CMPT 318
Data Analytics

Learning Algorithms

Adapted in part from
Chapter 11 of Data Science from Scratch (Grus)
Chapter 5 of Deep Learning (Goodfellow)

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 in *T*, as measured by *P*, improves with experience *E*.

(Goodfellow 2017)

Tasks

- Classification
- Regression
- NLP: Transcription, Translation
- Structured output
- Anomaly detection
- Synthesis and sampling
- Density estimation

(Goodfellow 2017)

Computer Vision Tasks

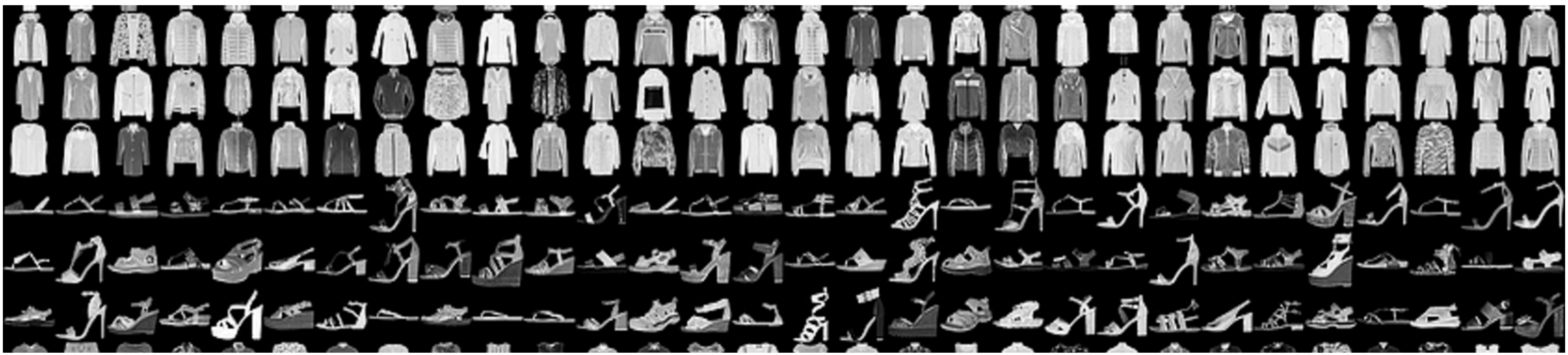
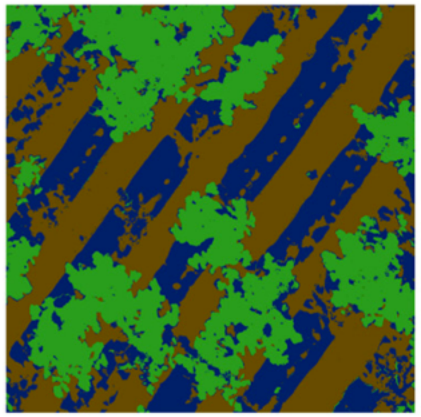


Image Classification

(Goodfellow 2017)

Computer Vision Tasks

Regression



Semantic Segmentation

Object Detection

Instance Segmentation

(Goodfellow 2017)

Performance

- Measure the “accuracy” of the model
- Easy to measure for some tasks, e.g. classification
- Hard to measure for others, e.g. density estimation
- May be intractable

(Goodfellow 2017)

Experience

- aka, the *Data* (x) & *Labels* (y)
- Unsupervised: learn $p(\mathbf{x})$
- Supervised: learn $p(\mathbf{y} \mid \mathbf{x})$
- *Form design matrix:* $\boldsymbol{X} \in \mathbb{R}^{150 \times 4}$
    - *150 examples (rows); 4 features (columns)*

(Goodfellow 2017)

Generalization

- In practice: minimize training error
- Real goal: minimize generalization error
  - "expected value of the error on a new input"
- Assume: 1) examples independent
  2) test/training are identically distributed

(Goodfellow 2017)

Model Capacity

⬆ Increase number of features/parameters

⬆ Represent more functions (families of functions)

⬇ Imperfect optimization procedure

Capacity should fit the problem/data

(Goodfellow 2017)

