Week 05

Multilayer Perceptron 2 (regularization)

Prof. Ph.D. Woo Youn Kim

Chemistry, KAIST

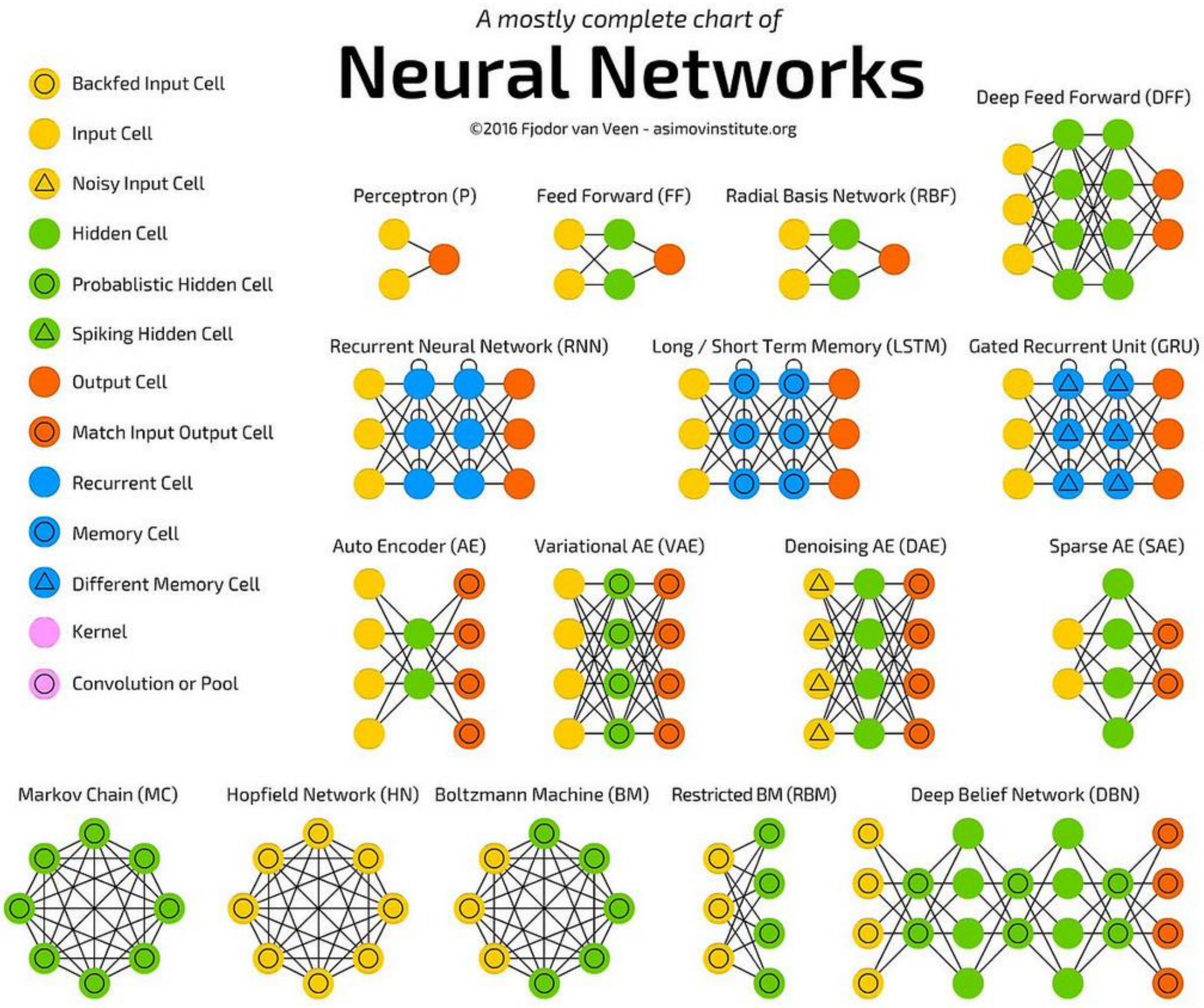
Goal

5주	주제	Deep learning & multilayer perceptron (MLP)
	목표	Understanding the perceptron concept and a basic principle of deep learning
	내용	Universal approximation theorem backpropagation, vanishing gradient, activation function, ReLU
6주	주제	Multilayer perceptron 2
	목표	Knowing various issues on MLP and techniques to resolve them
	내용	Overfitting, regularization, dropout, batch normalization, cross validation
7주	주제	Convolutional Neural Network (CNN) & SMILES
	목표	Understanding CNN and molecular representation with SMILES
	내용	Convolution, receptive field, stride, pooling Supervised learning of Log P and TPSA

KAIST

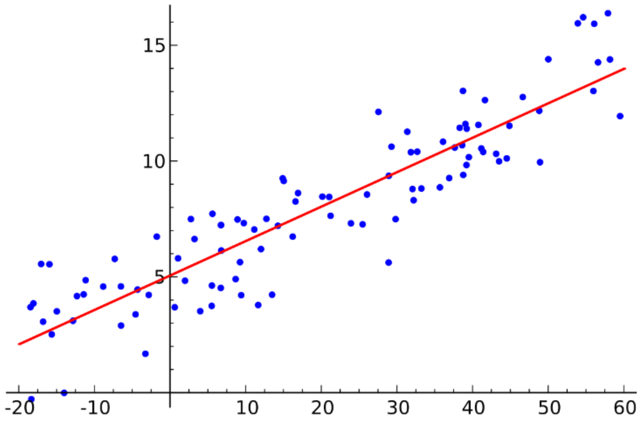
Contents

- Types of deep learning
- Generalization
- Model capacity
- Regularization
  - Data augmentation
  - Cross validation
  - L1,L2 regularization
  - Dropout



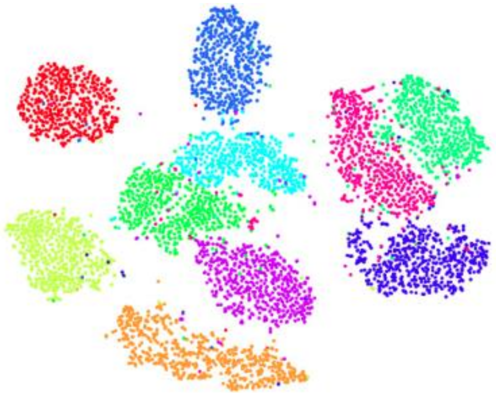
http://www.asimovinstitute.org/neural-network-zoo/

Types of deep learning



- **Supervised Learning: classification or regression**

  The network makes its guesses, then compare its answers to the known “correct” ones and make adjustments according to its errors.



no labeling

- **Unsupervised Learning: clustering**

  Searching for a hidden pattern in a data set without known answers.

- **Reinforcement Learning: game, robotics, self-driving car, etc.**

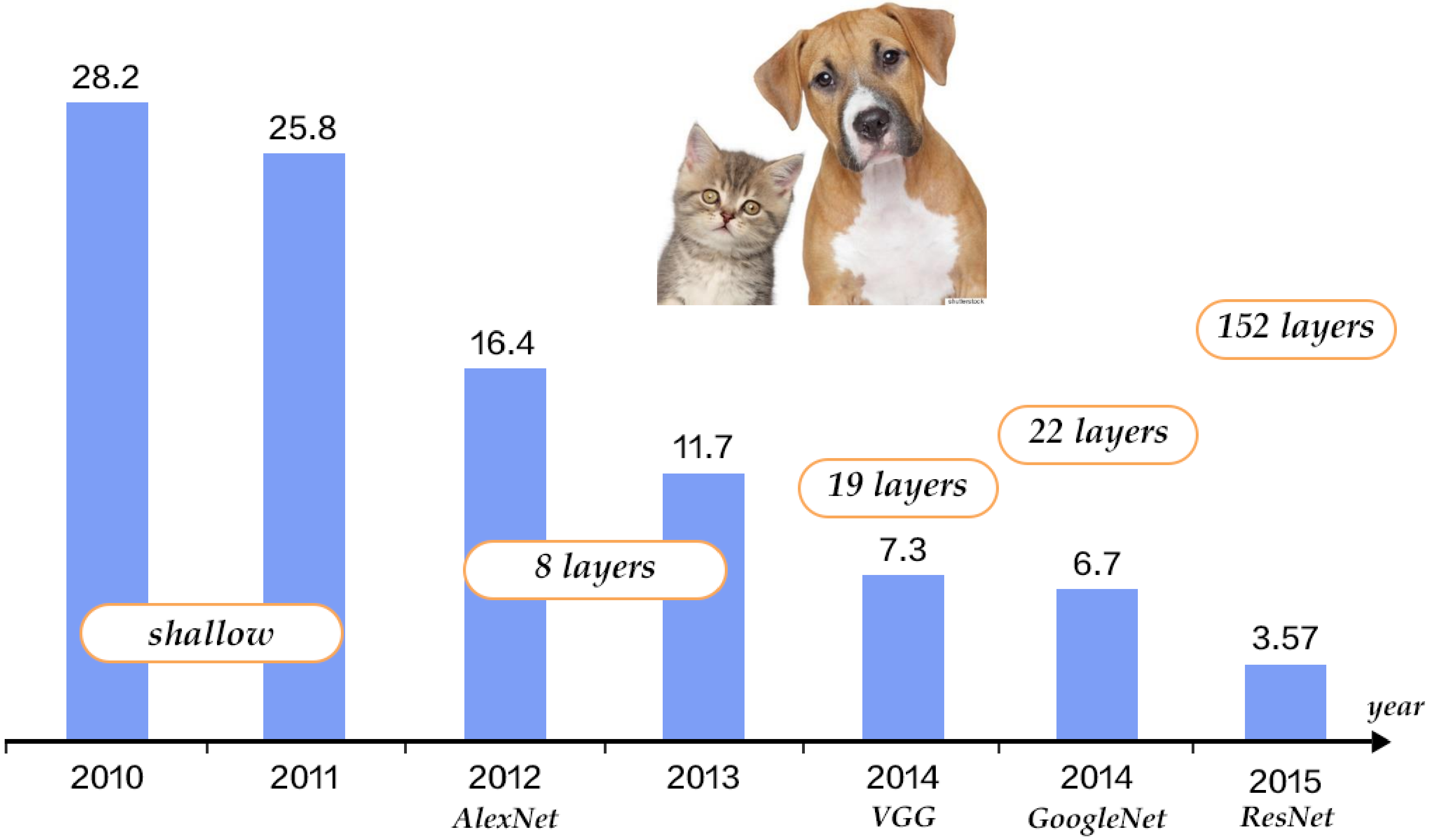
  A strategy built on observation.

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

Leap of deep learning: supervised learning

- **ILSVRC (ImageNet Large Scale Visual Recognition Challenge) Winners**

: 2012년 이후로 모든 대회 1등은 '**Deep convolutional networks**' 를 사용하였다.



2012년을 기점으로 Deep Learning은 전세계에서 활발하게 연구되기 시작

KAIST

Leap of deep learning: unsupervised learning

Generative model

1 **Upload photo**

The first picture defines the scene you would like to have painted.

2 **Choose style**

Choose among predefined styles or upload your own style image.

3 **Submit**

Our servers paint the image for you. You get an email when it's done.

- 손석희 아나운서의 목소리를 만들어내는 인공지능 모델

(박근혜 전 대통령, 문재인 대통령도 되었었는데, 현재 삭제된 상태)

: https://carpedm20.github.io/tacotron/

KAIST

Leap of deep learning: reinforcement learning

✓ **왜 강화 학습인가?**

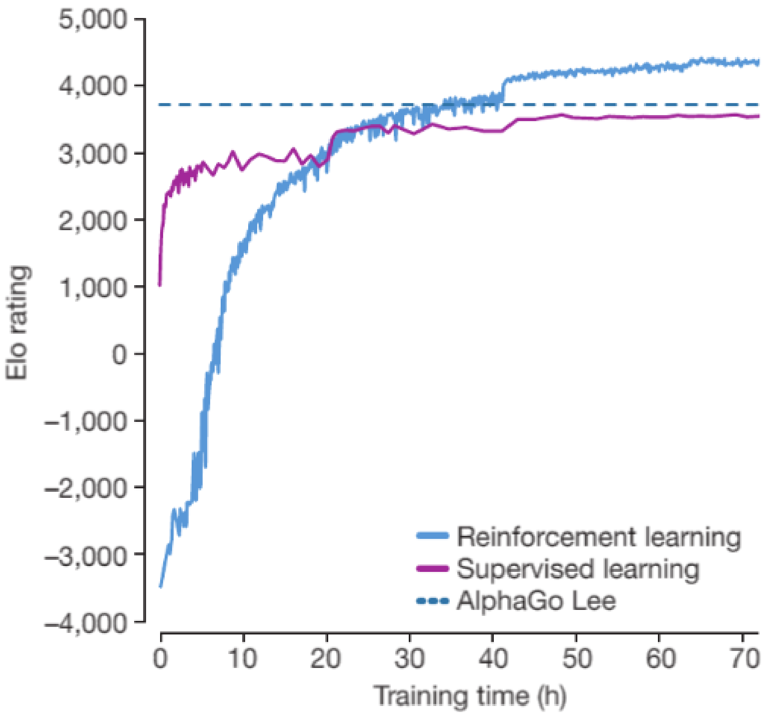
- 특징: 반복된 경험을 통해 정답(지도) 없이 스스로 학습
- 알파고의 등장으로 각광 받는 최신 인공지능 기술: 자율주행, 언어학습 등
- AlphaGoZero: 강화 학습만으로 이전 알파고 이김
- https://www.youtube.com/watch?v=KJ15iGGJFvQ

