Large Scale Data Analysis Using Deep Learning

Machine Learning Basics - 1

U Kang
Seoul National University

In This Lecture

- Overview of Machine Learning
- Capacity, overfitting, and underfitting
- Evaluation: training set, validation set, and test set
  - Hyperparameters

Machine Learning (ML)

- Deep learning is a kind of machine learning

- Mitchell(1997): “A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks T, as measured by P, improves with experience E”

Task T

- ML tasks are usually described in terms of how the ML system should process an example
- Common tasks
  - Classification: produce f: $R^n \rightarrow \{1, \ldots, k\}$
  - Classification with missing inputs
  - Regression: produce f: $R^n \rightarrow R$
  - Transcription: observe a relatively unstructured representation of some kind of data, and transcribe it into discrete, textual form
    - Optical character recognition (OCR): photograph -> character
    - Speech recognition

Task T

- Common tasks
  - Machine translation
  - Structured output: any task where the output is a vector with important relationships between the different elements
    - Includes transcription and translation
    - Parsing
    - Pixel-wise segmentation of images: assigns every pixel in an image to a specific category
    - Image captioning
  - Anomaly detection
    - Credit card fraud detection

Task T

- Common tasks
  - Synthesis and sampling: generate new examples that are similar to those in the training data
    - Speech synthesis: text -> audio waveform
  - Imputation of missing values
  - Denoising: given a corrupted example $\tilde{x}$, predict the clean example x (or, predict $p(x \mid \tilde{x})$)
  - Density estimation, or probability mass function estimation: learn a function $p_{model}: R^n \rightarrow R$ where $p_{model}(\boldsymbol{x})$ is a probability density or mass function
    - Missing value imputation: given $p(\boldsymbol{x})$, we can compute $p(\boldsymbol{x}_i|\boldsymbol{x}_{-i})$

Performance Measure P

- Classification
  - Accuracy (proportion of examples for which the model produces the correct output) = (TP+TN)/Total
  - Precision = TP/ (TP + FP)
  - Recall = TP/ (TP + FN)

		Predicted condition	
		Positive	Negative
True condition	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

Performance Measure P

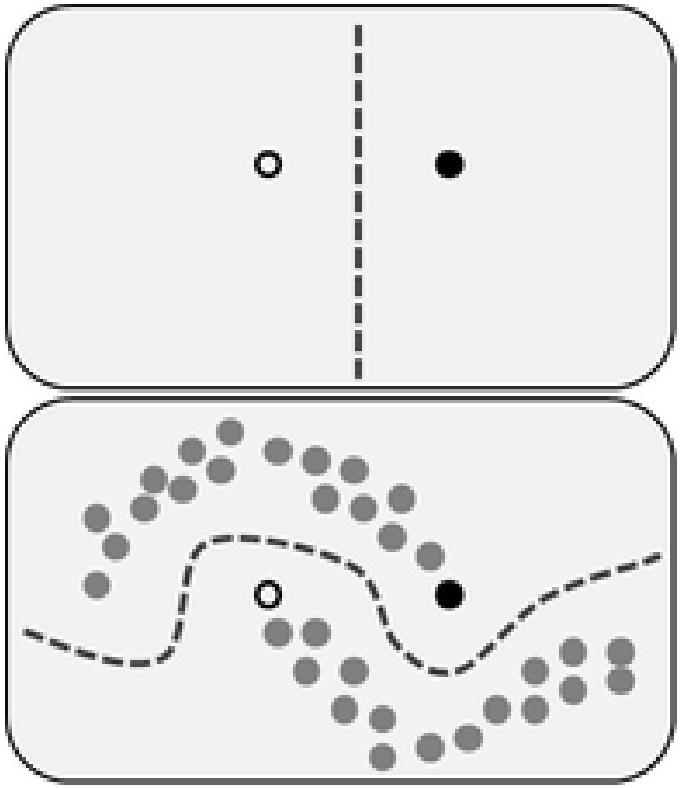
- Density estimation: requires a continuous-valued score for each example
  - Most popular: average log-probability the model assigns to examples (also called maximum likelihood estimator)
- Training set vs test set
  - Performance is measured on test set
  - Training error vs test error (generalization error)

