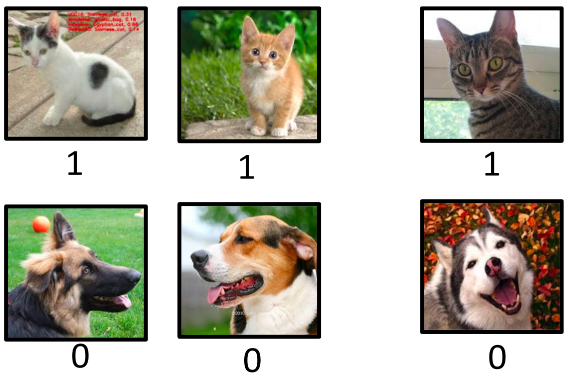
Recap: ML basics

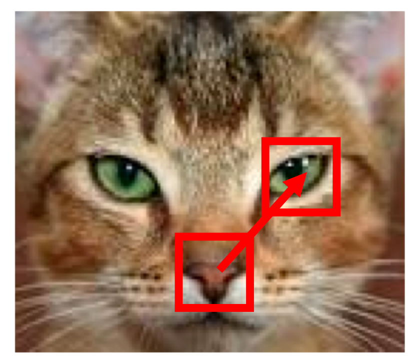
Different settings of learning



Supervised

Unsupervised
Clustering

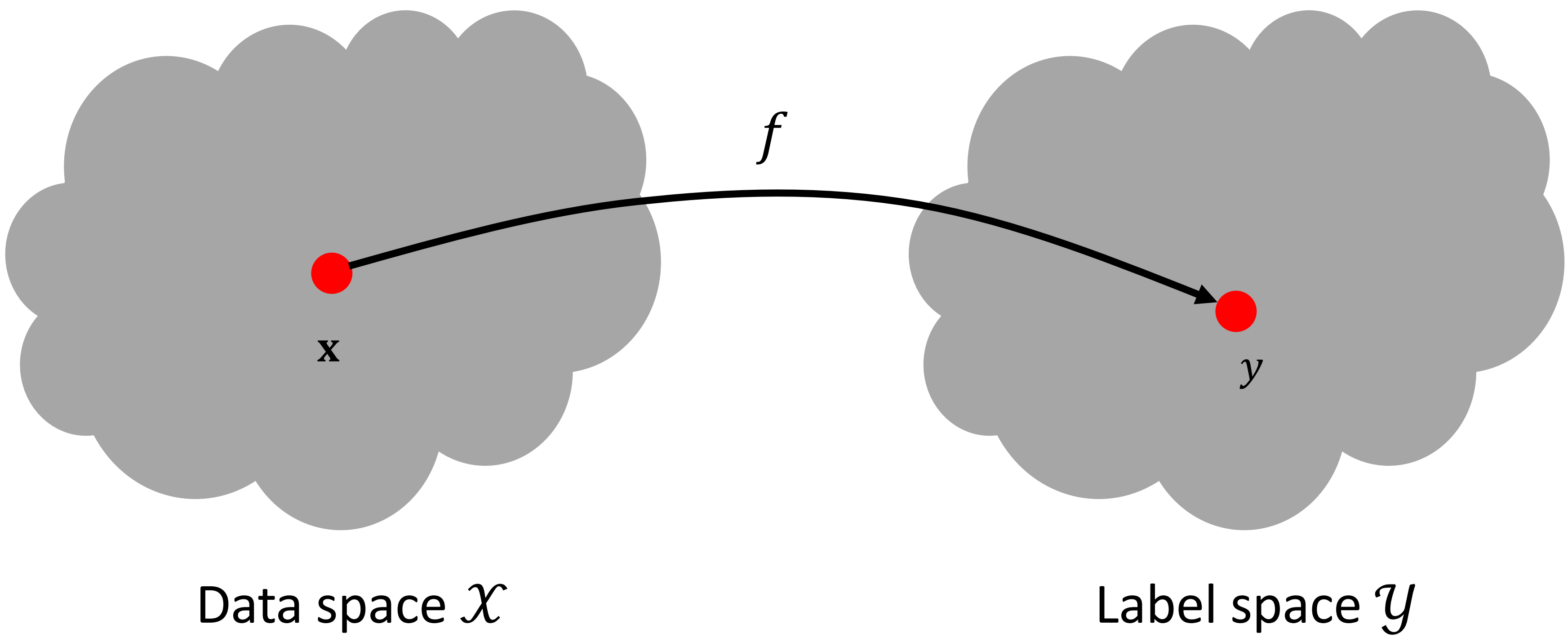
Semi-supervised