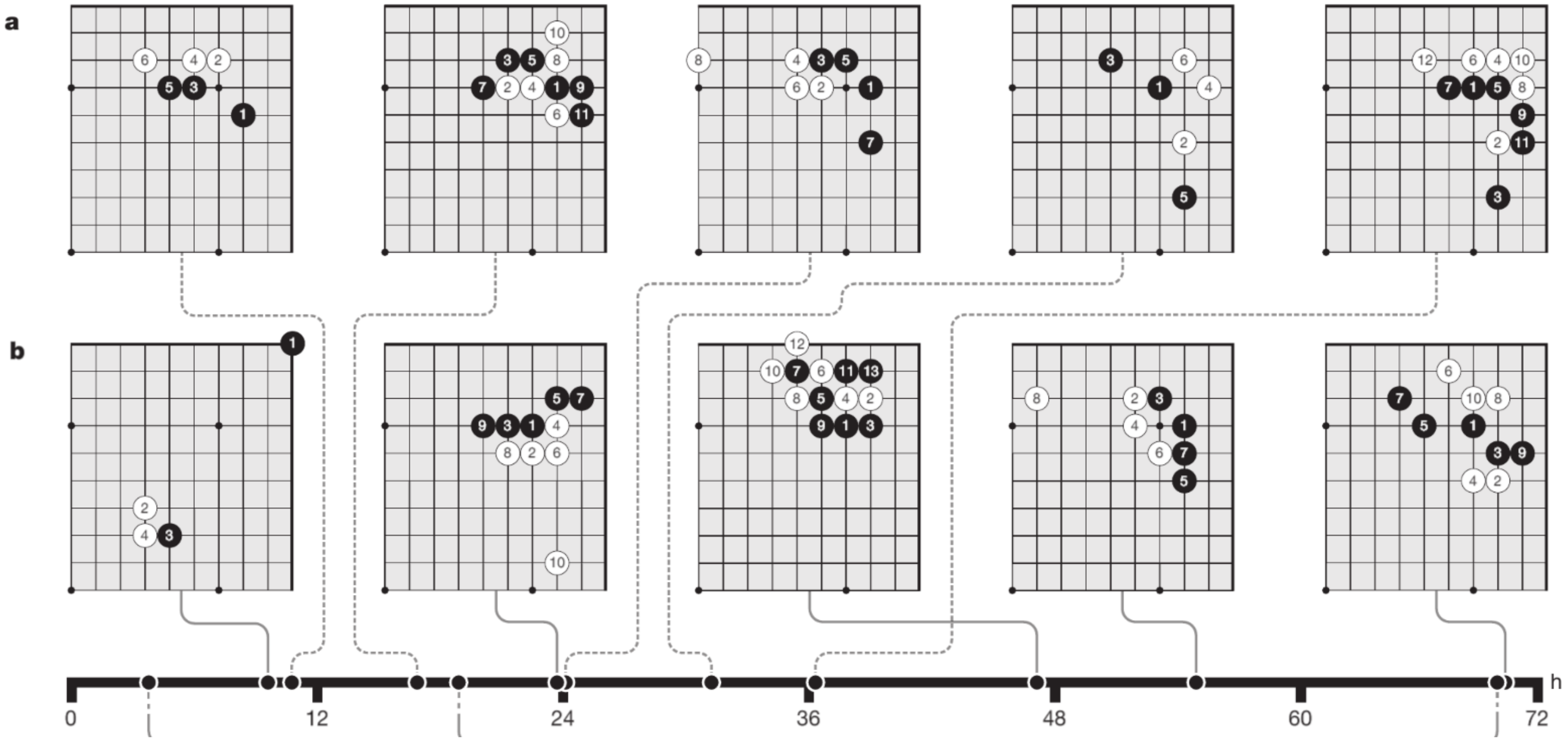
ARTICLE

doi:10.1038/nature24270

Mastering the game of Go without human knowledge

David Silver[1]*, Julian Schrittwieser[1]*, Karen Simonyan[1]*, Ioannis Antonoglou[1], Aja Huang[1], Arthur Guez[1], Thomas Hubert[1], Lucas Baker[1], Matthew Lai[1], Adrian Bolton[1], Yutian Chen[1], Timothy Lillicrap[1], Fan Hui[1], Laurent Sifre[1], George van den Driessche[1], Thore Graepel[1] & Demis Hassabis[1]





Big guys in deep learning

Andrew

- Stanford university
- Google Brain(2011 ~ 2012)
- Baidu (2014~2017.3)

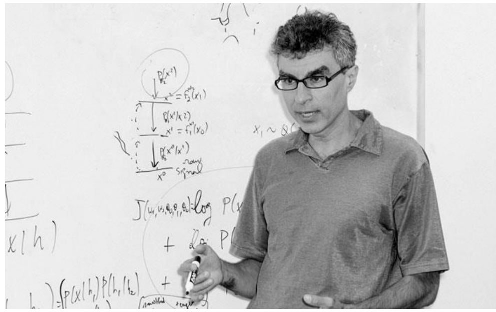
Geoffrey

- University of Toronto
- Google
- BPNN, RBM, Autoencoder, …
- AlexNet

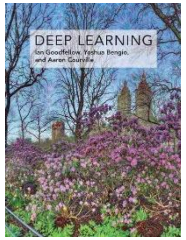
Yann

- New York University
- Postdortoral student of Geoff
- Facebook
- CNN - LeNet

Yoshua Bengio



- Universite de Montreal
- CIFAR



Ian Goodfellow

- Google Brain
- Open AI
- GAN

And also you can be …

KAIST

Source

Deep Learning

An MIT Press book

Ian Goodfellow and Yoshua Bengio and Aaron Courville

Exercises Lectures External Links

The Deep Learning textbook is a resource intended to help students and practitioners enter the field of machine learning in general and deep learning in particular. The online version of the book is now complete and will remain available online for free.

The deep learning textbook can now be ordered on Amazon.

https://www.deeplearningbook.org/

In literature

Chem Sci. 9, 513 (2018)

3.5.1 Logistic regression. Logistic regression models (Logreg) apply the logistic function to weighted linear combinations of their input ~~features to obtain~~ model predictions. It is often common to use regularization to encourage learned weights to be sparse.[77] Note that logistic regression models are only defined for classification tasks.

3.5.2 Support vector classification. Support vector machine (SVM) is one of the most famous and widely-used machine learning method.[78] As in classification task, it defines a decision plane which separates data points of different class with maximized ~~margin. To fu~~rther increase performance, we incorporates regularization and a radial basis function kernel (KernelSVM).

ACS Cent. Sci. 4, 268 (2018)

9 heavy atoms[31] and another with 250 000 drug-like commercially available molecules extracted at random from the ZINC database.[32] We performed random optimization over hyperparameters specifying the deep autoencoder architecture and training, such as the choice between a recurrent or convolutional encoder, the number of hidden layers, layer sizes, regularization, and learning rates. The latent space representations for the QM9 and ZINC data sets had 156 dimensions and 196 dimensions, respectively.

arXiv:1706.04223v3. 2018

The model consists of a discrete autoencoder regularized with a prior distribution,

$$\min_{\phi,\psi} \quad \mathcal{L}_{\text{rec}}(\phi,\psi) + \lambda^{(1)} W(\mathbb{P}_Q, \mathbb{P}_{\mathbf{z}})$$

Generalization

Generalization

- The central challenge in machine learning is that our algorithm must perform well on new, previously unseen inputs
- The ability to perform well on previously unobserved inputs is called generalization.
- **Training error**: measure on the training set ➔ optimization problem to reduce the training error

$$\frac{1}{m^{(\text{train})}}||\boldsymbol{X}^{(\text{train})}\boldsymbol{w}-\boldsymbol{y}^{(\text{train})}||_2^2$$

- **Test error** or **generalization error** ➔ separating machine learning from optimization

$$\frac{1}{m^{(\text{test})}}||\boldsymbol{X}^{(\text{test})}\boldsymbol{w}-\boldsymbol{y}^{(\text{test})}||_2^2$$

- The generalization error is defined as the expected value of the error on a new input

Generalization

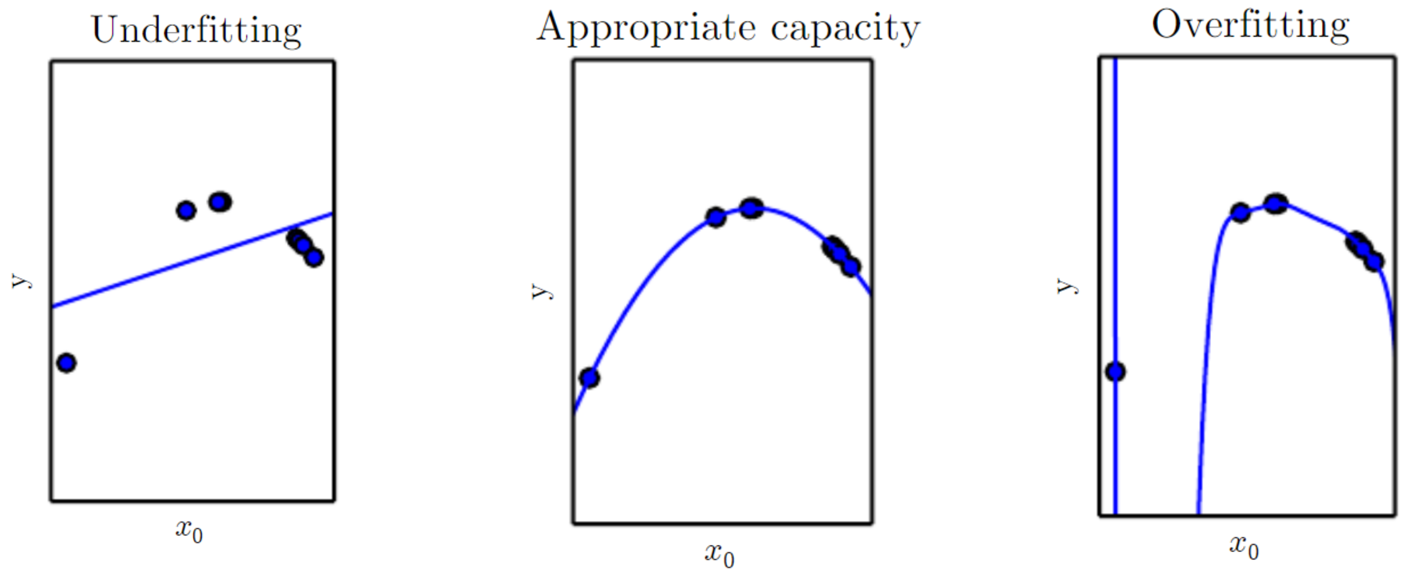
- **Generalization of a model**: making an algorithm that perform well on new inputs, not just on the training data, i.e., there is no universal model working for all tasks

- The factors determining how well a machine learning algorithm will perform are its ability to
  1. Make the training error small.
  2. Make the gap between training and test error small.

- The two central challenges in machine learning

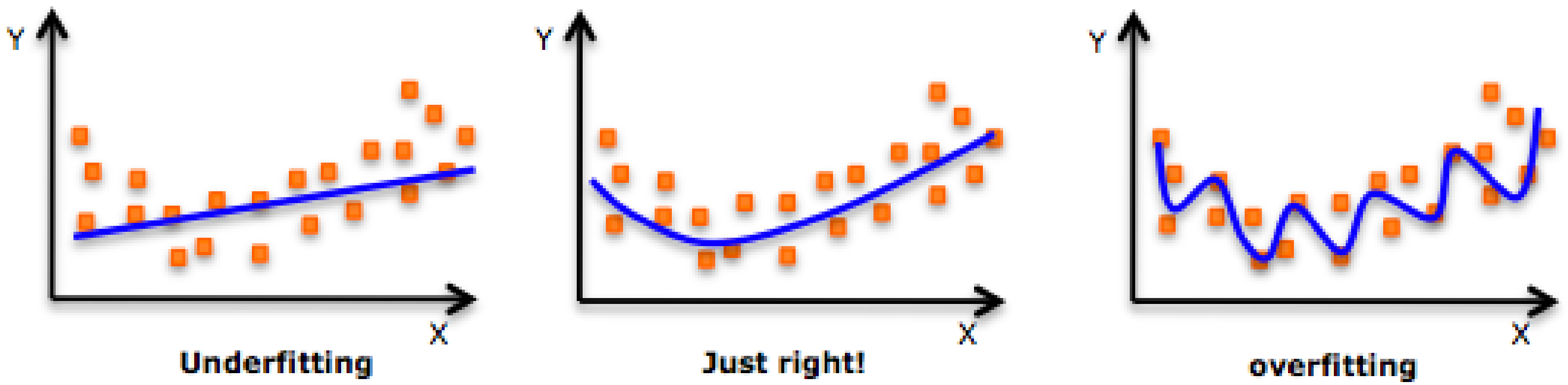
Underfitting & Overfitting

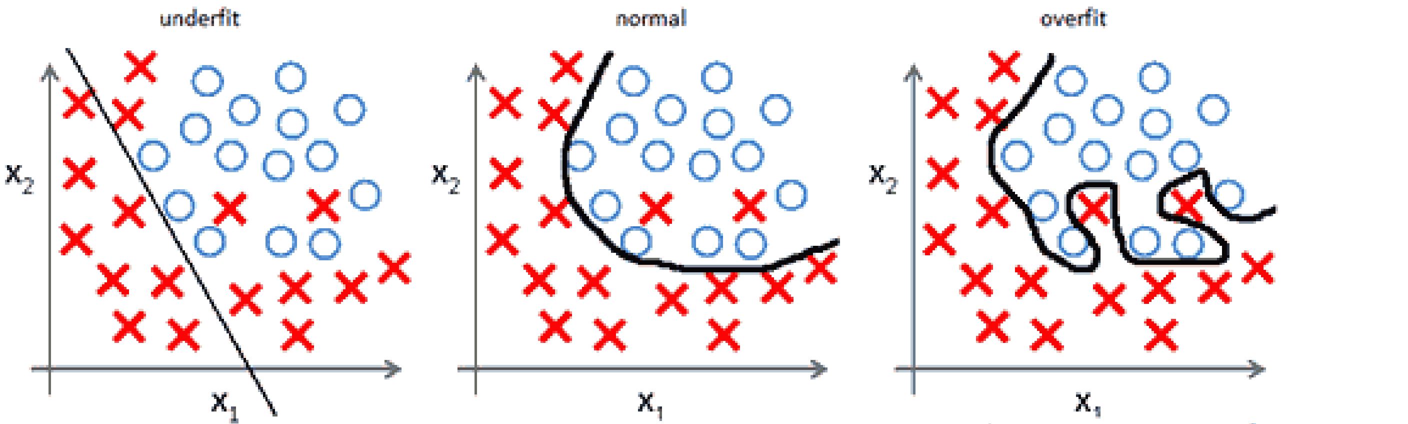
- **Underfitting** occurs when the model is not able to obtain a sufficiently low error value on the training set.
- **Overfitting** occurs when the gap between the training error and test error is too large
- The main challenge is to find a **right model complexity** for a given task