Experience E

- ML algorithms are broadly categorized as unsupervised or supervised by what kind of experience they are allowed to have during the learning process with datasets
  - A dataset is a collection of many examples
- Unsupervised learning algorithms: learn useful properties of the structure of dataset
  - Given x, learn probability distribution p(x)
  - E.g., Clustering
- Supervised learning: each example is associated with a label or target
  - Given x and y, learn to predict y from x by estimating p(y|x)
  - E.g., Classification

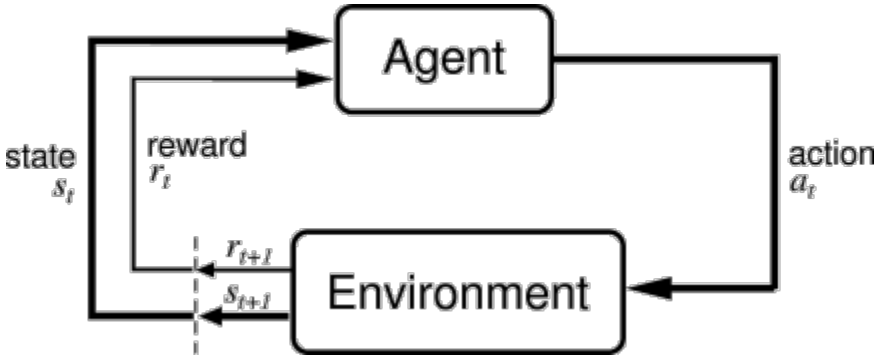
Experience E

- Semi-supervised learning: some examples include a supervision target but others do not



Experience E

- Reinforcement learning: interact with an environment
  - Feedback loop between the learning system and its experience
  - In a sense, the dataset changes



Design Matrix

- Matrix containing a different example in each row
- Iris dataset
  - 150 examples with 4 features for each example
  - Design matrix $X \in R^{150 \times 4}$

Linear Regression

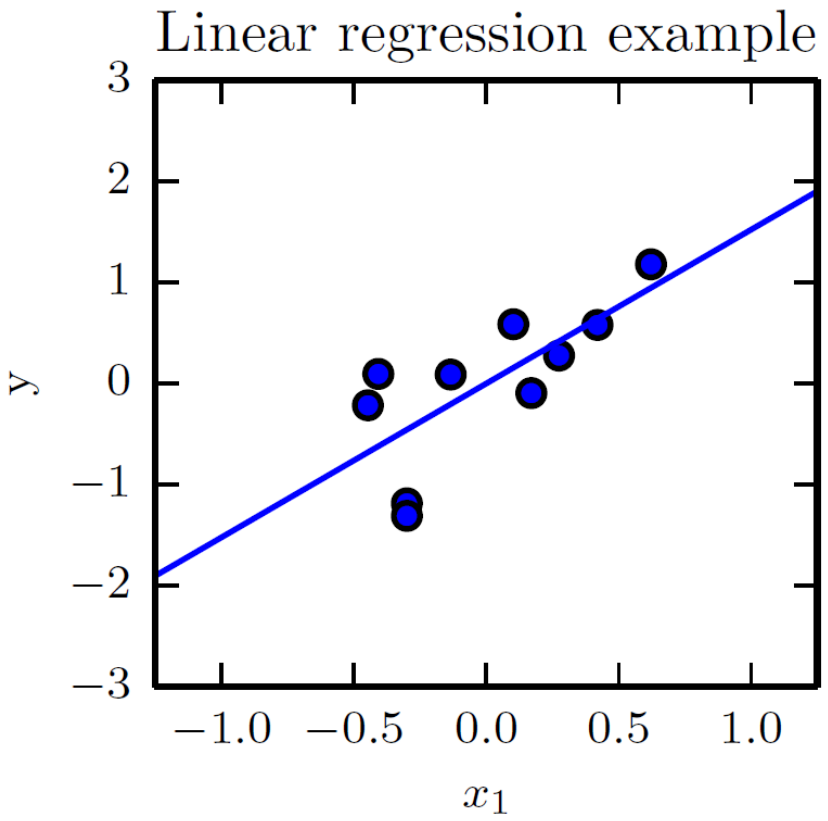
- Predict y from x by outputting $\hat{y} = w^T x$
- Goal: minimize mean squared error (MSE)
  - $MSE = \frac{1}{m} ||\hat{y} - y||_2^2$
- Minimizing MSE
  - $\nabla_w MSE = 0$
  - $\nabla_w ||Xw - y||_2^2 = 0$
  - $\nabla_w (Xw - y)^T (Xw - y) = 0$
  - $\nabla_w (w^T X^T Xw - 2w^T X^T y + y^T y) = 0$
  - $2X^T Xw - 2X^T y = 0$
  - $w = (X^T X)^{-1} X^T y$

