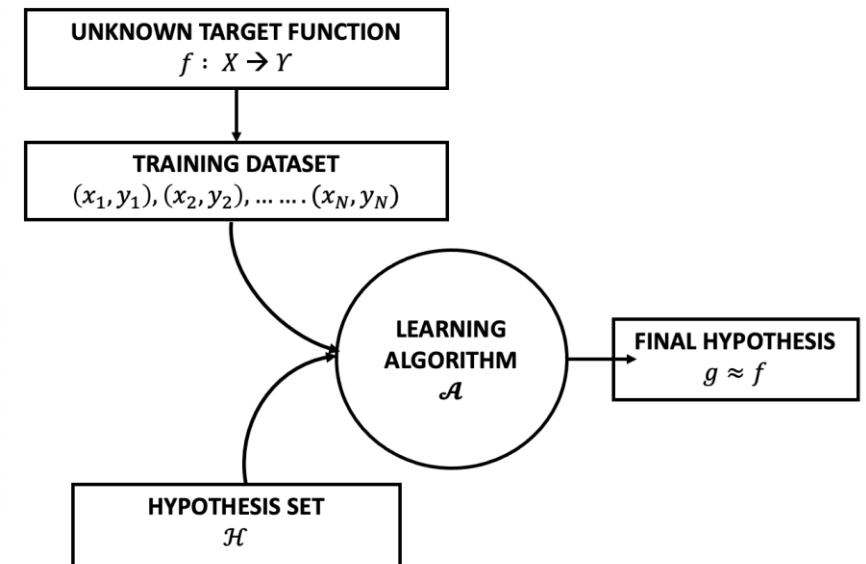
📈 77 benchmarks
582 papers with code

► [See all 473 tasks](#)

Thinking Framework

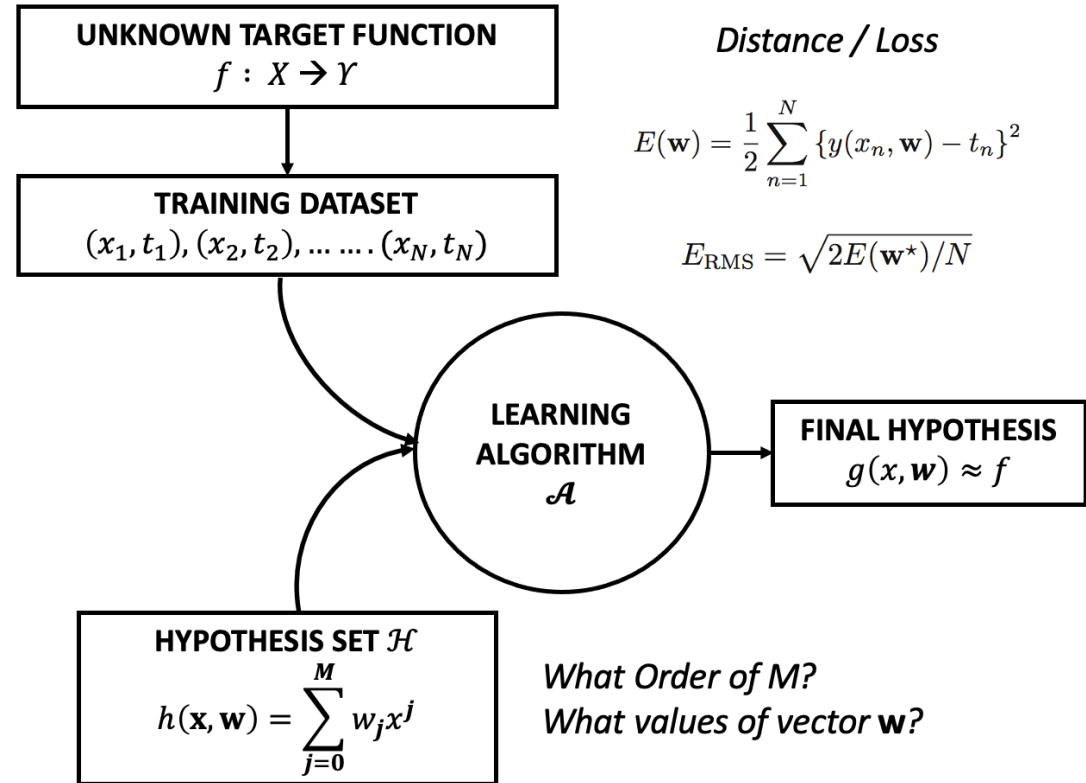
Discriminative Supervised Learning:

- Problem: Estimate “unknown” function that maps data to labels
- This Problem is Translated to:
 - Classification – If Labels are Categorical
 - Regression – If Labels are Continuous
- Approach: Make hypothesis set that potentially has an approximated solution
- Technique: Use algorithm to find an approximated function