KAIST

More examples





$g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$

$g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2 + \theta_5 x_1 x_2)$

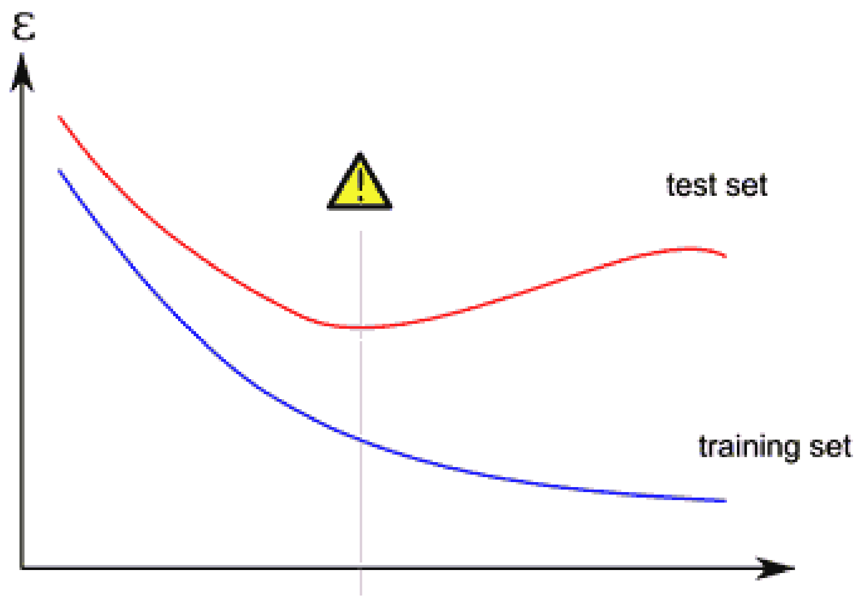
$$g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2 + \theta_5 x_1 x_2 + \theta_6 x_1^2 x_2 + \theta_7 x_1 x_2^2 + \theta_8 x_1^2 x_2^2 + \theta_9 x_1^3 + \dots)$$

Diagnosis of overfitting

Best way to see if you overfit:

- split data in training (~70%) and test set (~30%)
- train the model on the training set
- evaluate the model on the training set
- evaluate the model on the test set
- generalization error: difference between them, measures the ability to generalize

Choose examples for training/testing sets randomly



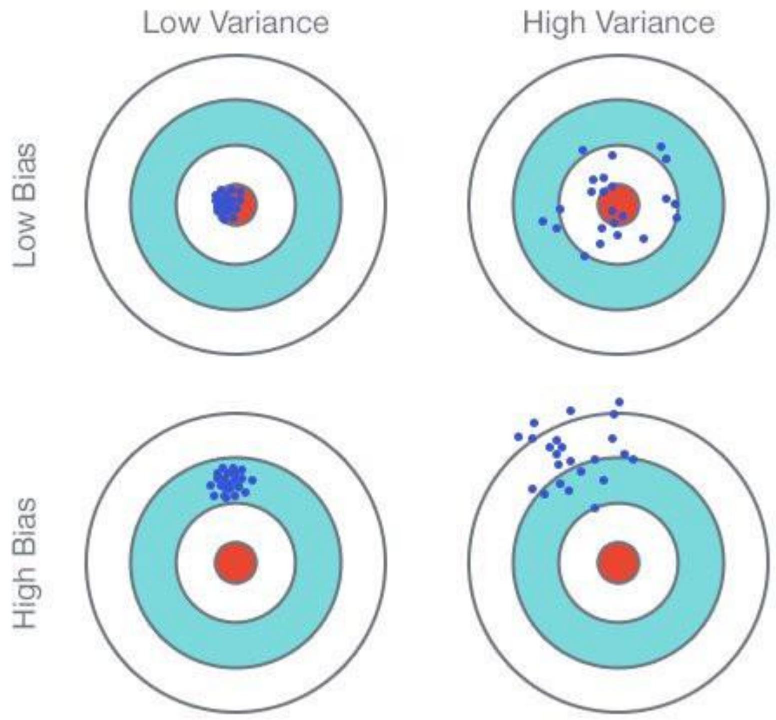
low error on the training data, but high on the testing data
➔ overfitting

Variance & Bias

Generalization error can be decomposed into bias and variance.

- **bias**: tendency to constantly learn the same wrong thing

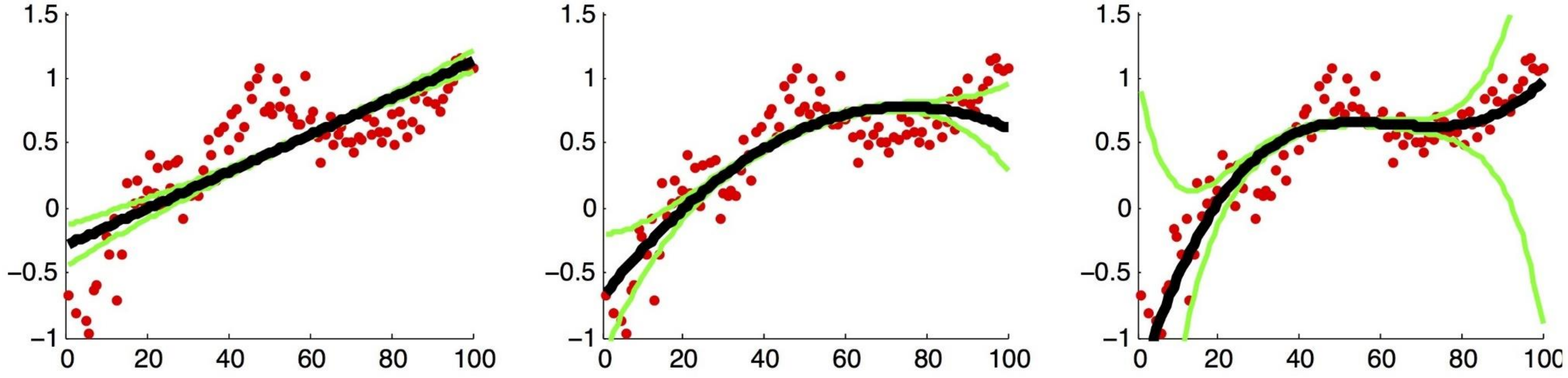
- **variance**: tendency to learn random things irrespective to the input data



http://mlwiki.org/index.php/Overfitting

Question

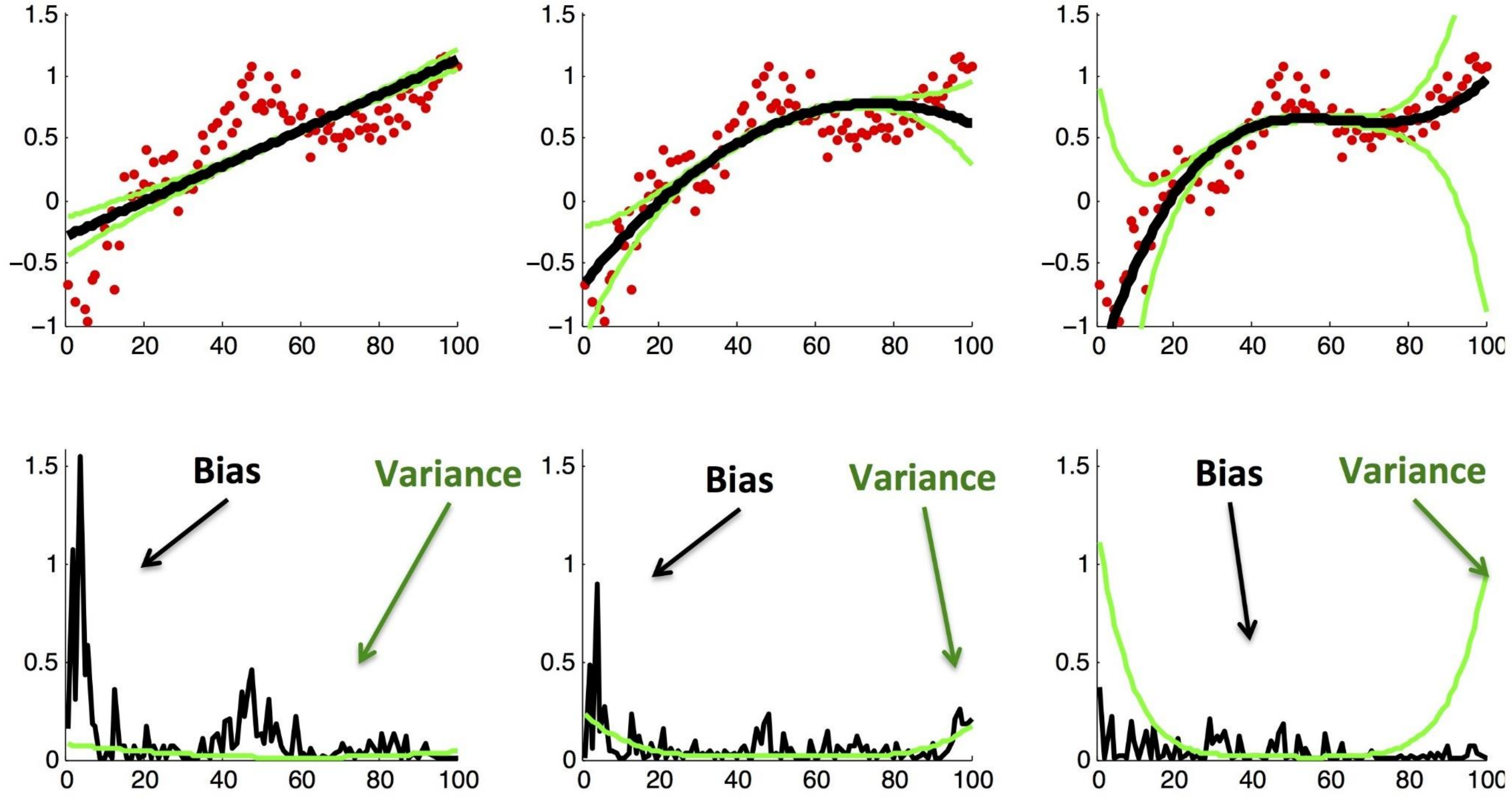
Three different models trained with a same small set of randomly chosen data points



Which one is underfit or overfit?

https://www.quora.com/What-is-high-bias-and-high-variance-in-machine-learning-terminology-in-simplest-terms