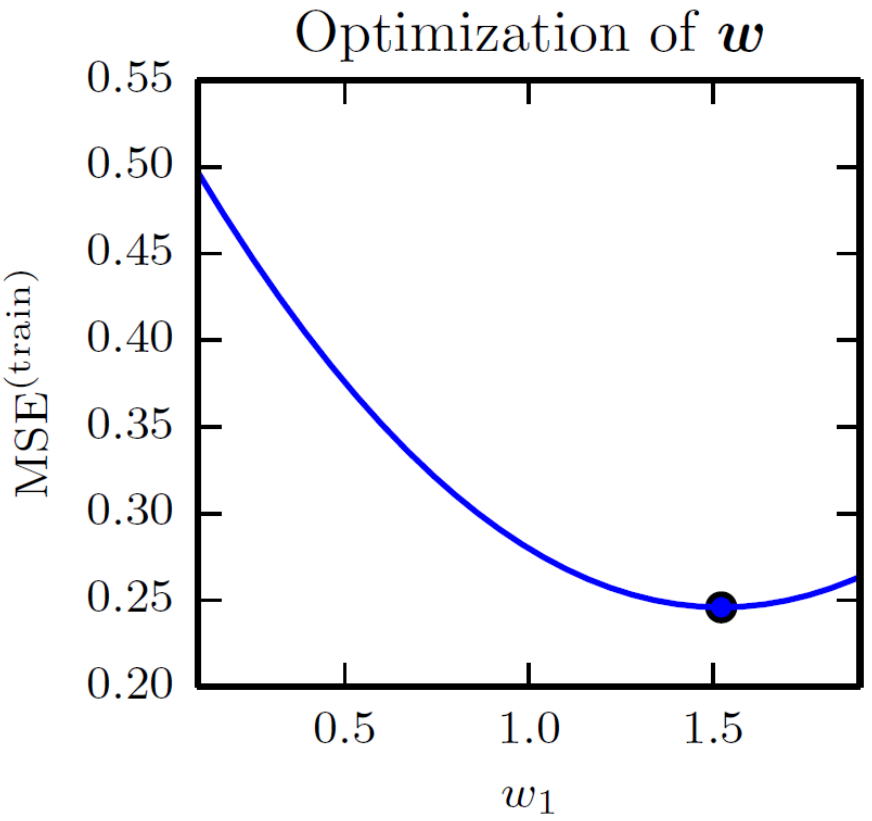
(*) Fact:
- $\nabla_w w^T Sw = 2Sw$ for symmetric S
- $\nabla_w w^T x = x$

(*)

(This is called 'normal equation')

Linear Regression





Capacity, Overfitting, and Underfitting

- Central challenge in ML: perform well on new, previously unseen inputs
- Minimize test error (generalization error)
- Data generating distribution p_{data} generates training and test data
- The factors determining ML algorithm's performance is its ability to
  - Make the training error small
  - Make the gap between training error and test error small