Partially labelled



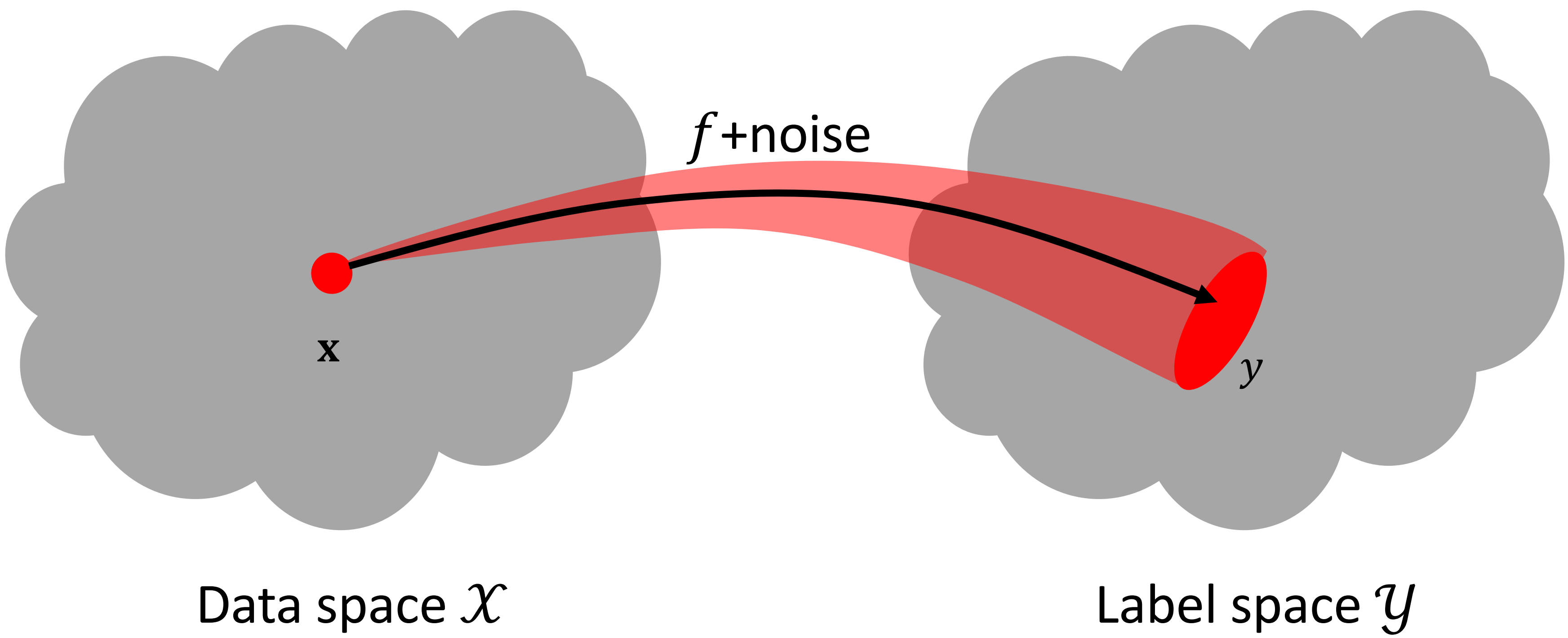
Self-supervised
Proxy task

Model + Data + Optimisation

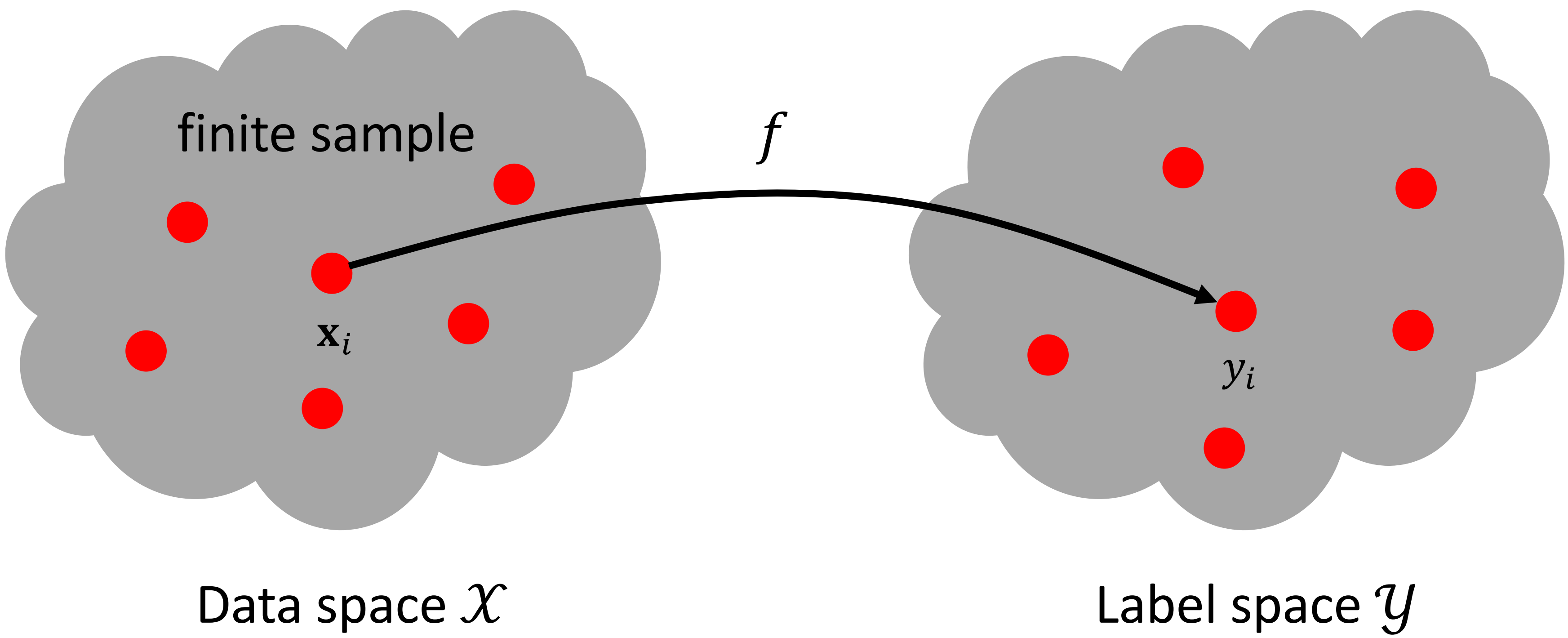
Supervised ML problem



Supervised ML problem