Answer

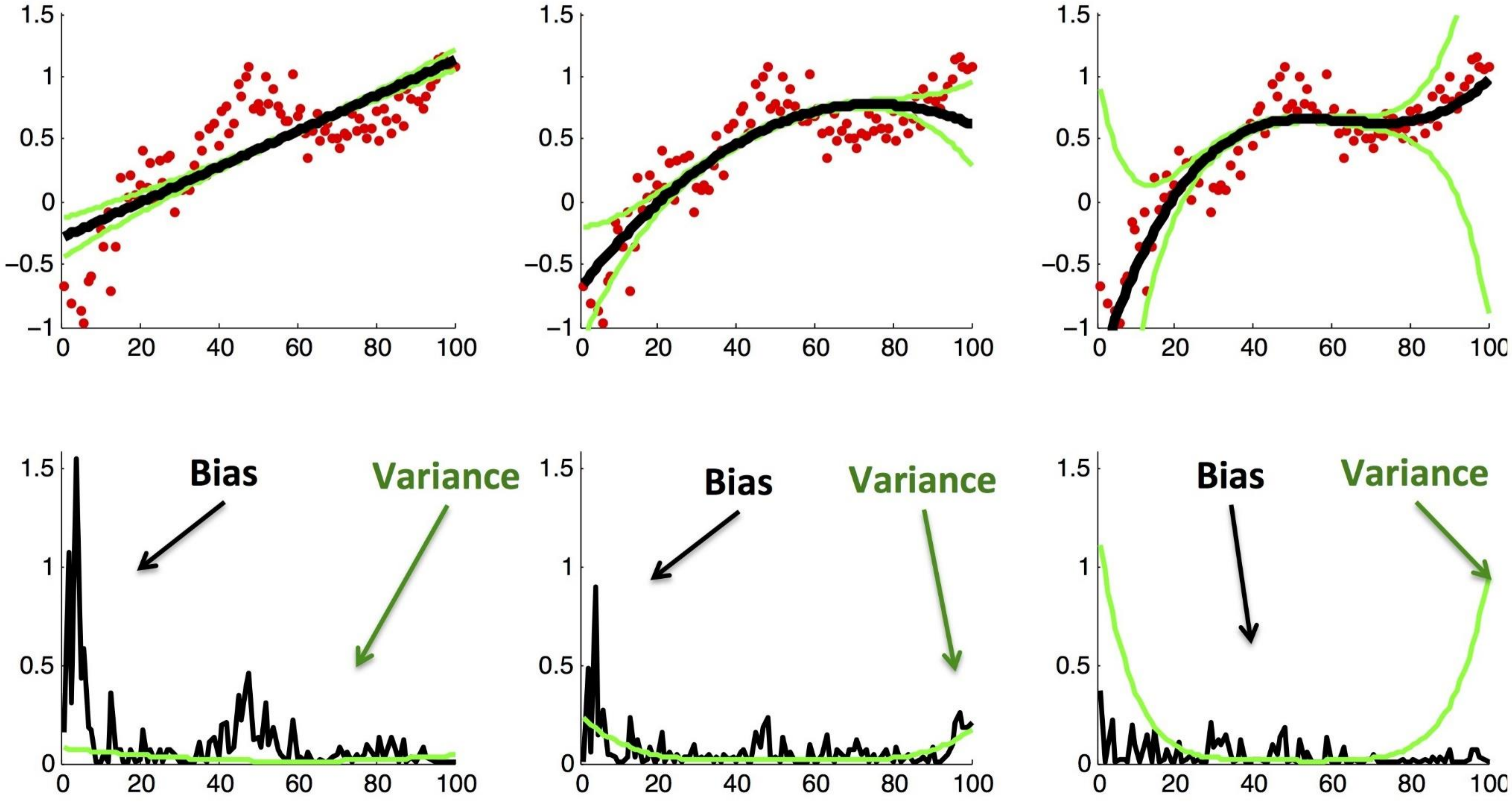


Quadratic model: the best trade-off of bias and variance

Linear model
High bias, low variance ➔ underfit

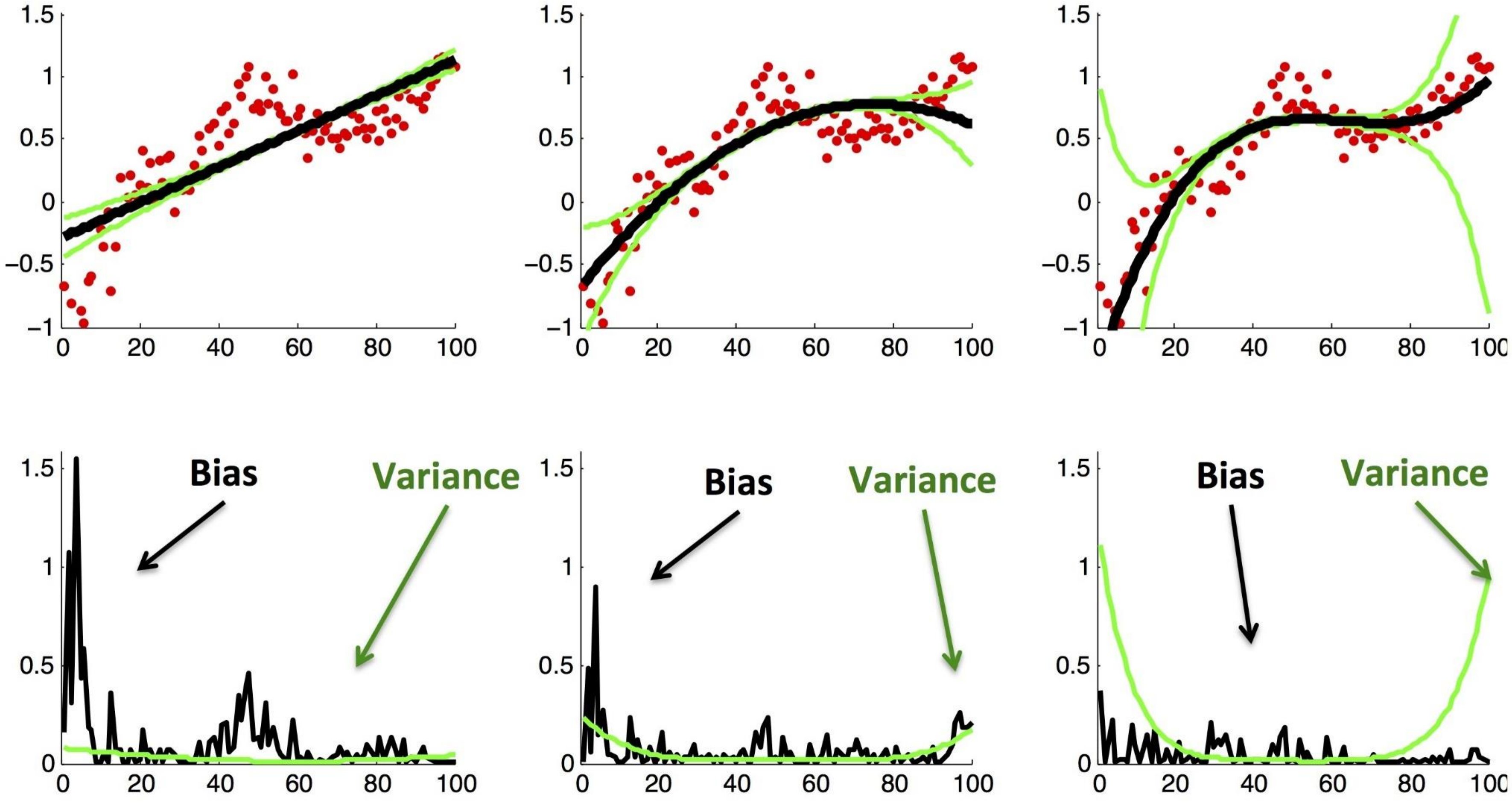
Cubic model
low bias, high variance ➔ overfit

Bias-variance tradeoff



If you train on even fewer data points, then the bias-variance tradeoff will shift in favor of linear models, because the variance term will dominate.

Bias-variance tradeoff



If you train on more, then the tradeoff will shift in favor of cubic models, because the bias term will dominate.

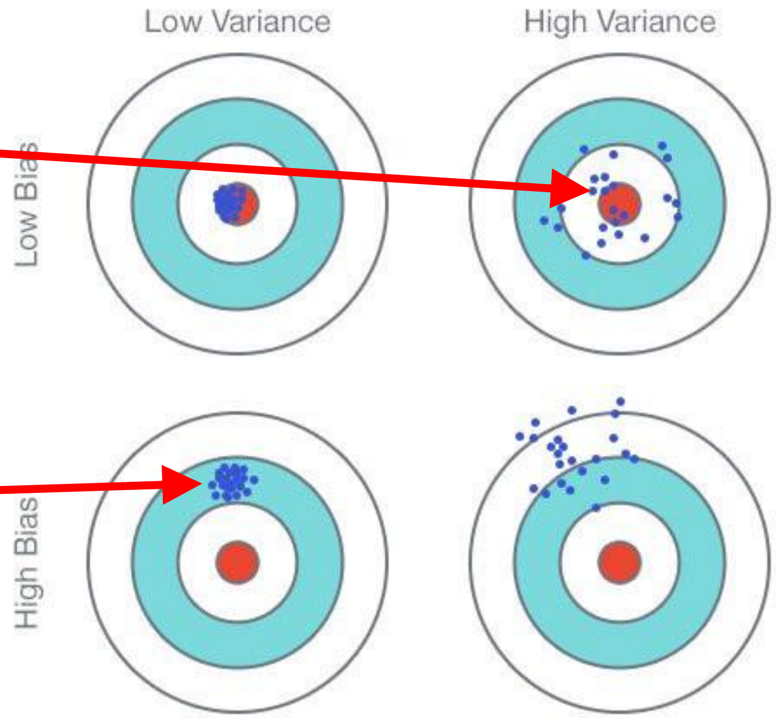
Variance & Bias

Overfitting

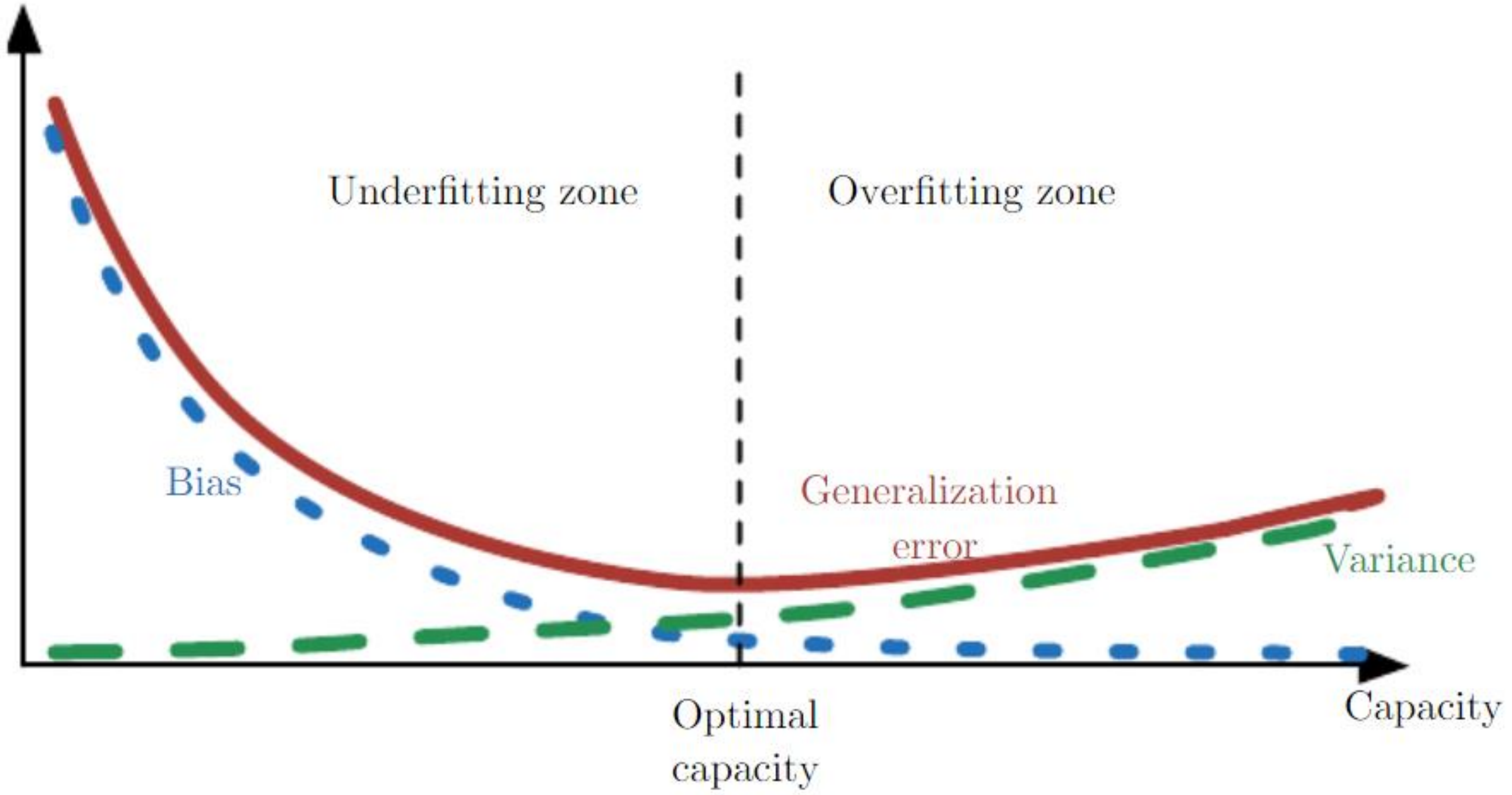
- low bias, high variance

Underfitting

- high bias, low variance



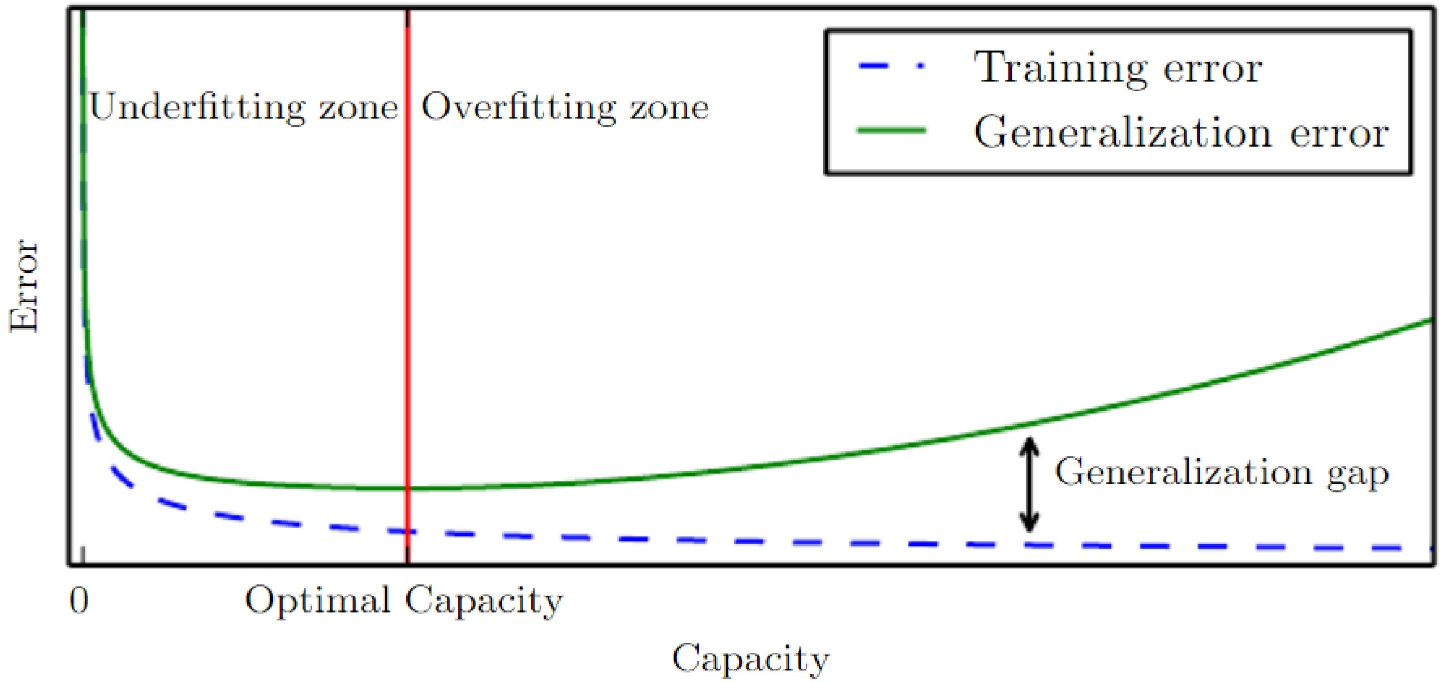
Variance & Bias



Model capacity

Model capacity

- Control an expected error by altering its **capacity** (= # parameters)
- Insufficient capacity; unable to solve complex tasks ➔ underfit
- High capacity; higher than needed to solve the present task ➔ overfit
- ML algorithms will generally perform best when their capacity is appropriate for the true complexity of the task with a given amount of training data.



