Negative Log Likelihood

Likelihood: $L(\text{model} \mid \text{data}) = P(\text{data} \mid \text{model})$

- How likely is a model given this data? ⇒ Used to compare diff. models
- Assume data is fixed → find probability of getting this data based on model.

In ML, we want to maximize $L(\theta \mid x_1, x_2, ..)$

- Given data is i.i.d., $L(\theta \mid x_1, x_2, \ldots) = L(\theta \mid x_1) \cdot L(\theta \mid x_2) \cdot \ldots \cdot L(\theta \mid x_n)$
  ⇒ Problem of <u>underflow</u>! Keep multiplying probs = v. small value
- Log Likelihood turns products into sums:
  $\Rightarrow L(\theta \mid x_1, x_2 ..) = \prod_{i=1}^{n} L(\theta \mid x_i)$
  $\Rightarrow \log L(\theta \mid x_1, x_2, ..) = \sum_{i=1}^{n} \log L(\theta \mid x_i)$
- Hence we minimize "negative log likelihood" ⇔ maximize $\log L(\theta \mid x_1, x_2, \ldots)$

NLL is the same as Cross-Entropy:

- Let actual vals $= y^{(n)} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$ → one-hot vector
- Let predicted vals $= \hat{y}^{(n)} = \text{softmax}(\vec{z}) = \frac{1}{\sum e^{z_i}} \begin{bmatrix} e^{z_1} \\ e^{z_2} \\ \vdots \end{bmatrix}$ → softmax is great as these are interpreted as probabilities!
- $NLL = -\log P_\theta(y \mid x_1, x_2, \ldots)$

  $= -\log \prod_{n=1}^{N} P_\theta(y^{(n)} \mid x^{(n)})$

  $= -\sum_{n=1}^{N} \log P_\theta(y^{(n)} \mid x^{(n)})$ → prob my model assigns to $y^{(n)}$ given $x^{(n)}$

  $= -\sum_{n=1}^{N} \sum_{k=1}^{K} \underline{y_k^{(n)} \cdot \log \hat{y}_k^{(n)}}$ → equal to prob. assigned to corr. class in softmax!! ($y_k^{(n)}$ = one-hot; extracts prob of corr. class)

  This is CROSS-ENTROPY!!

  $H(p,q) = -\sum_{k=1}^{K} p_k \log q_k$