Generalization and Capacity

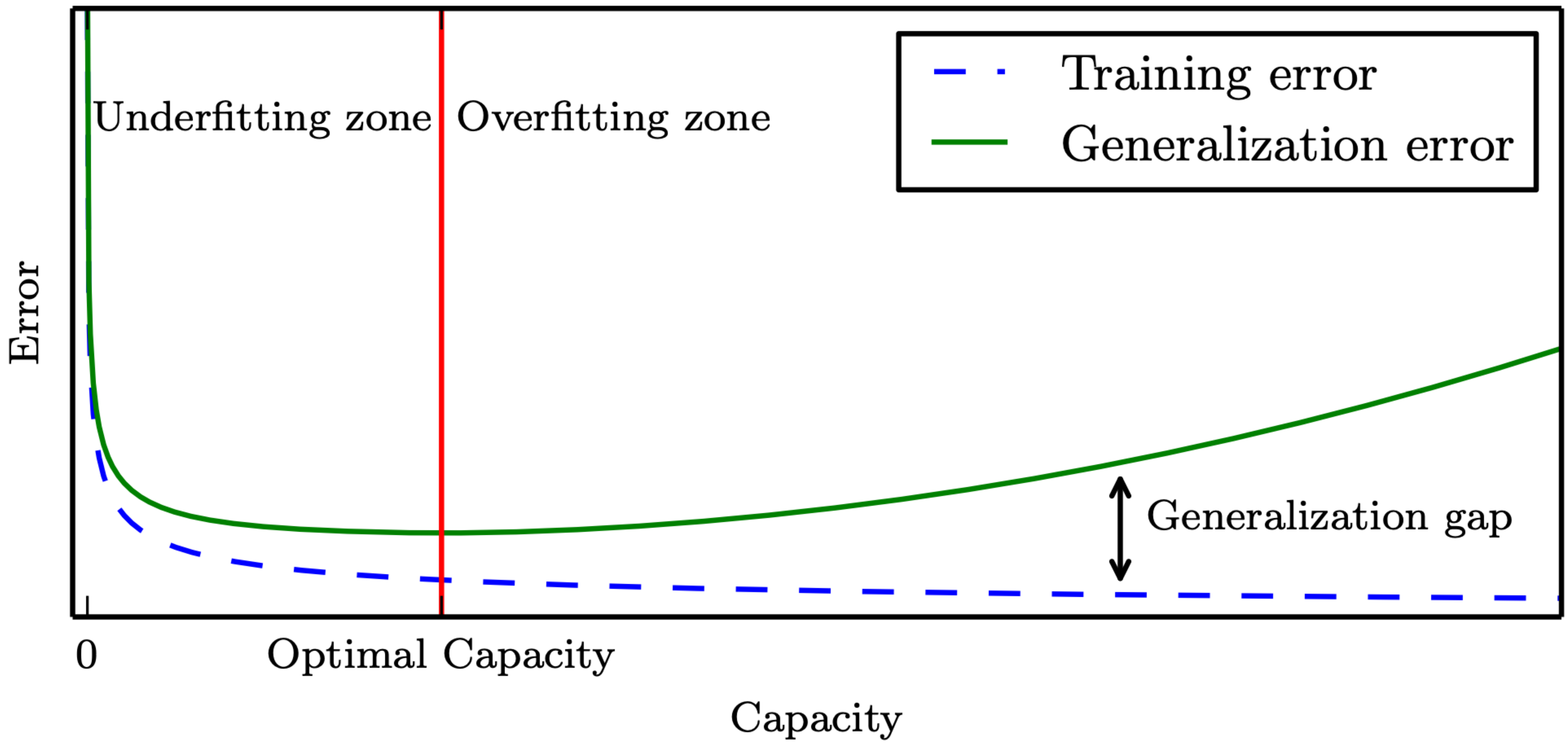


Figure 5.3

(Goodfellow 2017)