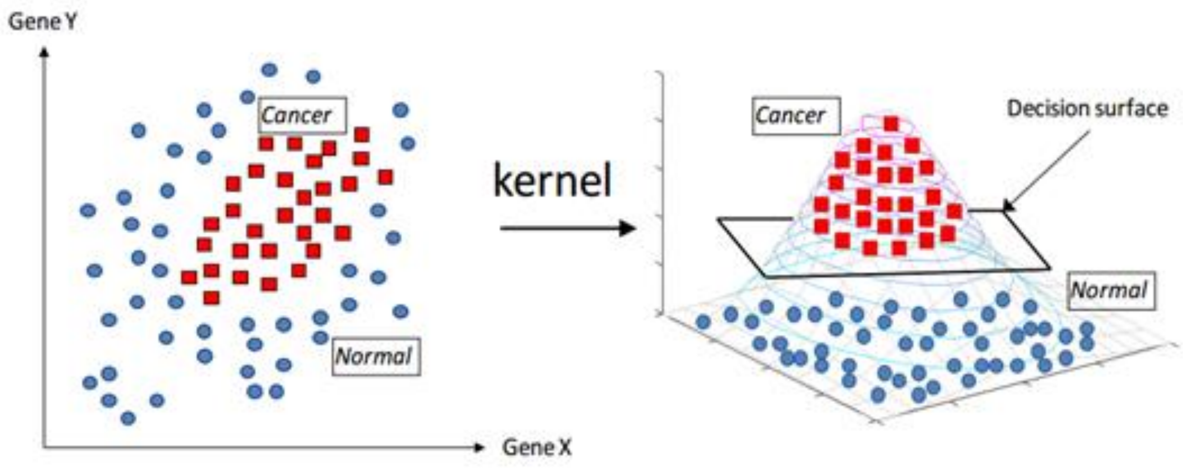
Representational capacity

- Capacity is not determined only by the number of parameters.

- A choice of model specifies which family of functions the learning algorithm can choose.

  ➔ **representational capacity** of the model

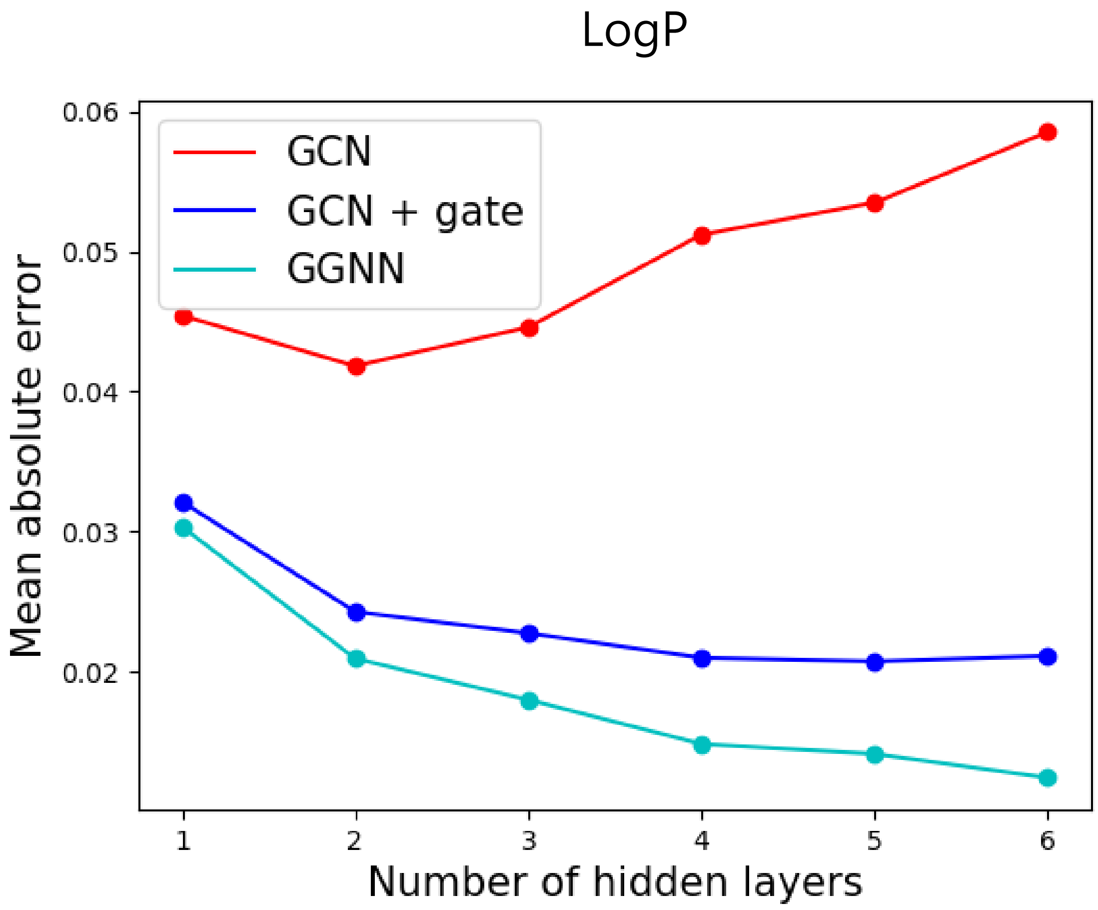
- For example, linear ➔ nonlinear such as SVM and DNN



Model selection

1. Model with a representational capacity in the hypothesis space
   ex) DNN, CNN, RNN, etc

2. Model with optimal complexity (i.e., # parameters)
   ex) # nodes/layer, # layers

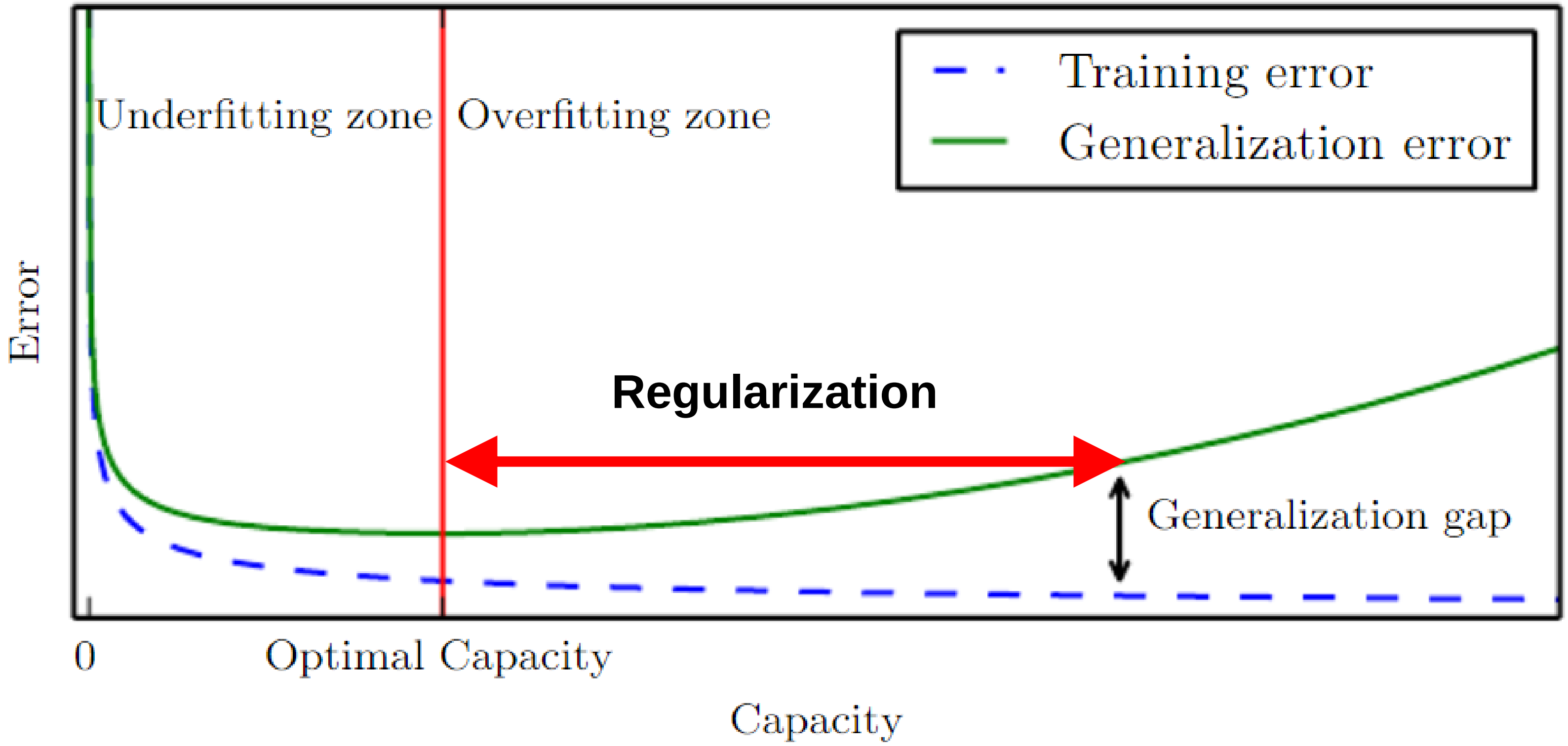


KAIST

Regularization

Regularization

- The main challenge is to find an optimal capacity of model for a given task.
- **Regularization** is any modification we make to a learning algorithm that is intended to reduce its generalization error but not its training error.



Strategy

1. Data augmentation: Big data

2. Model selection vis cross validation

3. L1,L2-Regularization

4. Dropout

Big Data

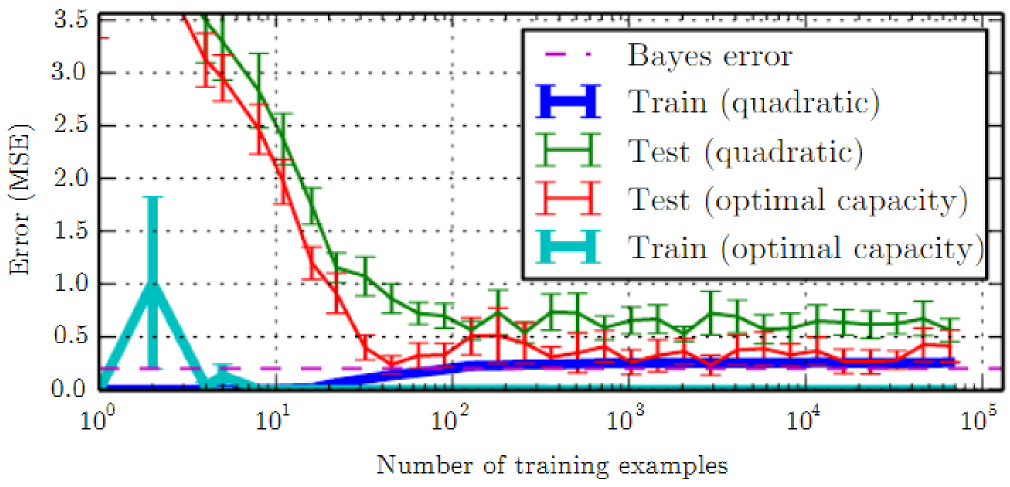
- Even an ideal model (let's say oracle) will incur some error on many problems

  because there may still be some noise in the data distribution.

- The error incurred by an oracle making predictions from the true distribution p(x, y) is called the **Bayes error**.

- Training and generalization error vary as the size of the training set varies.

- Expected generalization error can never increase as the number of training examples increases.