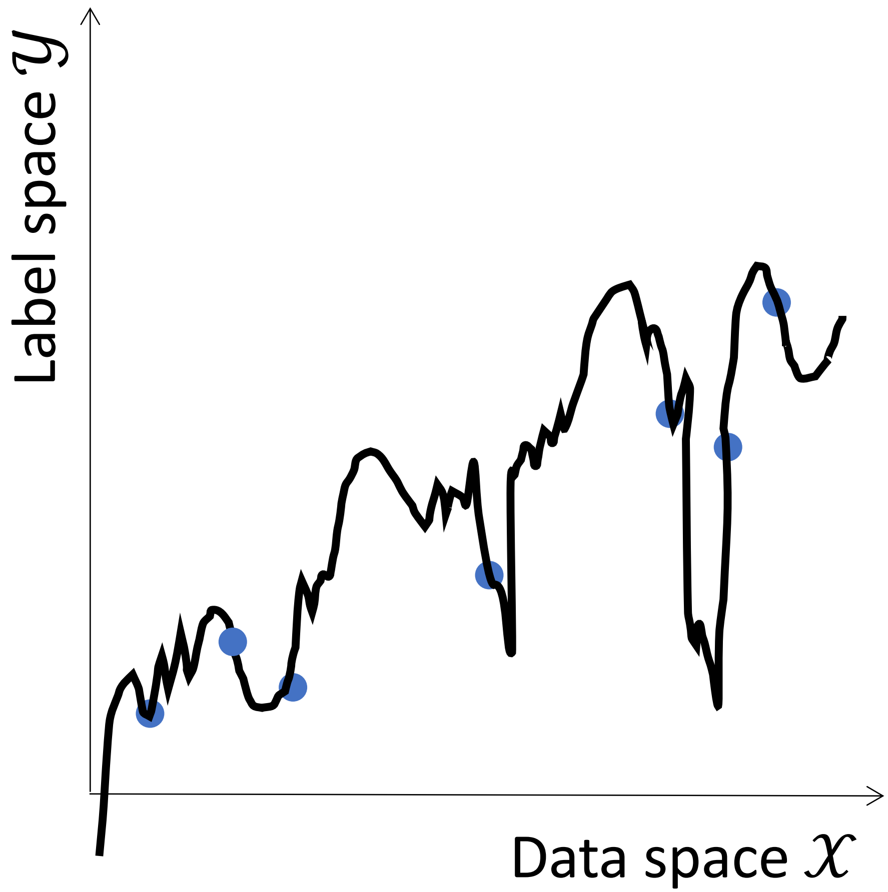


Supervised ML problem

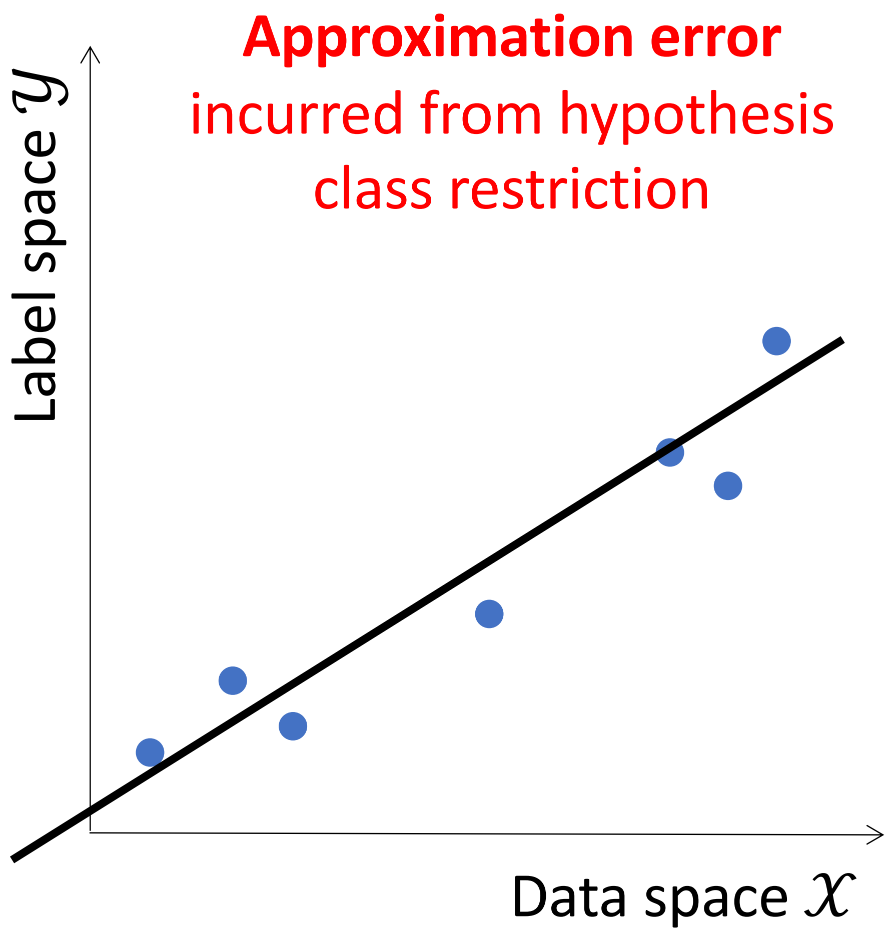


estimate f from finite sample

Function approximation

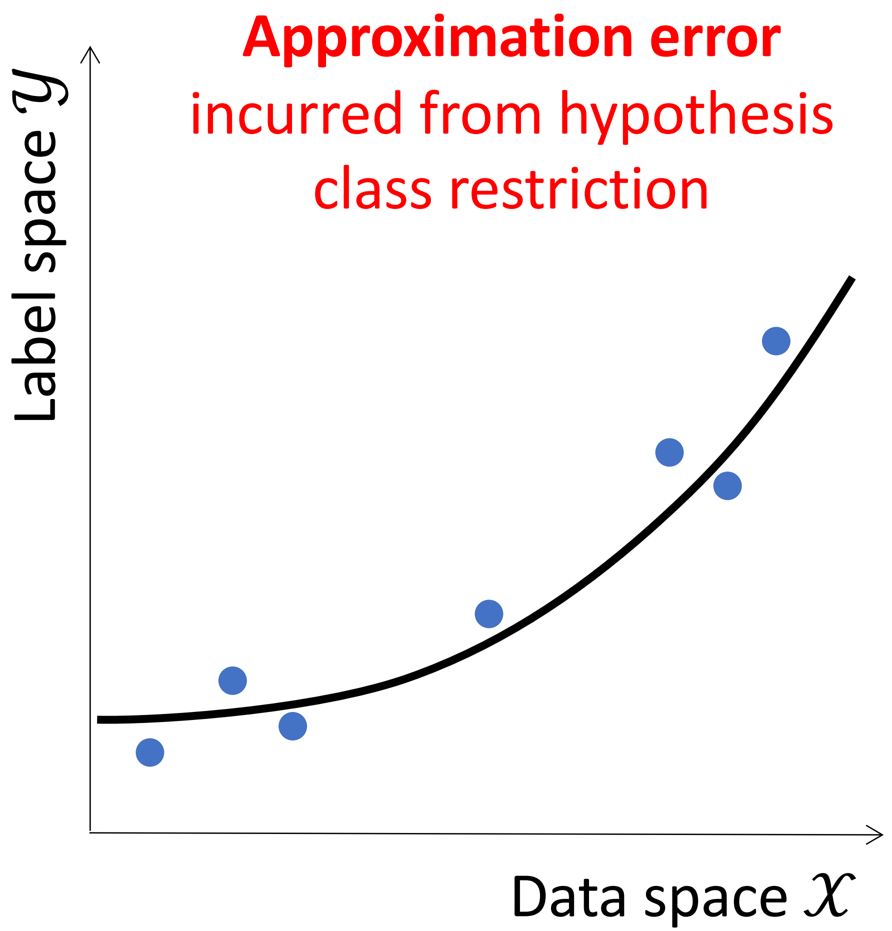


Function approximation