Increasing Model Capacity

Recall Polynomial Model

$$\hat{y} = \sum_j \beta_j x^j$$

Underfitting and Overfitting in Polynomial Estimation

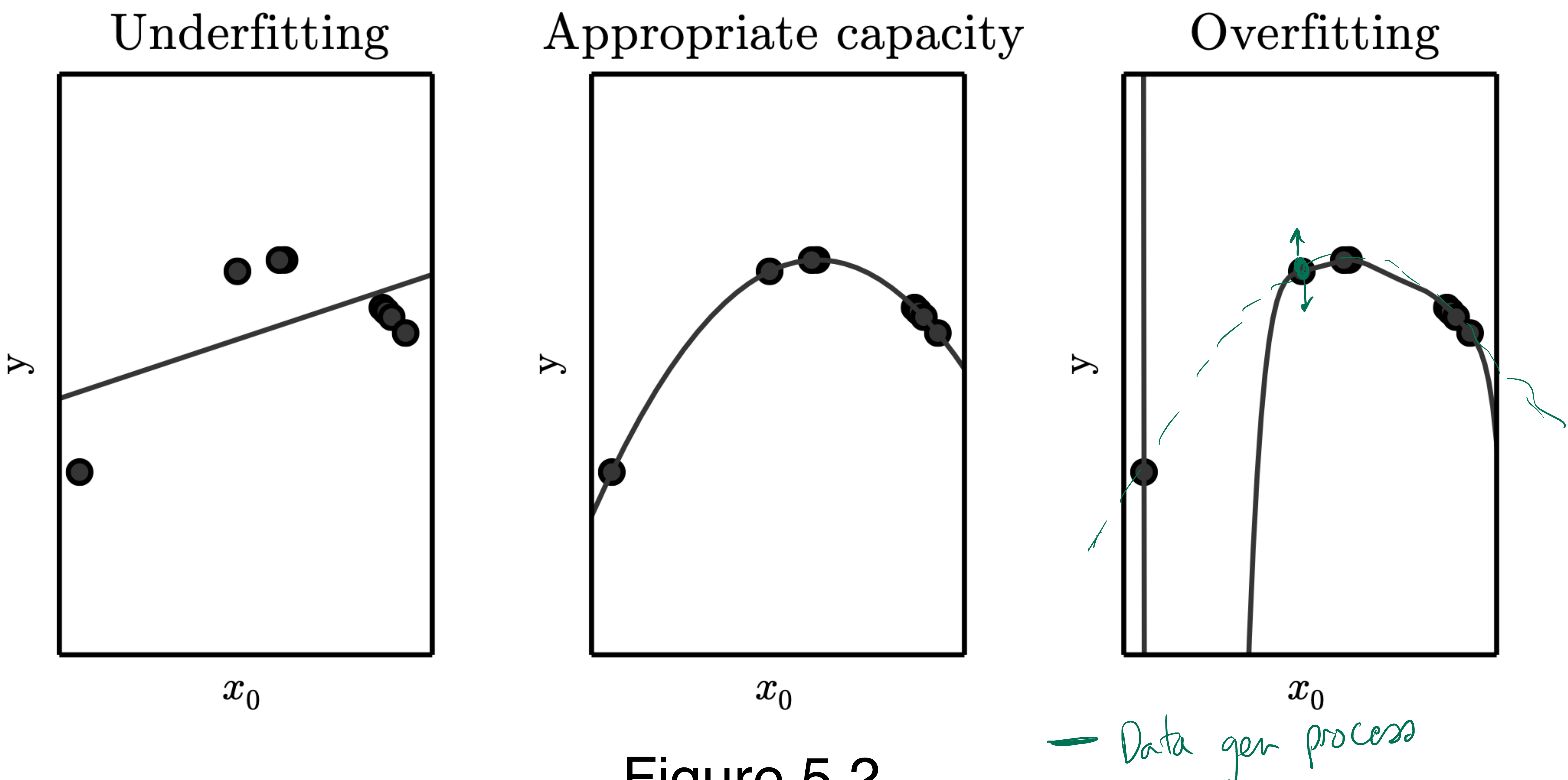


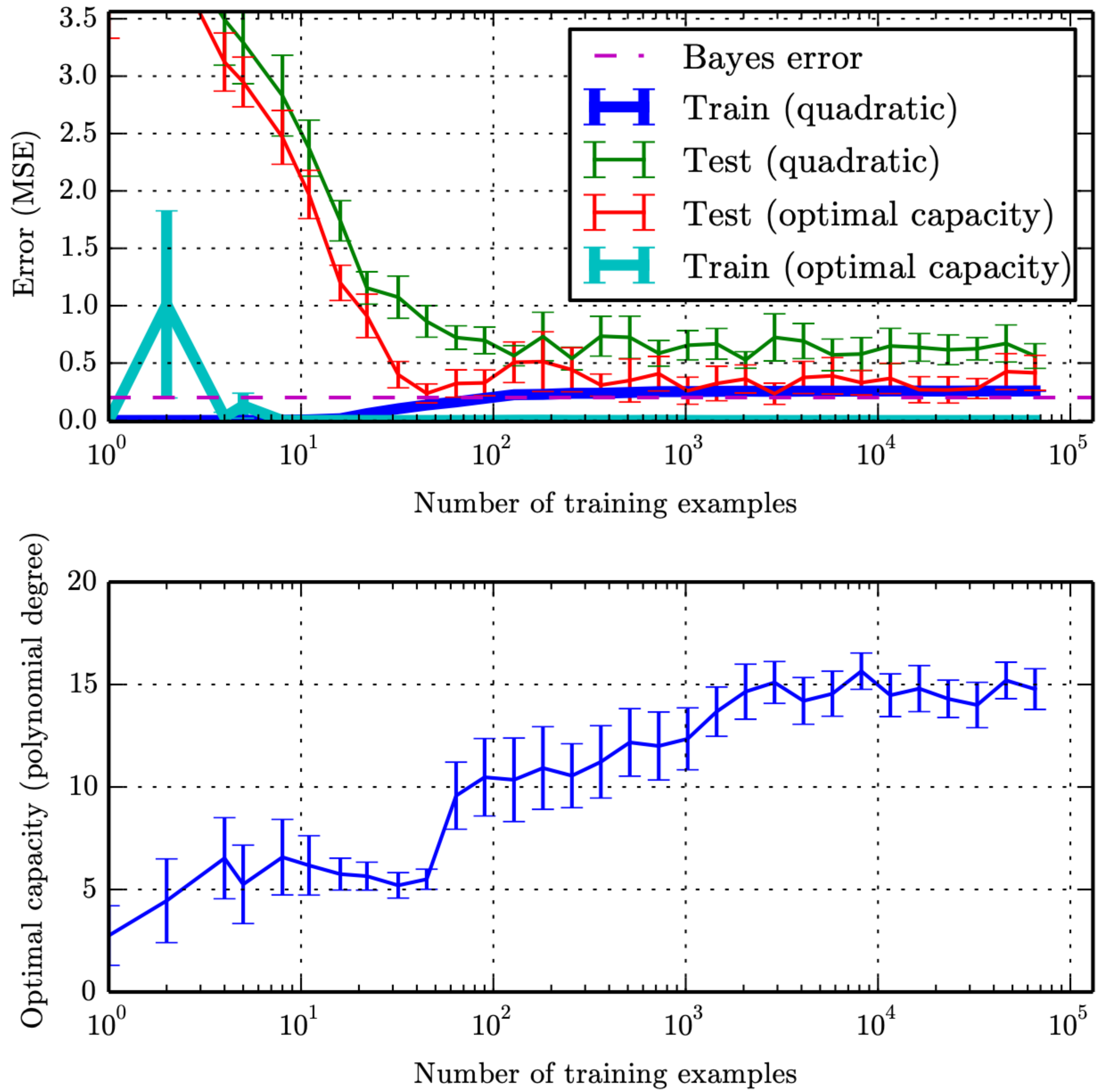
Figure 5.2

(Goodfellow 2017)

Training Set Size Example

synthetic regression problem for noisy degree-5 polynomial

Figure 5.4



(Goodfellow 2017)

Regularization

- We can "reduce" model capacity by:

1. Restricting the type and amount of functions

2. Allowing lots of functions, but providing a preference for one solution over another,
e.g. prefer solutions with smaller norm

$$J(\boldsymbol{w}) = \mathrm{MSE}_{\mathrm{train}} + \lambda \boldsymbol{w}^\top \boldsymbol{w}, \tag{5.18}$$

Regularization: any modification we make to a learning algorithm that is intended to reduce its generalization error but not its training error.