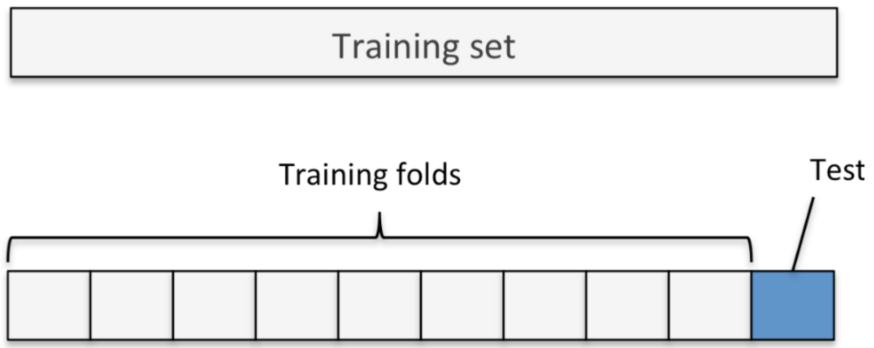


high variance ➔ use more data

Cross validation

Generally cross-validation is used to find the best value of parameters by reducing variability in the data

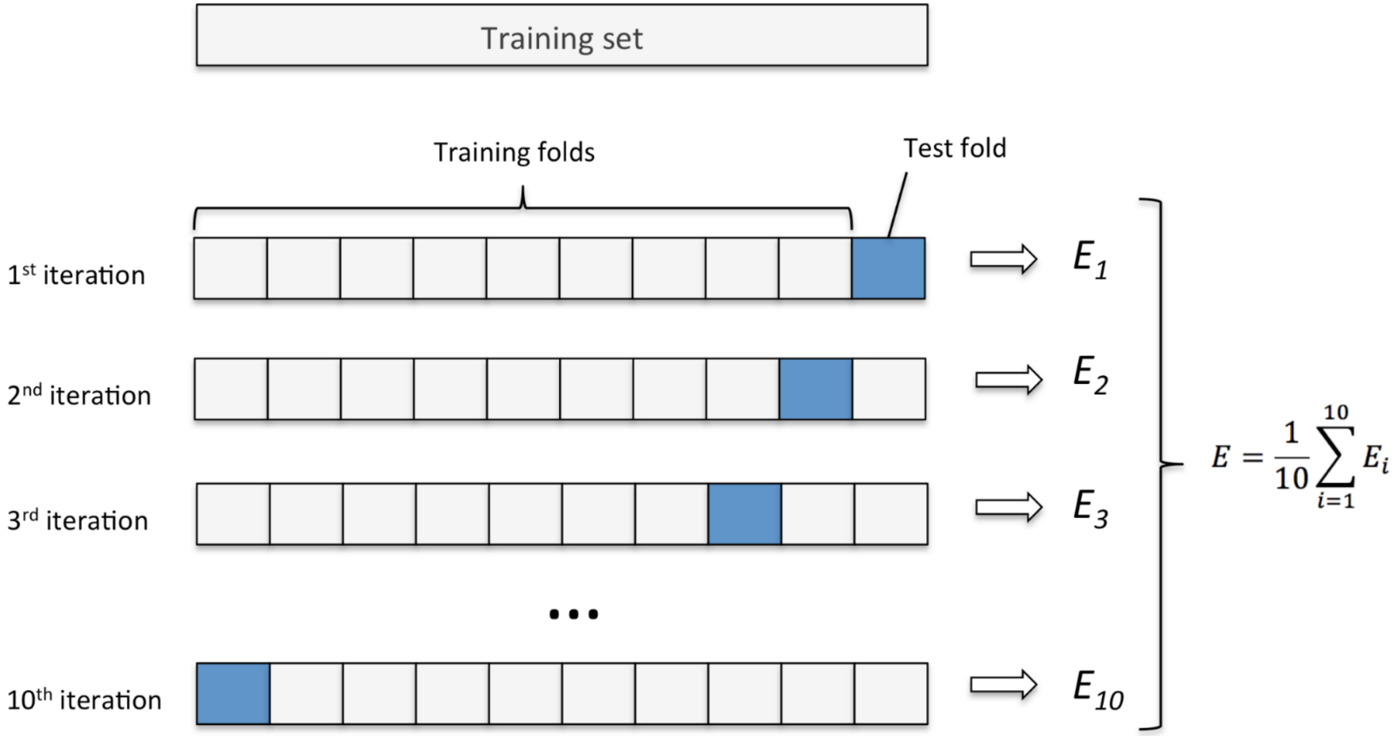
STEP 1: make a cross-validation set to test the performance of our model depending on the parameter



Cross validation

Generally cross-validation is used to find the best value of parameters by reducing variability in the data

STEP 2: perform multiple rounds of cross-validation using different partitions



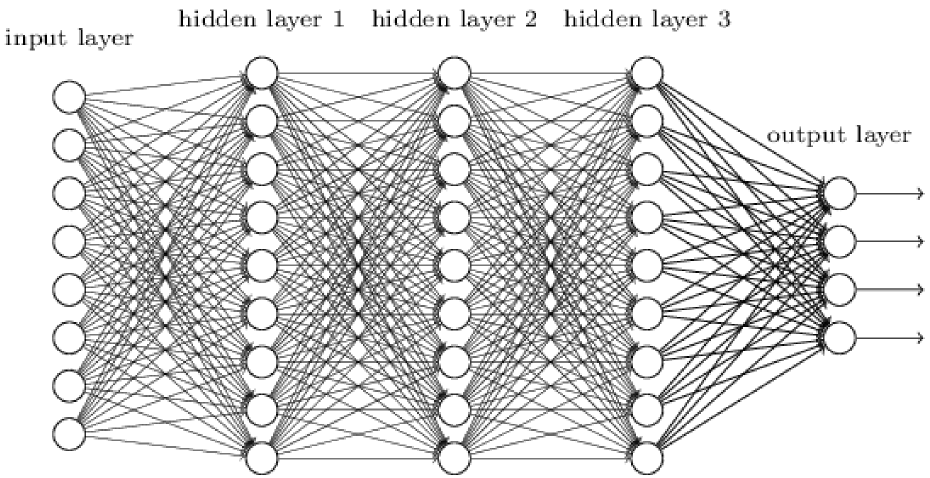
Because a certain model may show best performance with a specific data set

Then, average the results over all the rounds

Cross validation

Generally cross-validation is used to find the best value of parameters by reducing variability in the data

STEP 3: repeat the STEP 1 and 2 process with all model variations (hypothesis and # parameters)



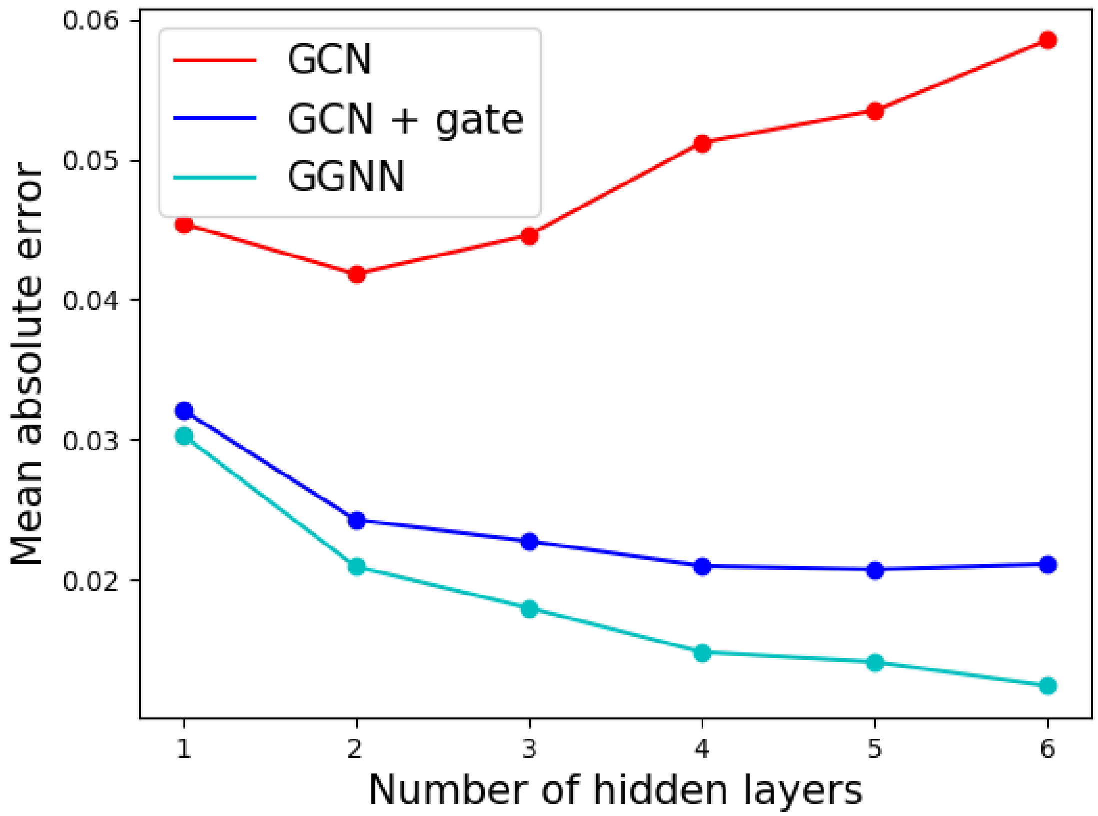
How much deep and wide?

DNN1 DNN2 DNN3 · · · DNNk

Cross validation

Generally cross-validation is used to find the best value of parameters by reducing variability in the data

STEP 4: choose the best one and apply it to the test set to obtain the final test error



KAIST

L2 Regularization

Excess reduction of training error may cause undesirable increase of weights to be biased by specific data points such as outlier

$$\mathbf{C} = \mathbf{C}_0 + \lambda\Omega \quad \Rightarrow \quad \mathrm{w} = \mathrm{w} - \alpha\frac{\partial C}{\partial w} = w - \alpha\frac{\partial C_0}{\partial w} - \alpha\lambda\frac{\partial \Omega}{\partial w}$$

SVM case

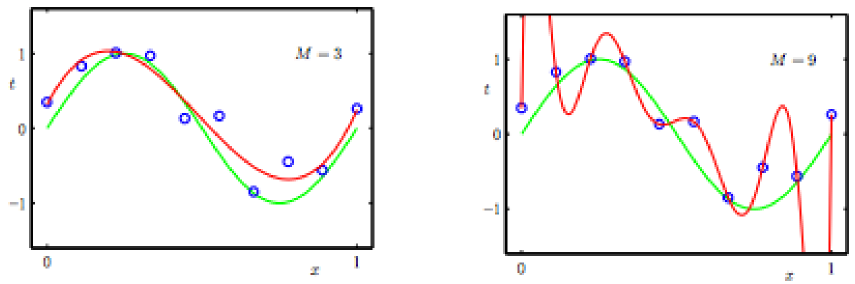
L2 (or ridge) regularization

$$\mathrm{C} = \mathrm{C}_0 + \lambda w^T w \quad \Rightarrow \quad \mathrm{w} = (1 - \alpha\lambda)w - \alpha\frac{\partial C_0}{\partial w}$$

Compared to the original case ($\lambda = 0$), large weights decrease by the first term as updated

➔ weight decay

M=3 M=9: *overfitting*



The weight decay prevents some weights enormously increasing which induce "overfitting".

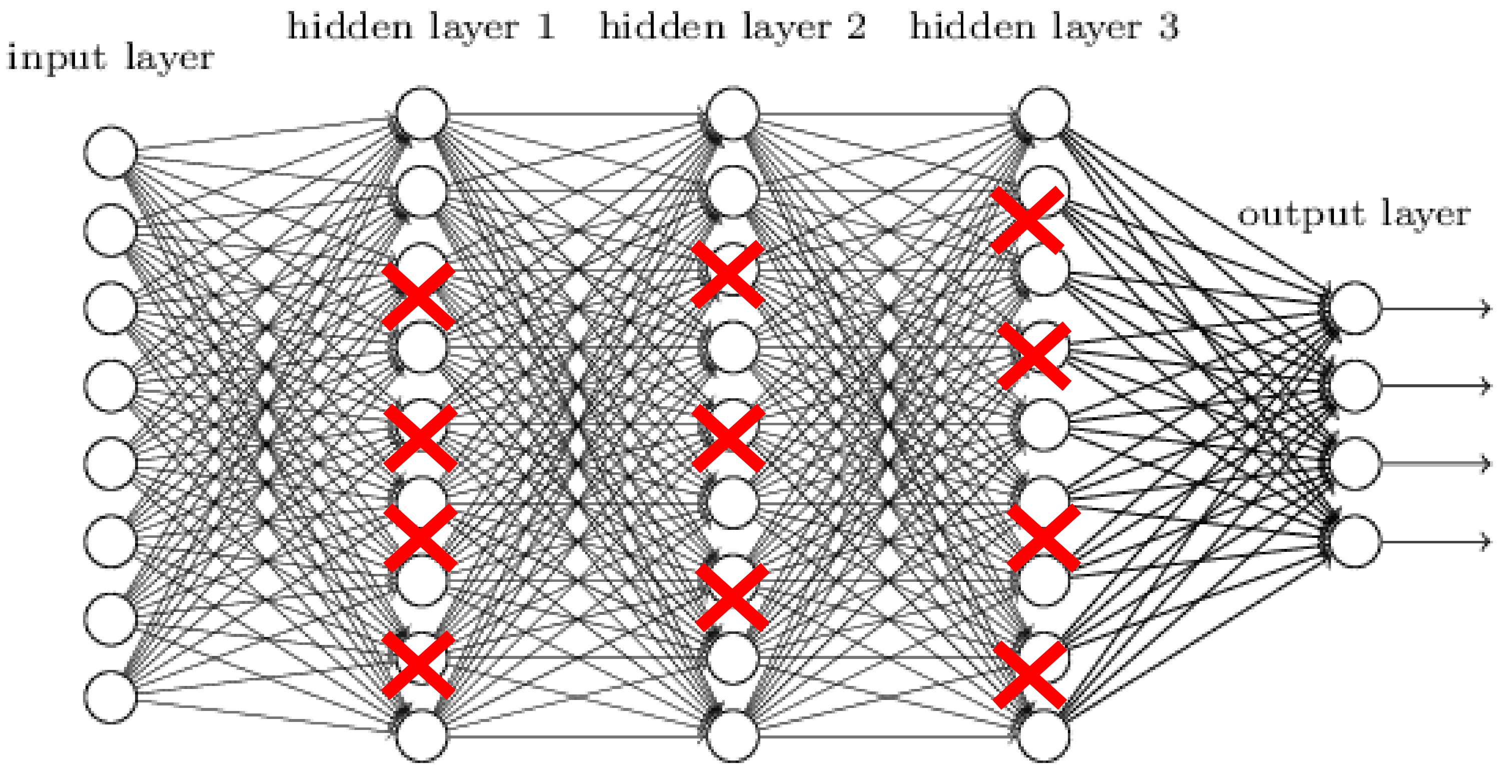
And it means that learning is not affected by "local noise" and "outlier"