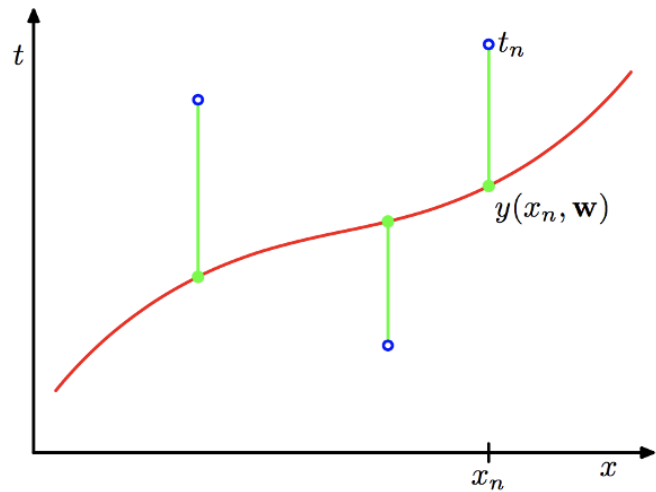
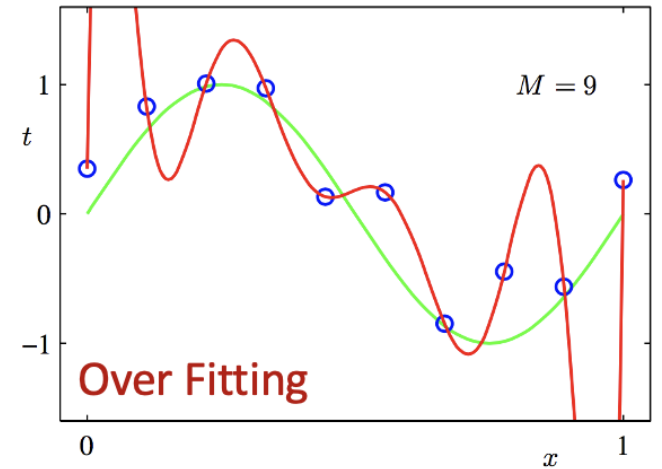
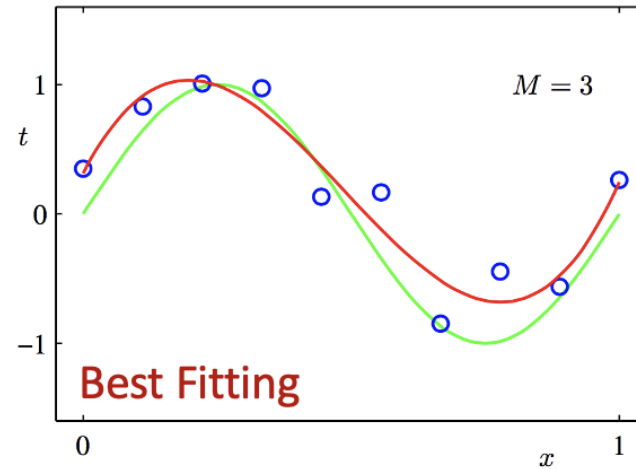
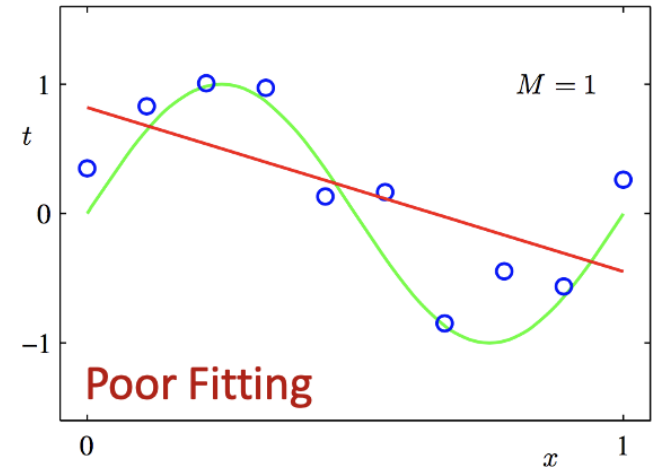
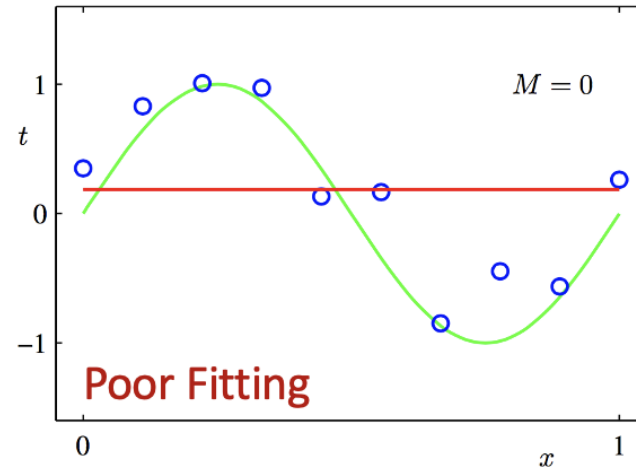
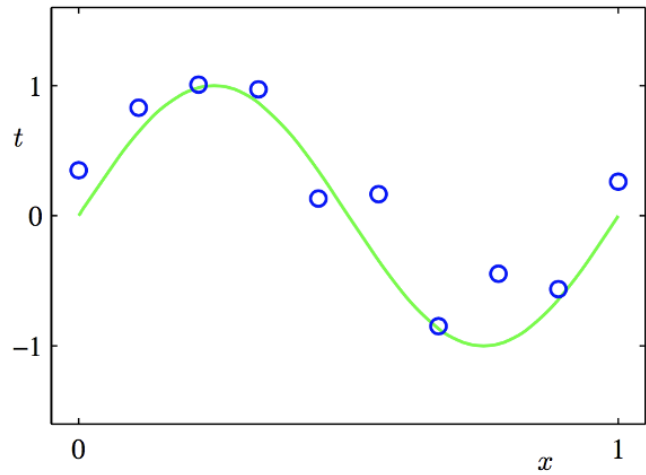