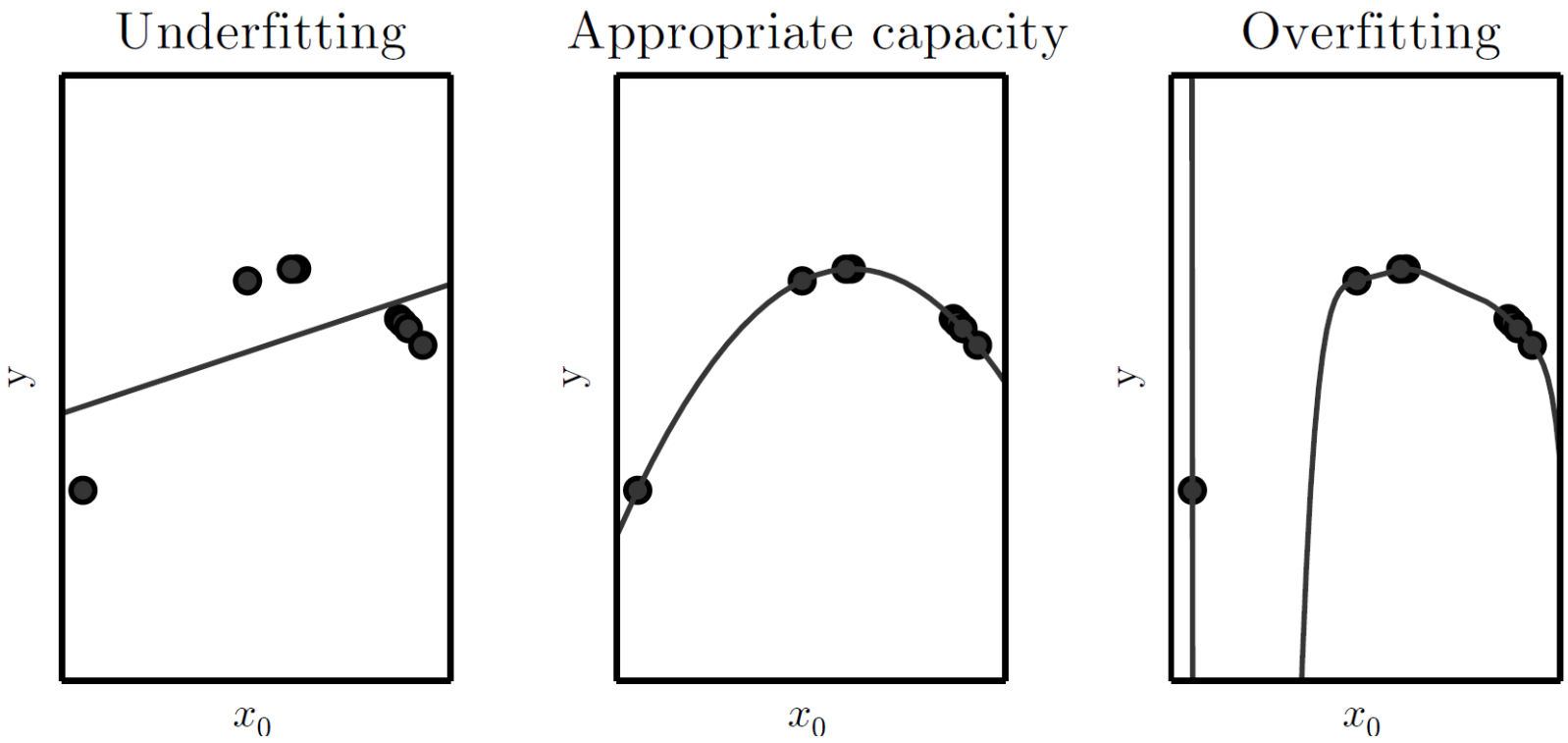
Capacity, Overfitting, and Underfitting

- Underfitting: a model is not able to obtain a sufficiently low error value on the training set
- Overfitting: the gap between the training error and test error is too large
- A model's capacity: its ability to fit a wide variety of functions
  - Models with low capacity: struggle to fit the training set (underfit)
  - Models with high capacity: overfit by memorizing properties of the training set that do not serve them well on the test set

Controlling Capacity

- Hypothesis space: the set of functions that a learning algorithm is allowed to select
- Choose hypothesis space of a model to control capacity
  - E.g., linear regression: $\hat{y} = b + wx$
  - By introducing x^2 as another feature, we learn a quadratic function $\hat{y} = b + w_1 x + w_2 x^2$
  - We can continue to add more powers of x, and still use normal equation to solve it
    - Called polynomial regression. Note that we can use the same normal equation as in the linear regression.