KAIST

L2 Regularization

Compared to the original case ($\lambda = 0$), large weights decrease by the first term as updated

➔ **weight decay**

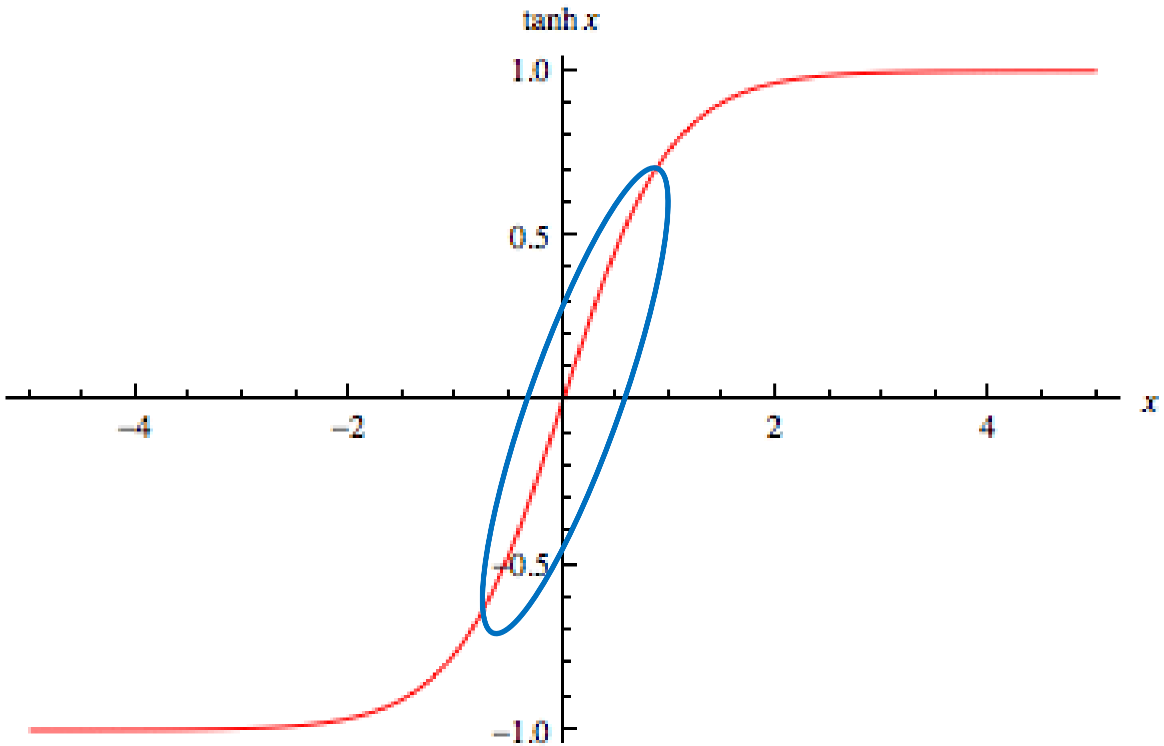


➔ much simpler DNN

Andrew Ng https://www.youtube.com/watch?v=NyG-7nRpsW8

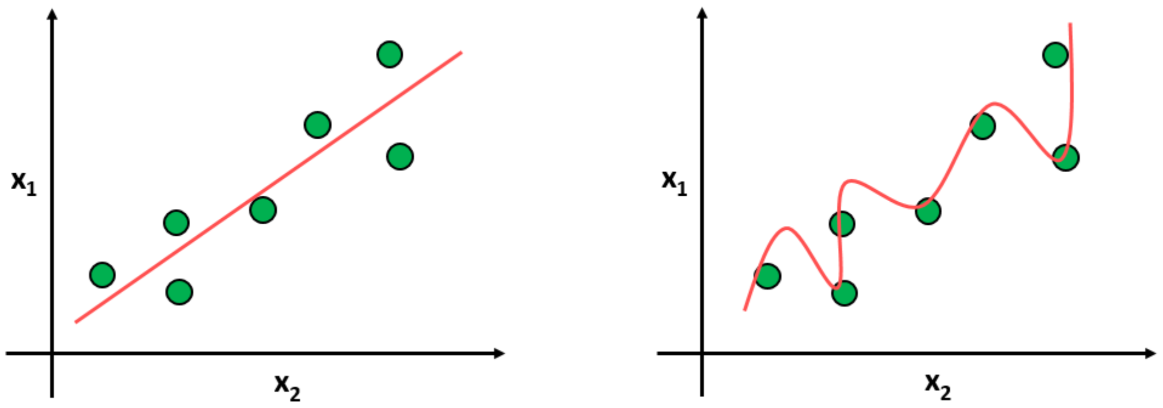
KAIST

L2 Regularization



$$z = f(\boldsymbol{w} \cdot \boldsymbol{x} + b) \rightarrow z \sim \boldsymbol{w} \cdot \boldsymbol{x} + b$$

Small w ➔ active function becomes linear➔ NN becomes roughly linear regression



Andrew Ng https://www.youtube.com/watch?v=NyG-7nRpsW8

L1 Regularization

Excess reduction of training error may cause undesirable increase of weights to be biased by specific data points such as outlier

$$\mathbf{C = C_0 + \lambda\Omega} \qquad \mathrm{w} = \mathrm{w} - \alpha\frac{\partial C}{\partial w} = w - \alpha\frac{\partial C_0}{\partial w} - \alpha\lambda\frac{\partial \Omega}{\partial w}$$

L1 (lasso) regularization

$$\mathrm{C} = \mathrm{C}_0 + \lambda\sum_i |w_\mathrm{i}| \quad \Longrightarrow \quad \mathrm{w} = w - \alpha\lambda \mathrm{sgn}(w) - \alpha\frac{\partial C_0}{\partial w}$$

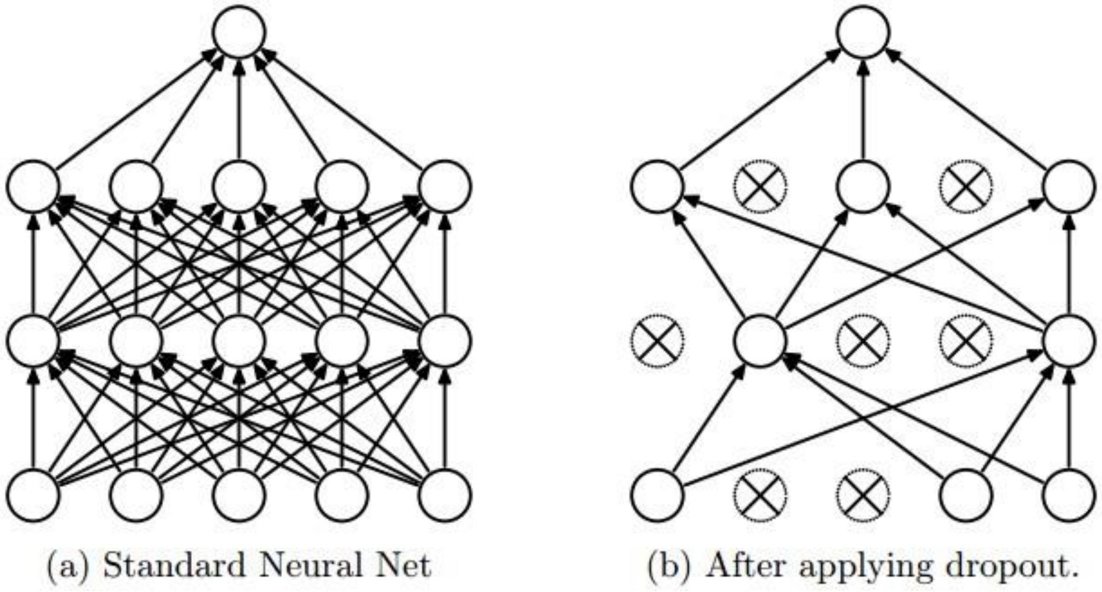
Compared to the original case ($\lambda = 0$), weights decrease depending on sign as updated
➔ **small weights go to zero** ➔ **important weights remain non-zero**

This helps **with** feature selection

Lasso: Least Absolute Shrinkage and Selection Operator

Dropout

Dropout randomly drops a subset of a layer’s perceptron’s activations to prevent from biasing specific data points.



사공이 많으면 배가 산으로 간다

A different set of activations is discarded across different iterations of learning.

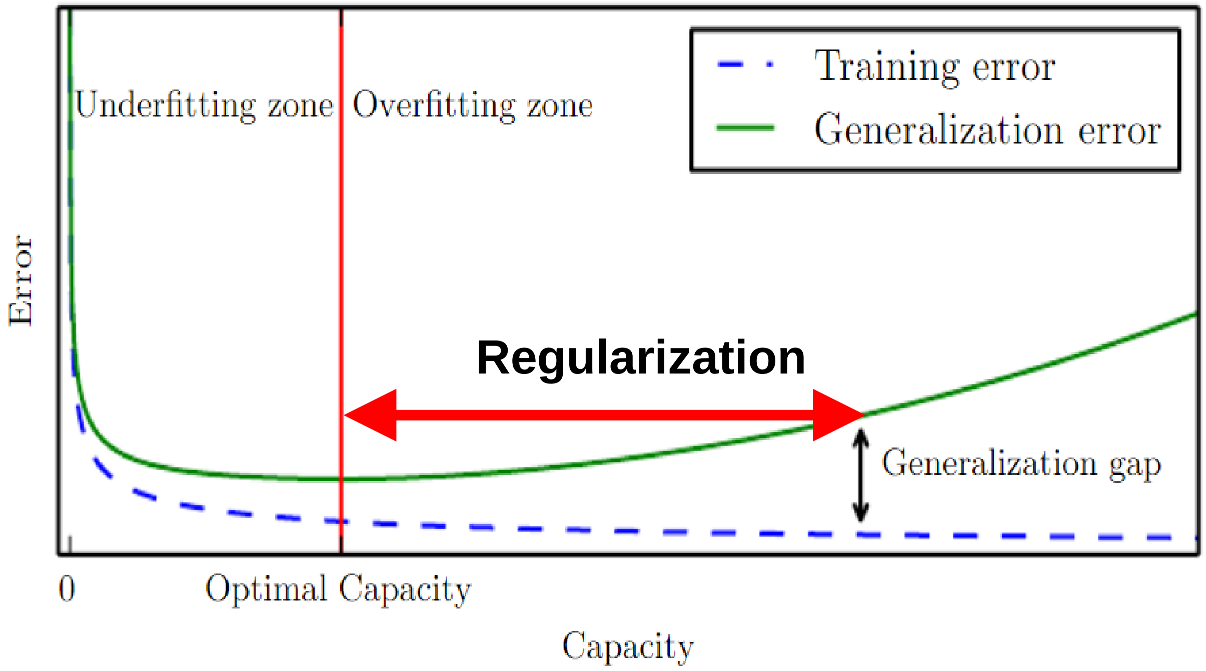
At last, connect all perceptrons

➔ taking average weights from many different neural network architectures.

KAIST

Summary

- The main challenge is to find an optimal capacity of model for a given task.
- **Regularization** is any modification we make to a learning algorithm that is intended to reduce its generalization error but not its training error.