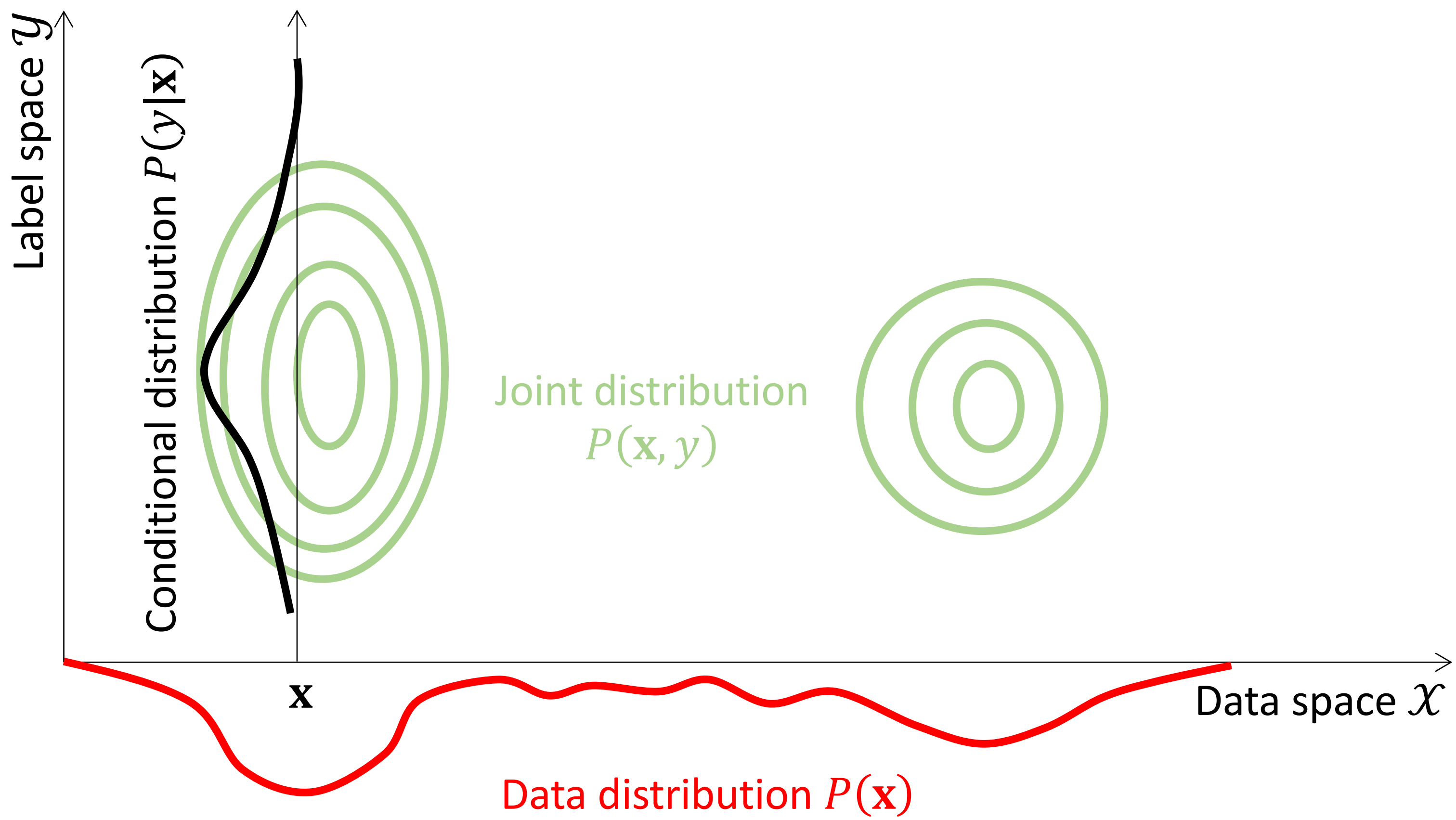
f ∈ hypothesis class (typically parametric)

Function approximation



f ∈ hypothesis class (typically parametric)

Probabilistic estimation



estimate $P(y|\mathbf{x})$ from finite sample