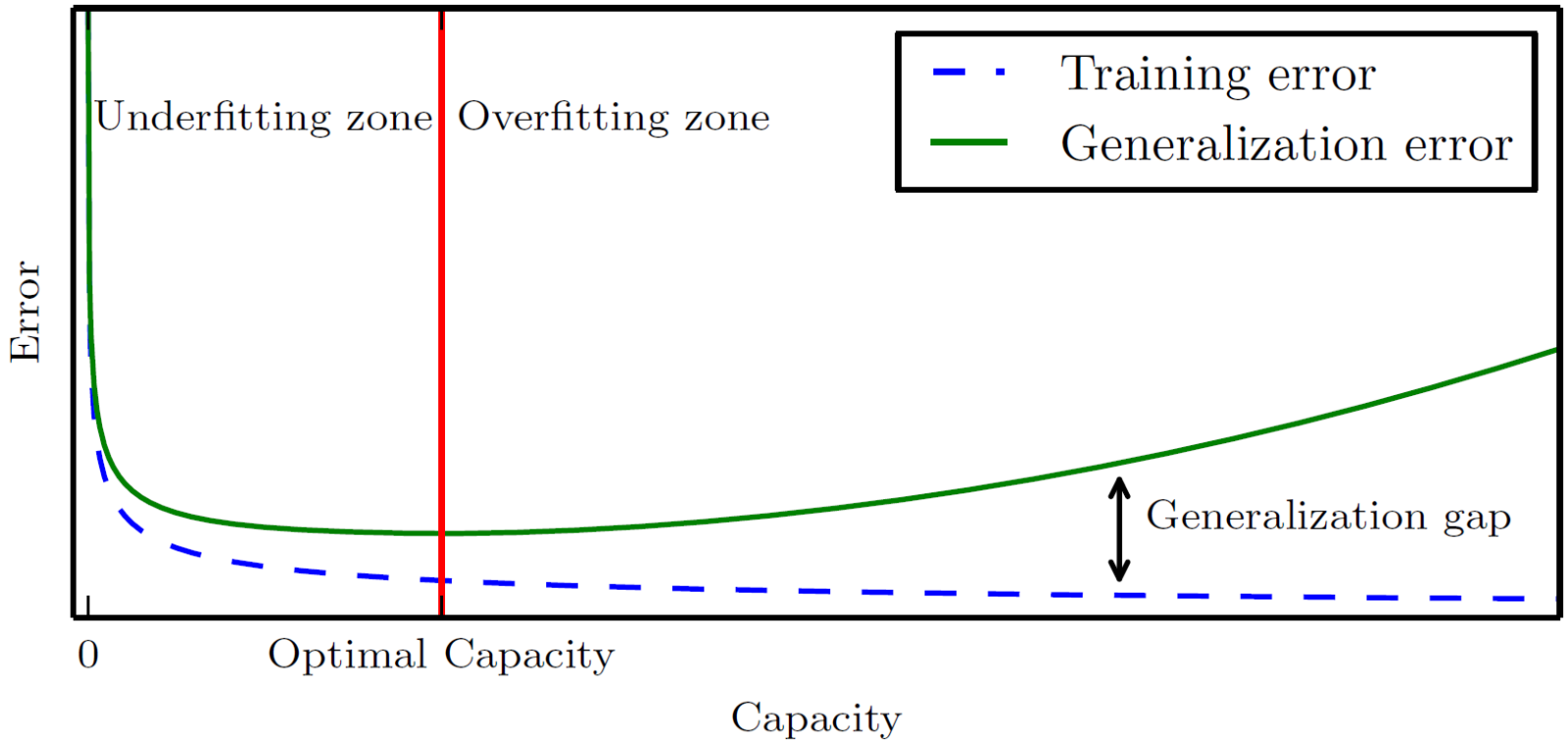
Underfitting and Overfitting in Polynomial Estimation