1. Data augmentation: Big data

2. Model selection vis cross validation

3. L1,L2-Regularization

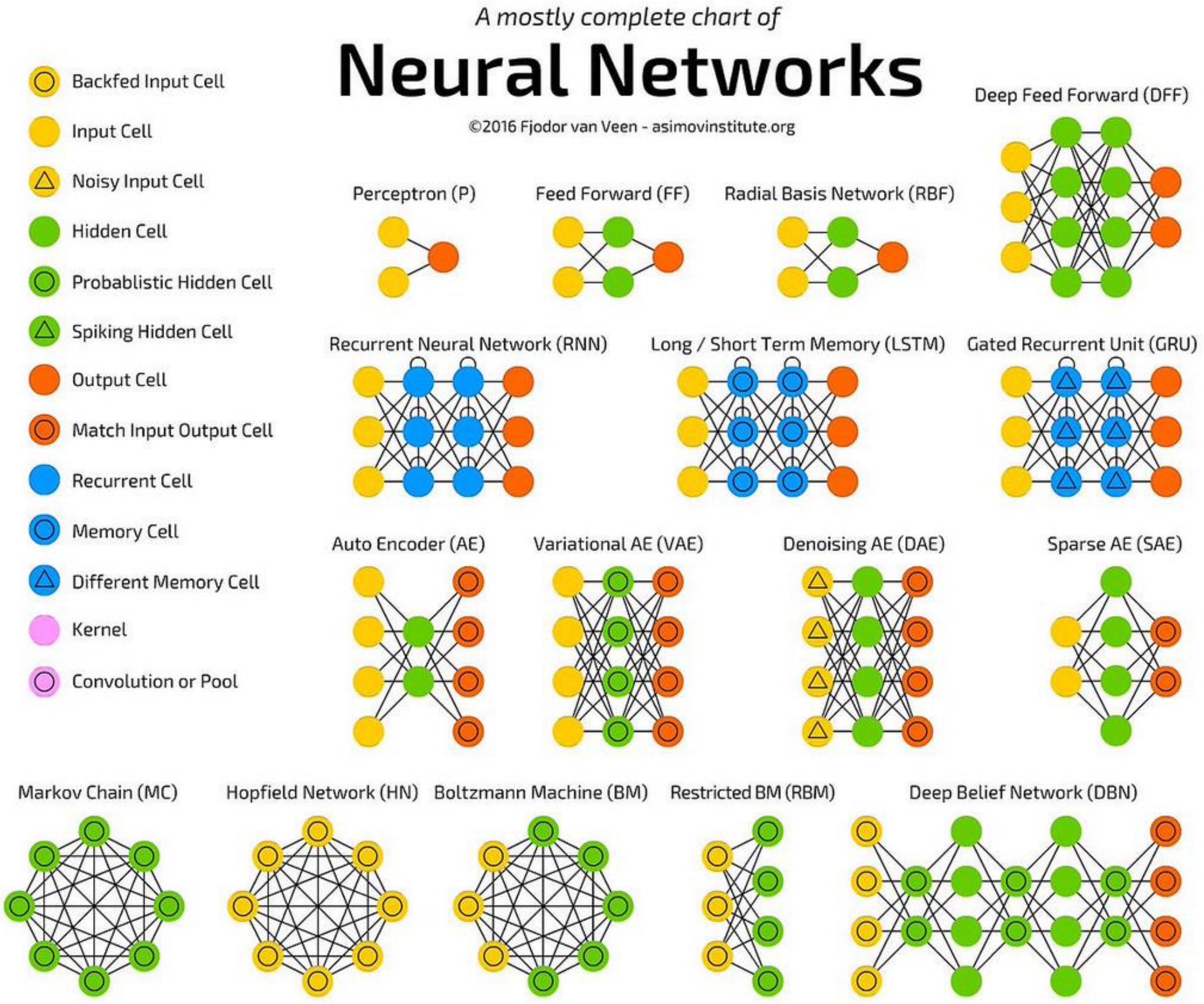
4. Dropout

KAIST

New terms

- Generalization
- Training error
- Test error
- Regularization
- Overfitting & underfitting
- Model capacity
- Optimal capacity
- Representational capacity
- Variance & bias
- Data augmentation
- Cross validation
- Validation set
- L1,L2 regularization
- Dropout

Choose a right model for a given problem to minimize generalization errors



http://www.asimovinstitute.org/neural-network-zoo/

KAIST

Course schedule

기간	분류	주제	학습활동	결과물
1주	주제	Introduction & math review		
	목표	Review of fundamental mathematics required to follow the course works.		
	내용	Data science, Linear algebra, Probability		
2주	주제	Machine learning fundamentals	Installation of anaconda, python3, numpy, scikit-learn, RDkit, tensorflow, exercising linear regression	Assign #1: linear regression
	목표	Understanding the basic principle of machine learning such as cost function and gradient descent.		
	내용	Linear regression, logistic classification		
3주	주제	Support vector machine (SVM) & summary	Exercising SVM for classification problem	Assign #2: regression using SVM
	목표	Understanding a key idea of SVM		
	내용	SVM, Regression and classification		
4주	주제	Deep learning & multilayer perceptron (MLP)	Applying MLP for classification problem and comparison between ReLU and sigmoid functions	
	목표	Understanding the perceptron concept and a basic principle of deep learning		
	내용	Universal approximation theorem backpropagation, vanishing gradient, activation function, ReLU		

KAIST

Course schedule

기간	분류	주제	학습활동	결과물
5주	주제	Multilayer perceptron 2	Exercising MLP for supervised learning	Assign #3: supervised learning with MLP and comparison with SVM
	목표	Knowing various issues on MLP and techniques to resolve them		
	내용	Overfitting, regularization, dropout, batch normalization, cross validation		
6주	주제	Convolutional Neural Network (CNN) & SMILES	Exercising CNN with SMILES for supervised learning of Log P and TPSA Ref. (1)	Assign #4: supervised learning of various molecular properties with CNN
	목표	Understanding CNN and molecular representation with SMILES		
	내용	Convolution, receptive field, stride, pooling Supervised learning of Log P and TPSA		
7주	주제	Molecular graphs & Graph Neural Network (GNN)	Exercising GCN with molecular graphs for supervised learning of Log P and TPSA Ref. (2), (3), (4)	Assign #5: improvement of vanilla GCN early-feedback(CELT)

KAIST

Course schedule

기간	분류	주제	학습활동	결과물
9주	주제	Recurrent neural network (RNN)	Exercising RNN with SMILES for supervised learning of Log P and TPSA Ref. (5)	Assign #6: supervised learning with RNN and comparison to GCN and SVM
	목표	Understanding RNN and molecular representations with SMILES		
	내용	RNN, LSTM, GRU, Feature extraction of molecules using RNN		
10주	주제	Message Passing Neural Network (MPNN)	Exercising GGNN with molecular graphs for supervised learning of Log P and TPSA Ref. (4), (6)	Assign #7: supervised learning with GGNN and comparison to GCN, GAT, RNN
	목표	Understanding the most general expression of graph neural network		
	내용	MPNN, molecular graph representation, GGNN, supervised learning of logP and TPSA		
11주	주제	Molecular generative model 1	Exercising VAE and CVAE for molecular design Ref. (7)	Assign #8: Optimization of molecular properties on latent space
	목표	Understanding the principle of autoencoder and unsupervised learning		
	내용	Molecular autoencoder, VAE, CVAE, de novo molecular design		
12주	주제	Molecular generative model 2	Molecular design from continuous latent space Ref. (8), (9)	Assign #9: comparison to the result of assign #8
	목표	Understanding difference between GAN and VAE		
	내용	GAN, ARAE ARAE: conditional molecular design		

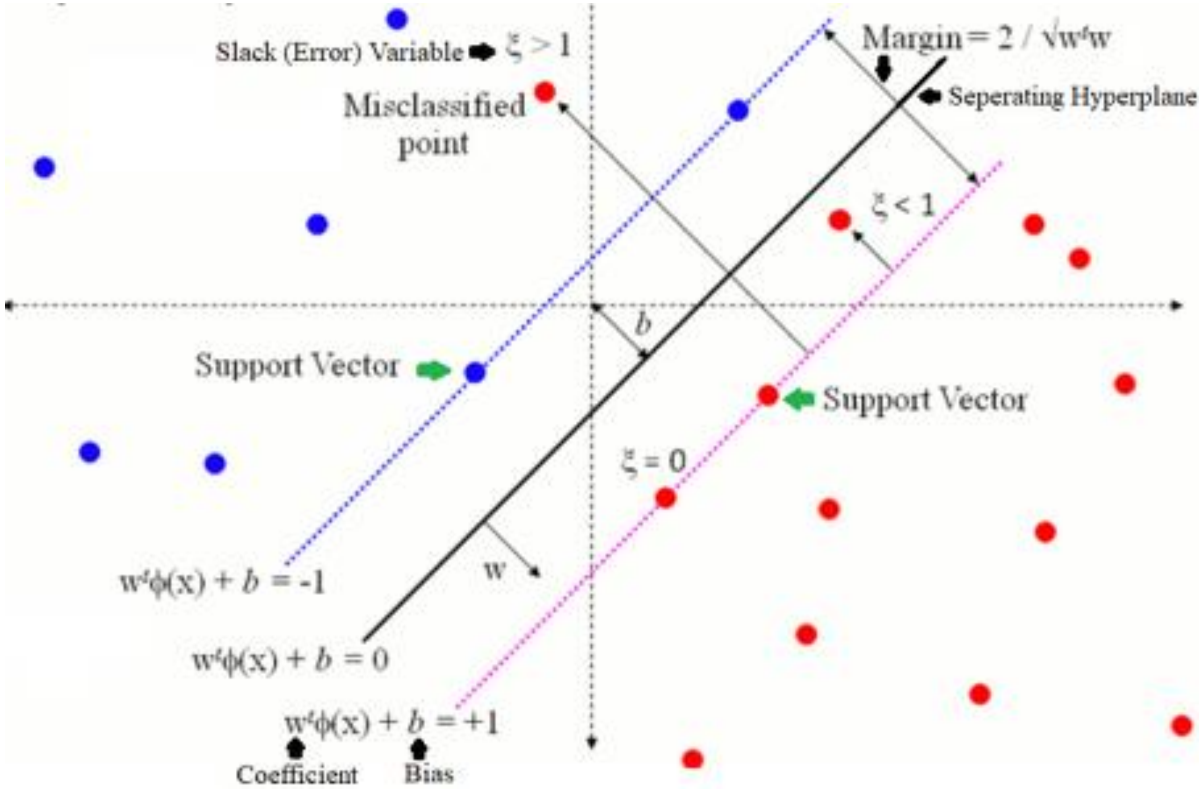
Course schedule

기간	분류	주제	학습활동	결과물
13주	주제	Molecular generative model 3	Molecular design with graph gene rative models Ref. (10), (11), (12)	Assign #10: scaf fold-based mole cular design
	목표	Understanding and graph structure based generative models		
	내용	Graph generative model, MolGAN, JTVAE		
14주	주제	No lecture (entrance interview)		
	목표			
	내용			
15주	주제	Term project presentation	Student presentation for the result s of their own term project	final feedback(CE LT)
	목표			
	내용			
16주	주제	Final exam	Student presentation for the result s of their own term project	
13주	목표			
	내용			

KAIST

Regularization

$$\min \frac{1}{2}\|\vec{w}\|^2 \quad \text{s.t.} \quad y_{\boldsymbol{i}}(\vec{w}\cdot\vec{x}_{\boldsymbol{i}}+b)=1$$

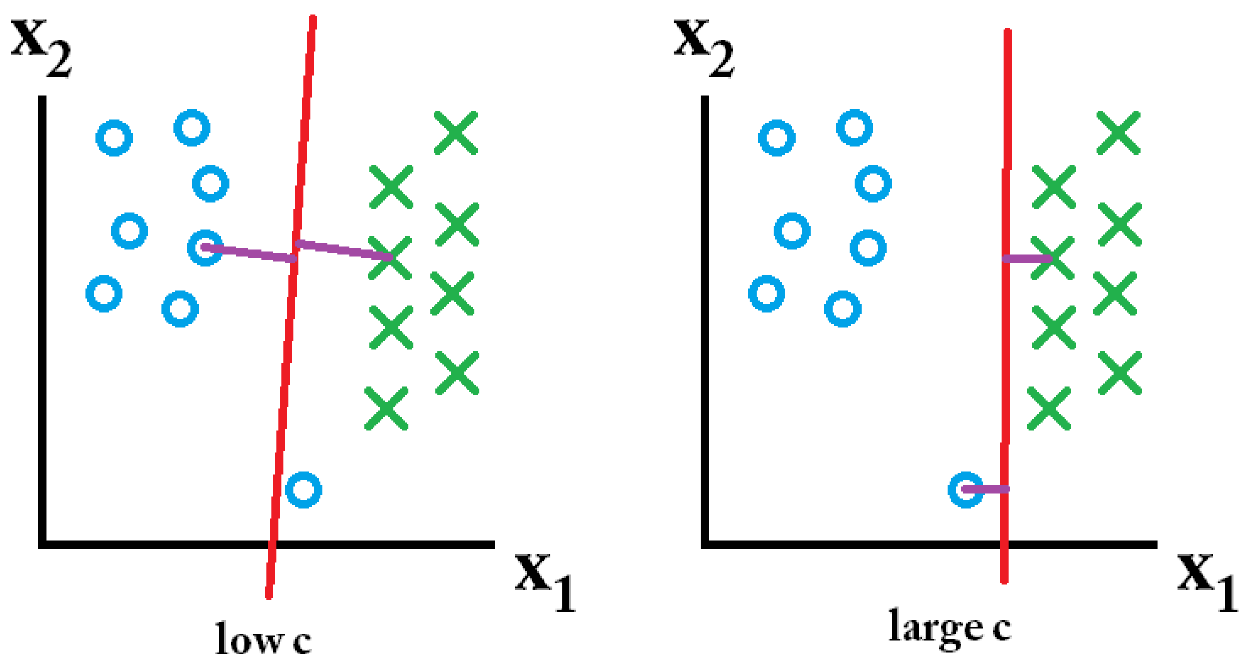


$$\min \frac{1}{2}\|\vec{w}\|^2 + C\sum_i \zeta_i \quad \text{s.t.} \quad y_{\boldsymbol{i}}(\vec{w}\cdot\vec{x}_{\boldsymbol{i}}+b)\geq 1-\zeta_i, \qquad \zeta_i \geq 0$$

to avoid overfitting

Regularization

The control parameter C



Extent of avoiding misclassifying data points

(i) small C

large ζ_i ➔ large margin with more misclassifying

(ii) large C

small ζ_i ➔ small margin with less misclassifying

$$\min \frac{1}{2}\|\vec{w}\|^2 + C\sum_i \zeta_i \qquad \text{s.t.} \quad y_{\boldsymbol{i}}(\vec{w}\cdot\vec{x}_{\boldsymbol{i}} + b) \geq 1 - \zeta_i, \qquad \zeta_i \geq 0$$