Polynomial Curve Fitting

- Simple Regression Problem
- Linear Model – Polynomial
- Goal: **Good Generalization**
- Learning Algorithms
 - Minimize Square Error Function E with GD
 - Or Minimize Root Mean Square (RMS)



Polynomial Curve Fitting

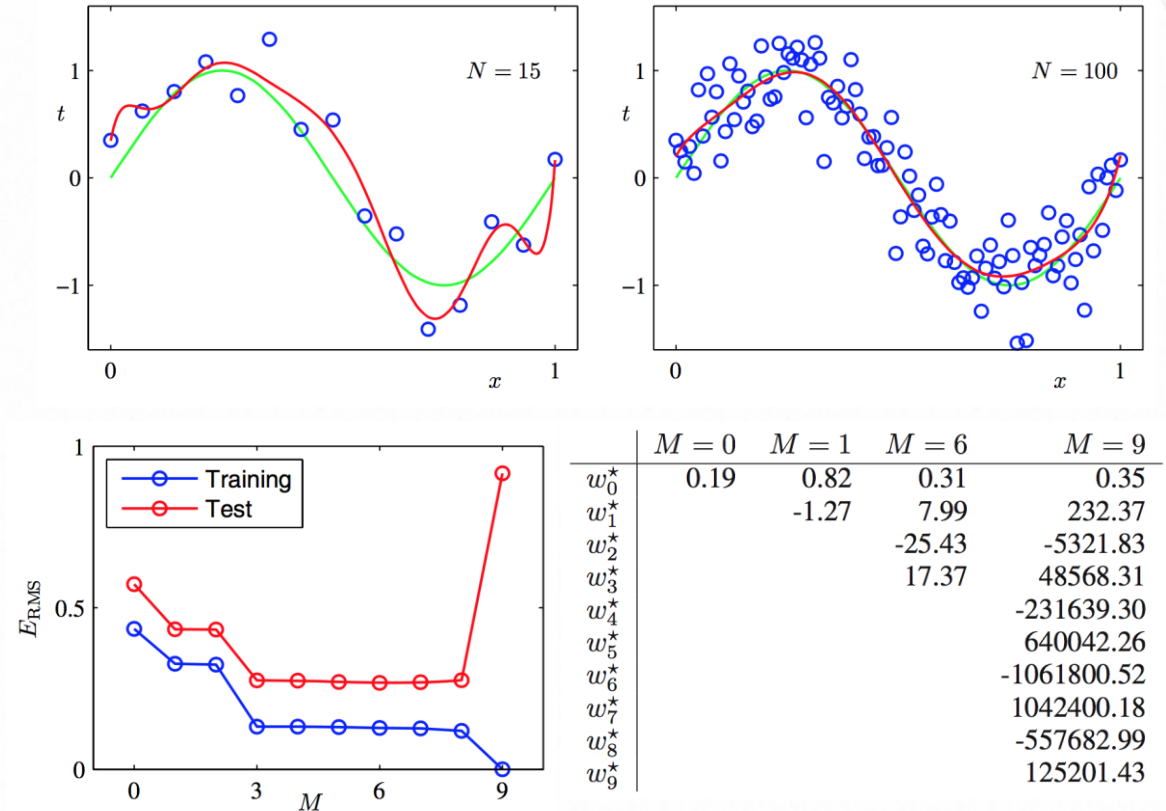


Polynomial Curve Fitting

- RMS for Equal Footing & Same Scale
- Min Training & Test Error ($3 \leq M \leq 8$)
- Wild Oscillation for Test Data at $M \geq 9$
- Error = Zero $M \geq 9$ (Over-fitting)
- Add More Dataset Get Less Over-fitting.