Difficulty in Determining Capacity

- Simpler functions are more likely to generalize well (to have a small gap between training and test error)
- However, we must still choose a sufficiently complex hypothesis to achieve low training error
- Typically, generalization error has a U-shaped curve as a function of model capacity

Generalization and Capacity



Difficulty in Determining Capacity

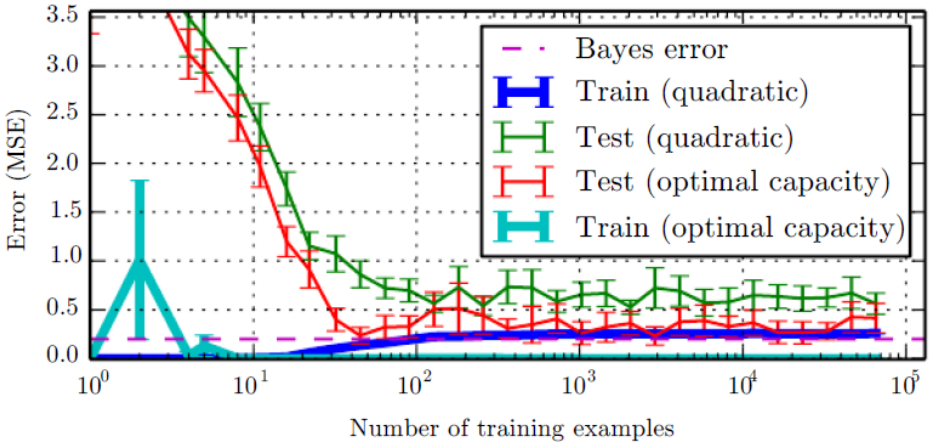
- Non-parametric model
  - Extreme case of arbitrary high capacity
  - Make its complexity as a function of the training set size
  - E.g., nearest neighbor regression
    - When asked to classify a test point $\boldsymbol{x}$, the model outputs $\hat{y} = y_i$, where $i = argmin\ ||X_{i,:} - \boldsymbol{x}||_2^2$
    - This algorithm is able to achieve the minimum possible training error

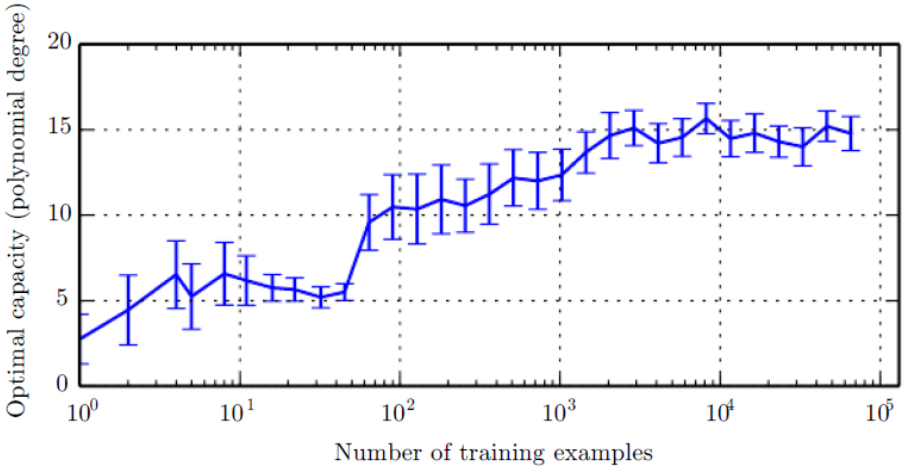
Difficulty in Determining Capacity

- Ideal model
  - An oracle that simply knows the true probability distribution that generates the data
  - Even such a model will still incur some error on many problems because there may still be some noise in the distribution
  - Bayes error: the error incurred by an oracle making predictions from the true distributions p(x,y)

Training Set Size

- Bayes error: constant
- Low capacity model
  - Test error (expected generalization error) decreases until the best possible error is achieved
- Optimal capacity model
  - The test error asymptotes to the Bayes error
  - Training error can fall below the Bayes error due to the ability of the training algorithm to memorize specific instances of the training data