(Goodfellow 2017)

9-order polynomial

Weight Decay

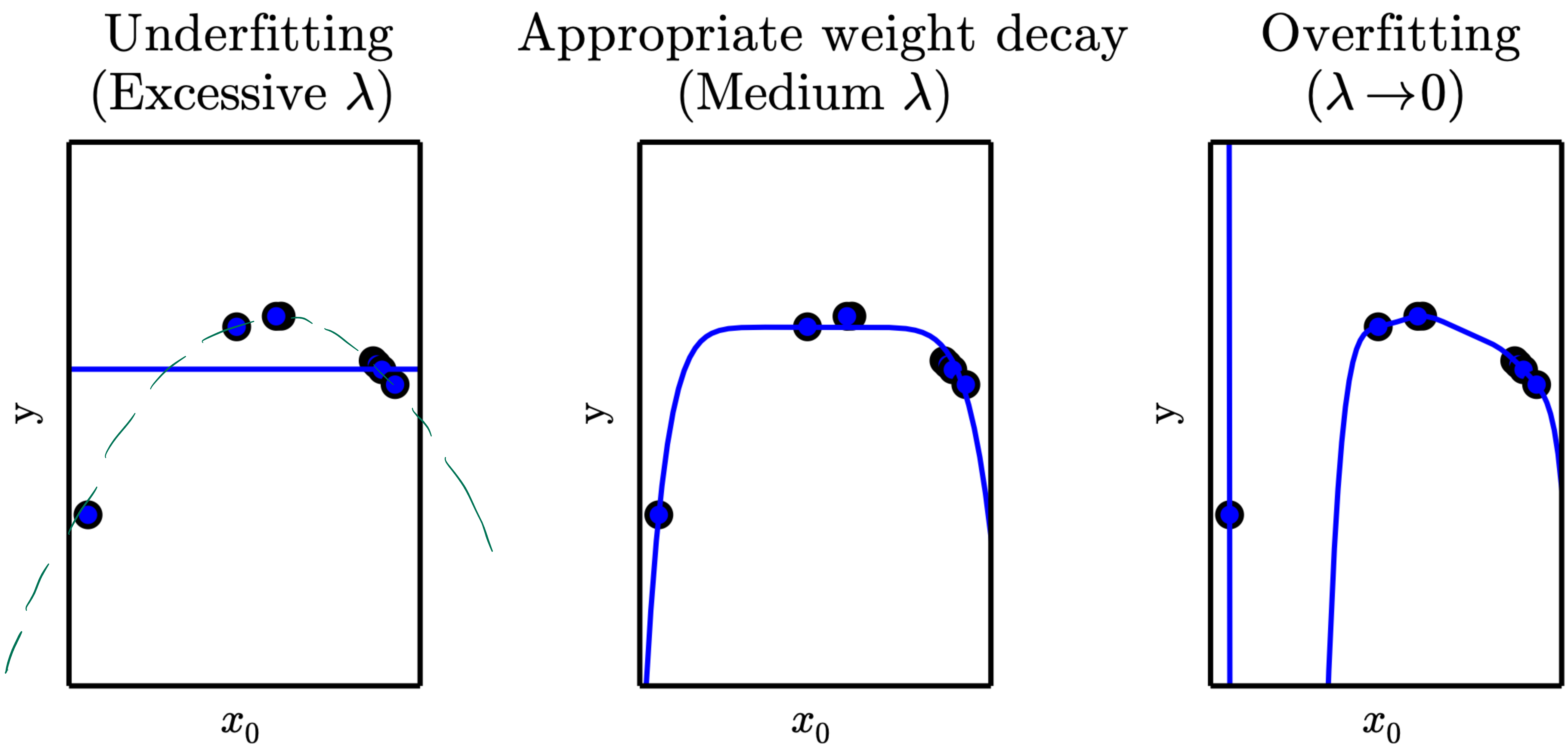


Figure 5.5

(Goodfellow 2017)

Ridge Regression

l_2-norm penalty on "weights"

$\text{loss} + \|w\|_2^2 \rightarrow \sum_i w_i^2$

w_i will be small

(Goodfellow 2017)

Lasso Regression

ℓ_1 - norm penalty on weights

$\text{loss} + \|w\|_1$ —— $\sum_i |w_i|$

w will be sparse

many w_i will be zero

(Goodfellow 2017)

Hyperparameters & Cross-Validation

- e.g., λ is a hyperparameter for regularization
- Usually many other hyperparameters
- Use ***validation set*** that the training algo doesn't see
- For small test sets: k-fold cross-validation

(Goodfellow 2017)