Maximum likelihood estimation

- Estimates the parameters, θ, of a distribution using a likelihood function, $\mathscr{L}(\mathbf{D}; \theta)$, given some data, **D.**

- We maximize the likelihood function by minimizing its -logarithm:

$$\mathscr{L}(\theta \,|\, \mathbf{D}) = P(\mathbf{D}; \theta)$$

$$= \left(\prod_{i=1}^{M} P(\mathbf{y}_i \,|\, \mathbf{x}_i; \theta) \right)^{\frac{1}{M}}$$

$$= \frac{1}{M} \sum_{i=1}^{M} \log P(\mathbf{y}_i \,|\, \mathbf{x}_i; \theta) \qquad \text{Note: technically natural log, but true to within a constant}$$

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} \sum_{i=1}^{M} -\log P(\mathbf{y}_i \,|\, \mathbf{x}_i; \theta)$$

- This expresses an optimization problem; the form of $p(\mathbf{D}; \theta)$ dictates how we solve it
  - In deep learning, this function is a neural network; we compute its gradient w.r.t. θ, and then estimate $\hat{\theta}$ using stochastic gradient descent (SGD).

Refresher: Neural Networks