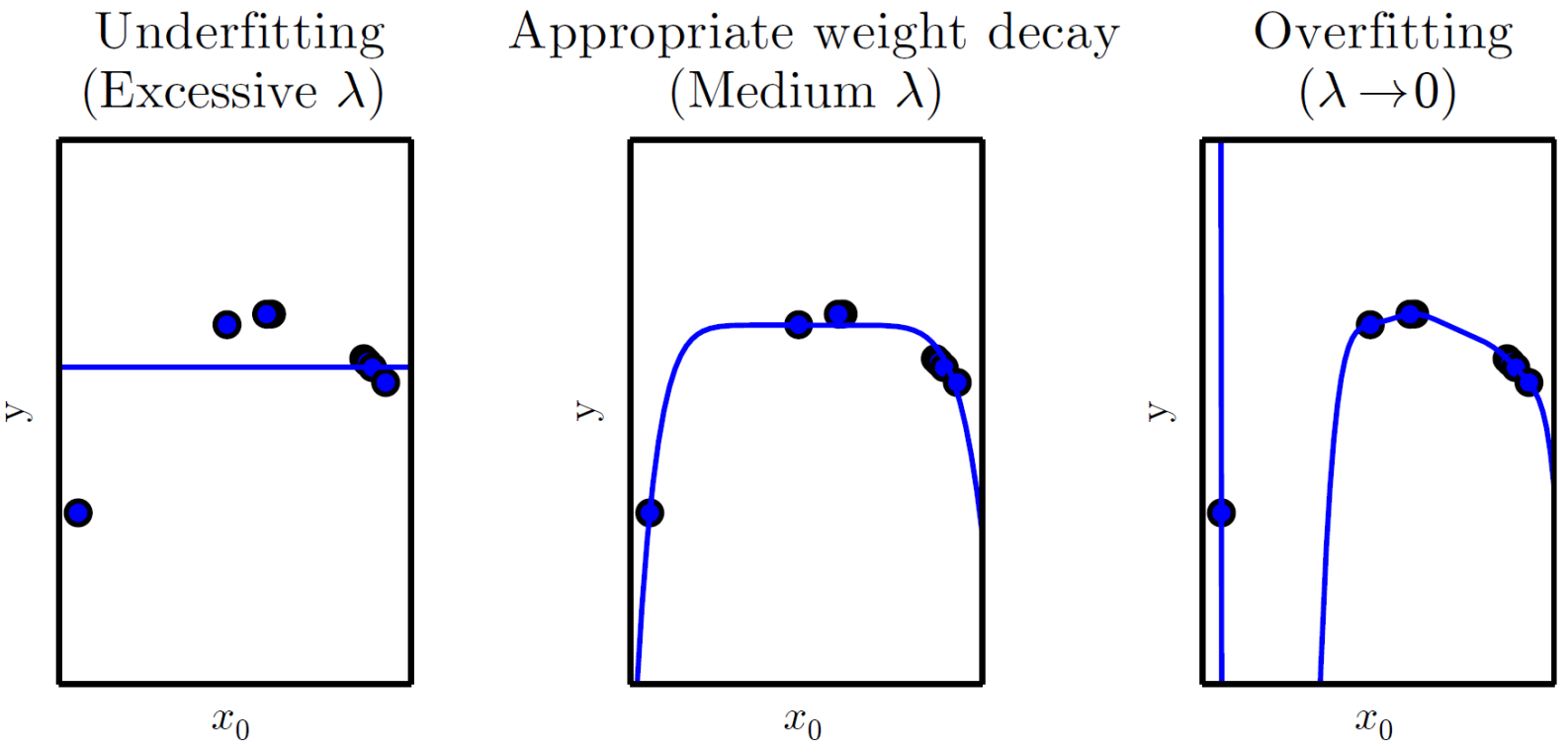
No Free Lunch Theorem

- Learning theory claims that an ML algorithm can generalize well from a finite training set of examples
- But in theory, it may not be true
  - No free lunch theorem (Wolpert, 1996): averaged over all possible data generating distributions, every classification algorithm has the same error rate when classifying previously unobserved points
  - No machine learning algorithm is universally any better than the other
- In reality, we do not average over all possible data generating distributions; we can design learning algorithms that perform well on some distributions