Probabilistic estimation

Parametric model with parameters $\boldsymbol{\theta}$

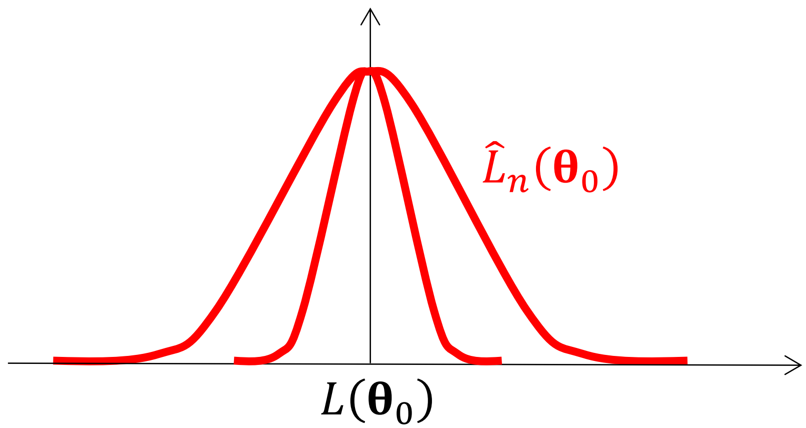
Estimate $p_{\boldsymbol{\theta}}(y|\mathbf{x})$ from finite sample by minimizing the loss