Ada yang bilang handle overfit yaitu tambahkan dengan data.

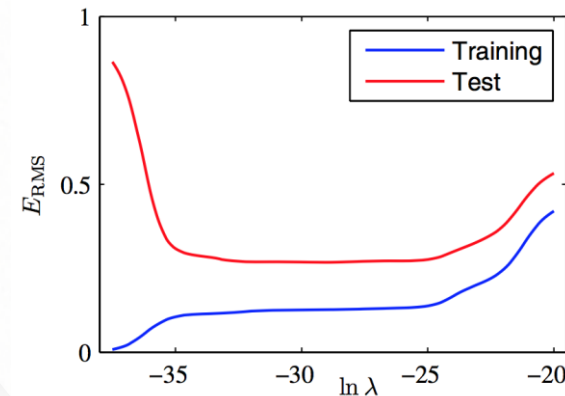
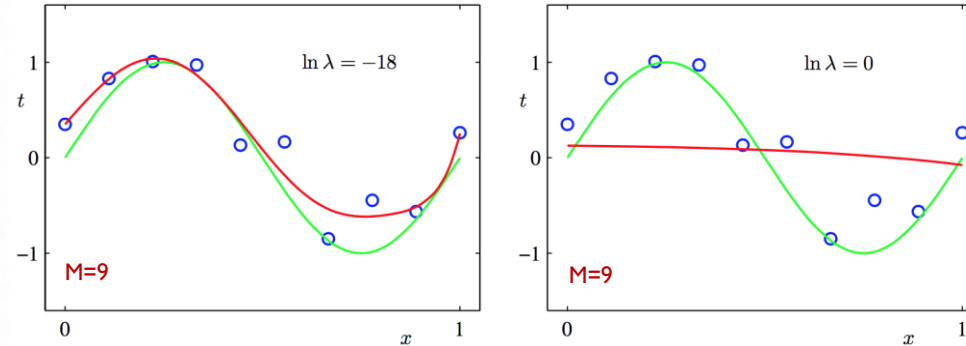
Selain itu, bisa gunakan regularisasi untuk mengatasi overfitting.



Polynomial Curve Fitting

- Regularization for Overfitting
- Penalty Term to E to Regulate Value of Parameters
- Quadratic Regularizer = Ridge Regression

$$\tilde{E}(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\lambda}{2} \|\mathbf{w}\|^2$$



	$\ln \lambda = -\infty$	$\ln \lambda = -18$	$\ln \lambda = 0$
w_0^*	0.35	0.35	0.13
w_1^*	232.37	4.74	-0.05
w_2^*	-5321.83	-0.77	-0.06
w_3^*	48568.31	-31.97	-0.05
w_4^*	-231639.30	-3.89	-0.03
w_5^*	640042.26	55.28	-0.02
w_6^*	-1061800.52	41.32	-0.01
w_7^*	1042400.18	-45.95	-0.00
w_8^*	-557682.99	-91.53	0.00
w_9^*	125201.43	72.68	0.01

Tools and Frameworks

- We Will Use Scikit-Learn Framework & Collab
- Simple and Efficient ML Tools (Complete?)
- Important Lib: Pandas, NumPy, and Matplotlib
- **Data Preprocessing with Scikit-Learn:**
 - Standardization, or Mean Removal and Variance Scaling
 - Non-linear Transformation
 - Normalization
 - Binarization
 - Encoding Categorical Features
 - Imputation of Missing Values
 - Generating Polynomial Features
 - Custom Transformers



The logo for Kaggle, featuring the word "kaggle" in a bold, blue, sans-serif font, positioned on a light blue background with a green circle above it.

Homework: House Prediction

Watching Youtube:

- Decision Tree: https://www.youtube.com/watch?v=_L39rN6gz7Y
- Random Forests: https://www.youtube.com/watch?v=j4Vdy0Vc_xQ

Hacking The Codes: <https://www.kaggle.com/learn/intro-to-machine-learning>

1. How Models Work
2. Basic Data Exploration
3. Decision Tree Model
4. Model Validation
5. Underfitting and Overfitting
6. Random Forest

The Kaggle logo is displayed in a light blue, lowercase sans-serif font. It is positioned within a large, dark blue rounded rectangle that has a fine white dot pattern. The background of the slide is a dark teal color with abstract geometric shapes in lighter shades of blue and green on the right side.

Open
Discussion
Every Tuesday
Nite 19.00PM