Regularization

- One way of designing (or changing capacity of) an ML algorithm is by adding or removing functions from the hypothesis space of solutions the learning algorithm is able to choose
- Another option is to express a preference for one solution to another
- Regularization: any modification we make to a learning algorithm that is intended to reduce its generalization error but not its training error
- In linear regression example, change the cost function so that $J(w) = MSE + \lambda w^T w$

Weight Decay



Hyperparameters and Validation Sets

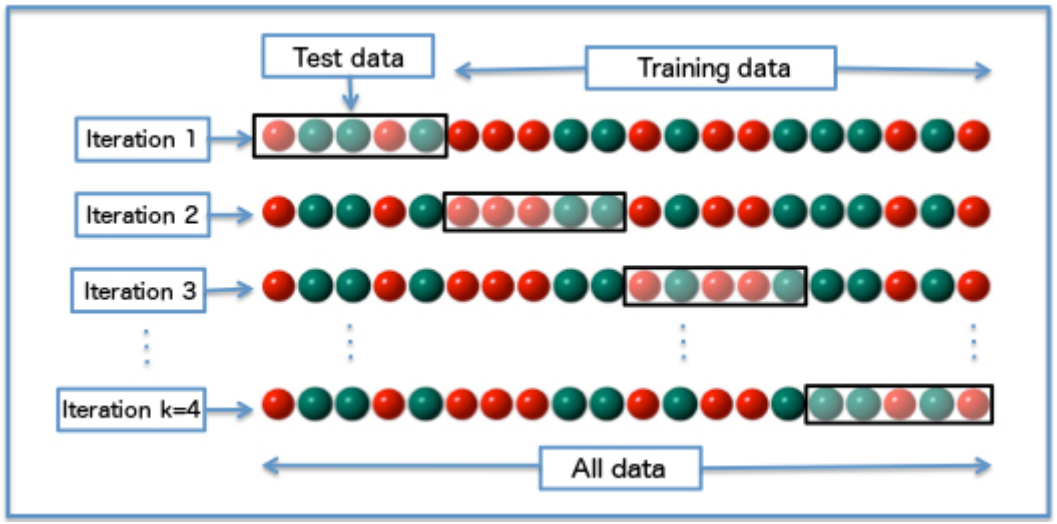
- Hyperparameters: settings to control the behavior of the learning algorithm
  - The values of hyperparameters are not adapted by the learning algorithm itself
  - E.g., in the polynomial regression, the degree of the polynomial acts as a capacity hyperparameter
  - E.g., in linear regression with regularization, λ is a hyperparameter used to control the strength of weight decay
- We do not learn hyperparameters on the training data
  - If so, we always choose hyperparameters that maximize possible model capacity, resulting in overfitting
    - E.g., we would choose $\lambda = 0$ in the linear regression case
  - Hyperparameters are selected from a *validation set*

Hyperparameters and Validation Sets

- Test error is evaluated on the test set
  - Since test examples should not be used in any way to make choices about the model, including its hyperparameters
- Typical evaluation scenario
  - Given a full dataset, divide it into training set R and test set T
  - Divide the training set R into (smaller) training set Rt and validation set Rv (typically, 80%: 20%)
  - Find the best parameters from Rt
  - Find the best hyperparameters from Rv
  - Evaluate the model in T

Cross Validation

- Dividing the dataset into a fixed training set and a fixed test set can be problematic if the test set size is small
- Cross-validation: use all of the examples for test, at the price of increased computation cost
  - K-fold cross validation: split the dataset into k disjoint subsets
  - The test error is the average of k trials
  - On i-th trial i, i-th subset of the data is used as the test set and the rest of the data is used as the training set



What you need to know

- Machine learning: a computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks T, as measured by P, improves with experience E
- Overfitting and underfitting are crucial for the performance of an ML algorithm
  - They can be avoided by controlling capacity of the model
  - To control capacity, extend hypothesis space or use regularization
- Evaluation in ML
  - Hyperparameters are learned on validation set
  - Cross validation is widely used

Questions?