$$L(\boldsymbol{\theta}) = -\mathbb{E}_{y|\mathbf{x}\sim p} \log p_{\boldsymbol{\theta}}(y|\mathbf{x})$$

$$\approx -\frac{1}{n}\sum_{i=1}^{n} \log p_{\boldsymbol{\theta}}(y_i|\mathbf{x}_i) = \hat{L}_n(\boldsymbol{\theta})$$

Estimation error (or generalization gap): incurred by using empirical finite-sample loss $\hat{L}$ instead of expected loss L

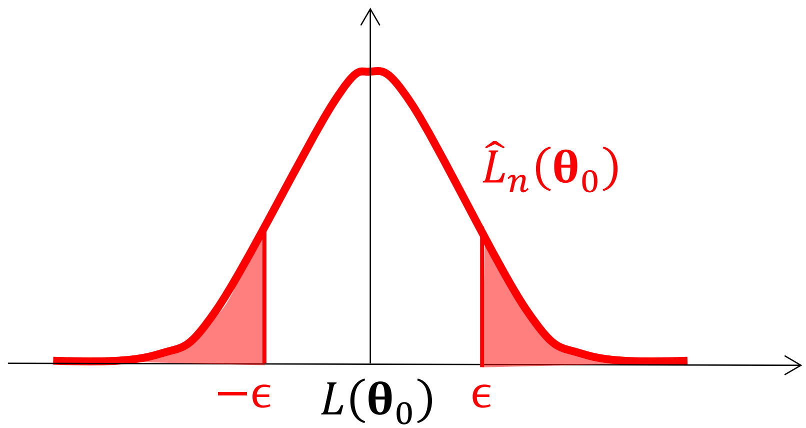
Generalization gap



$\hat{L}_n(\boldsymbol{\theta})$ is a random variable (depends on sampling)

For given parameters $\boldsymbol{\theta}_0$, $\hat{L}_n(\boldsymbol{\theta}_0)$ concentrates around expected value $L(\boldsymbol{\theta}_0)$ as $n \to \infty$ (**law of large numbers**)

Generalization gap



Hoeffding inequality

$$P\left(\left|L(\boldsymbol{\theta}_0) - \hat{L}_n(\boldsymbol{\theta}_0)\right| > \epsilon\right) \leq 2e^{-2\epsilon^2 n}$$

"probably approximately correct"

- reducing tolerance ϵ 10 fold requires 100 times larger sample n

Generalization bound

Search over the whole space of parameters $\boldsymbol{\theta} \in \mathcal{H}$

$$
\begin{aligned}
P\left(\left|L(\boldsymbol{\theta}^*) - \hat{L}_n(\boldsymbol{\theta}^*)\right| > \epsilon\right) &\leq P\left(\sup_{\boldsymbol{\theta}\in\mathcal{H}} |L(\boldsymbol{\theta}) - \hat{L}_n(\boldsymbol{\theta})| > \epsilon\right) \\
&= P\left(\bigcup_{\boldsymbol{\theta}\in\mathcal{H}} \left\{\left|L(\boldsymbol{\theta}) - \hat{L}_n(\boldsymbol{\theta})\right| > \epsilon\right\}\right) \\
&\leq \sum_{\boldsymbol{\theta}\in\mathcal{H}} P\left(\left|L(\boldsymbol{\theta}) - \hat{L}_n(\boldsymbol{\theta})\right| > \epsilon\right) \\
&\leq 2|\mathcal{H}|e^{-2\epsilon^2 n}
\end{aligned}
$$

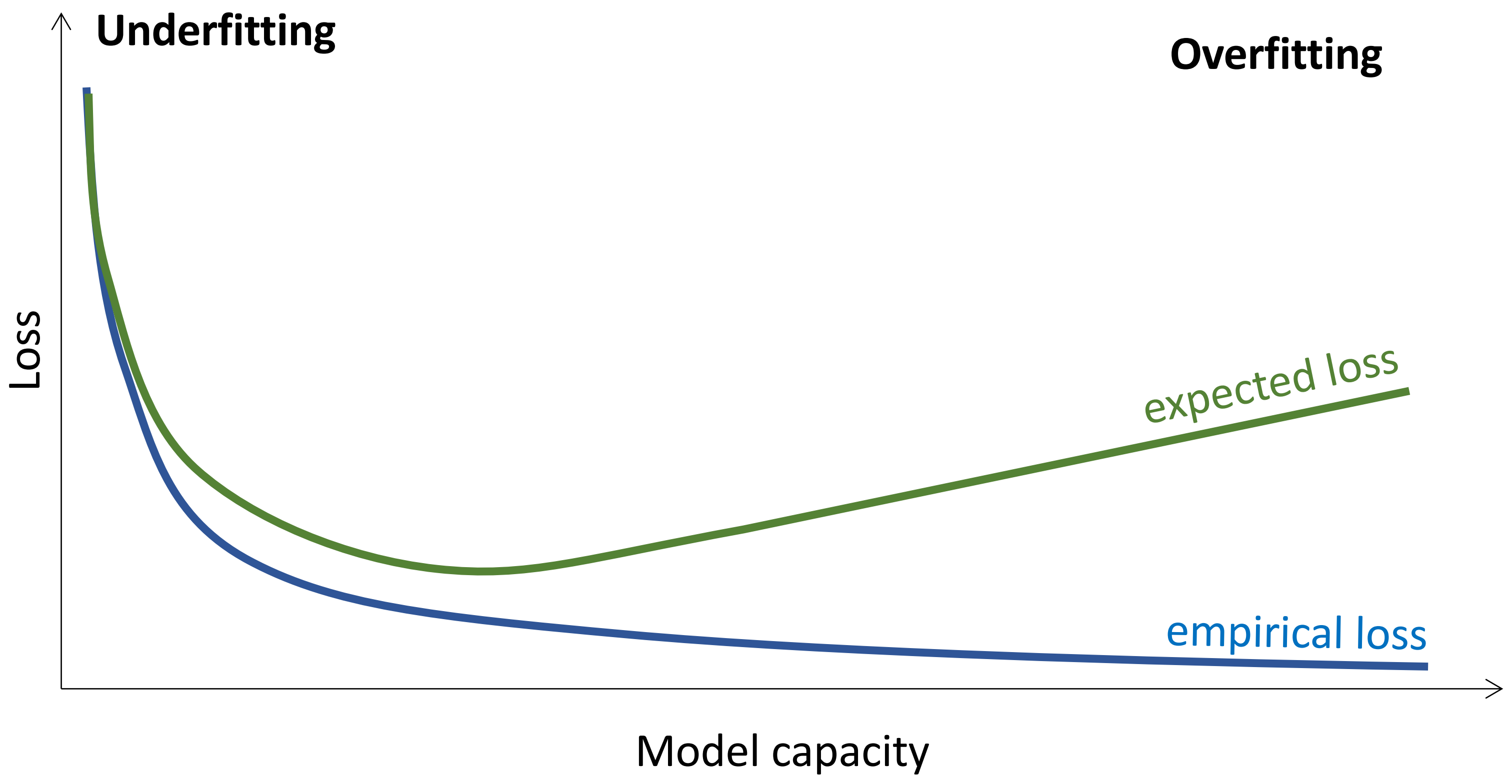
More meaningful bounds: **Vapnik-Chervonenkis, Rademacher**

Approximation error vs Estimation error

- Incurred by restricting model to hypothesis class
- **Decreased** by using richer hypothesis class

- Incurred by using empirical loss instead of expected loss
- **Increased** by using richer hypothesis class
- **Decreased** by using larger sample size n

Overfitting and underfitting



Main ingredients of an ML problem

$$\widehat{\boldsymbol{\theta}} = \operatorname*{argmin}_{\boldsymbol{\theta}} \frac{1}{n}\sum_{i=1}^{n} \ell(f_{\boldsymbol{\theta}}(\mathbf{x}_i), y_i)$$

- **Data** (training/test set, features)
- **Model/hypothesis class** $f_{\boldsymbol{\theta}}$ and **loss function** ℓ
- **Optimisation** (how to find best model parameters $\widehat{\boldsymbol{\theta}}$)

Function approximation

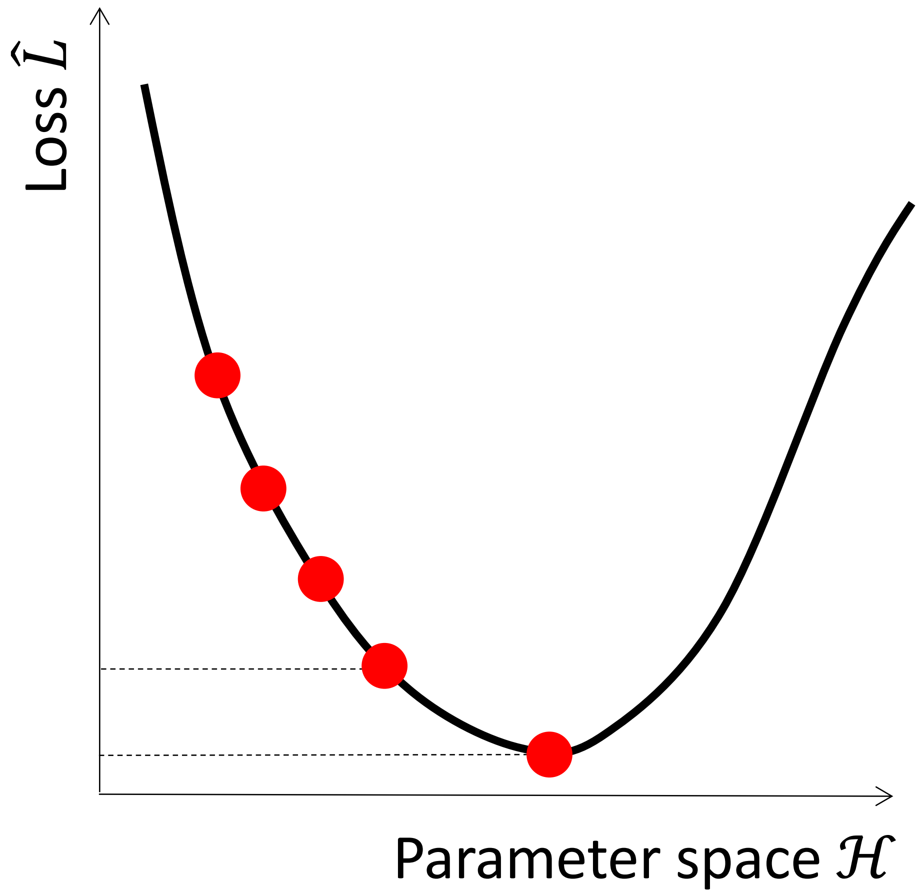
Statistical estimation

Optimisation theory

Optimisation error

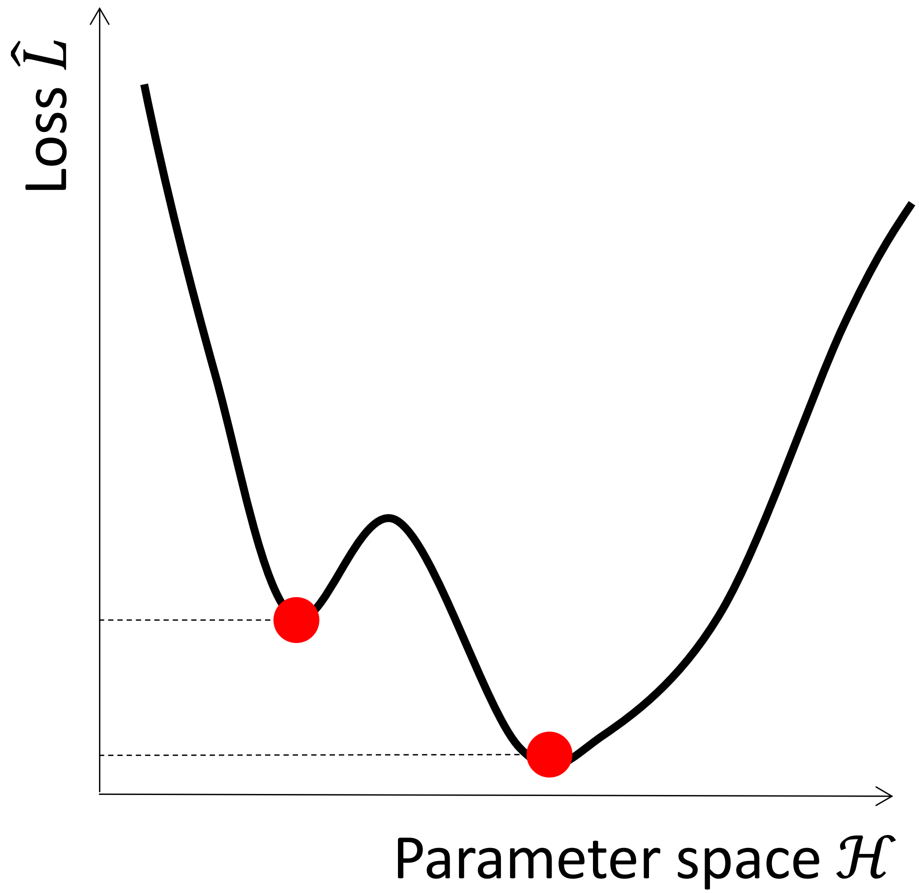
Optimisation error

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \hat{L}_n(\boldsymbol{\theta})$$



Optimisation error

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \hat{L}_n(\boldsymbol{\theta})$$



Estimation error vs Optimisation error

Estimation error

- Incurred by using empirical loss instead of expected loss
- **Decreased** by using larger sample size n

Optimisation error

- Incurred by not finding exact minimiser of empirical loss
- **Decreased** by longer compute time (number of iterations)