Markov networks: Learning
Derivative of the Log-Likelihood
Derivative of Log-likelihood: Part 1

★ Write the Markov network as a log-linear model

\[ P_\theta(x) = \frac{1}{Z(\theta)} \exp \left( \sum_i \theta_i f_i(x_i) \right) \]

where \( f_i \) is a feature, namely a 0/1 function (SAT formula), \( \theta_i \) is the weight associated with feature \( f_i \), \( x_i \) is the projection of \( x \) on \( f_i \) and \( Z(\theta) \) is the partition function.

★ Log-likelihood of the log-linear model given data \( D = (x^{(1)}, \ldots, x^{(M)}) \)

\[
LL(\theta : D) = \ln \left( \prod_{j=1}^{M} P_\theta(x^{(j)}) \right) \\
= \left[ \sum_i \theta_i \left( \sum_{j=1}^{M} f_i(x^{(j)}_i) \right) \right] - M \ln Z(\theta)
\]
Derivative of Log-likelihood: Part 2

- For convenience: Rewrite the Log-likelihood by dividing both sides by $M$

\[
\frac{1}{M} LL(\theta : D) = \frac{1}{M} \left[ \sum_i \theta_i \left( \sum_{j=1}^{M} f_i(x_i^{(j)}) \right) \right] - \ln Z(\theta)
\]

- The first expression on the right hand side is expected value of the feature from the data, multiplied by $\theta_i$. Therefore, we can rewrite the Likelihood as:

\[
\frac{1}{M} LL(\theta : D) = \sum_i \theta_i \mathbb{E}_D[f_i(x_i^{(j)})] - \ln Z(\theta)
\]
Derivative of Log-likelihood: Part 3

- Taking partial derivative with respect to $\theta_i$

$$\frac{\partial}{\partial \theta_i} \frac{1}{M} \log \text{LL}(\theta : \mathcal{D}) = \frac{\partial}{\partial \theta_i} \sum_i \theta_i \mathbb{E}_\mathcal{D}[f_i(x_i^{(j)})] - \frac{\partial}{\partial \theta_i} \ln Z(\theta)$$

- The first expression on the RHS equals $\mathbb{E}_\mathcal{D}[f_i(x_i^{(j)})]$. The estimate of this value is the number of times the feature $f_i$ is true in the data! Nice, easy to compute.

- The second expression, we have to derive separately
Derivative of Log-likelihood: Part 5

- Recall that $Z(\theta)$ is given by:

$$Z(\theta) = \sum_x \exp \left( \sum_i \theta_i f_i(x_i) \right)$$

- Partial derivative of $\ln Z(\theta)$:

$$\frac{\partial \ln Z(\theta)}{\partial \theta_i} = \frac{1}{Z(\theta)} \frac{\partial}{\partial \theta_i} Z(\theta)$$

$$= \frac{1}{Z(\theta)} \sum_x \frac{\partial}{\partial \theta_i} \exp \left( \sum_i \theta_i f_i(x_i) \right)$$

$$= \frac{1}{Z(\theta)} \sum_x \exp \left( \sum_i \theta_i f_i(x_i) \right) \frac{\partial}{\partial \theta_i} \sum_i \theta_i f_i(x_i)$$

$$= \sum_x \left[ \frac{1}{Z(\theta)} \exp \left( \sum_i \theta_i f_i(x_i) \right) \right] f_i(x_i) = \sum_x P_\theta(x) f_i(x_i)$$

$$= \mathbb{E}_{P_\theta}[f_i(x_i)]$$
In summary, we have the following expression:

$$\frac{1}{M} \frac{\partial}{\partial \theta_i} LL(\theta : D) = \mathbb{E}_D[f_i(x_i)] - \mathbb{E}_{P_\theta}[f_i(x_i)]$$

The first expression on the RHS is the number of times feature \( f_i \) is true in the data.

The second expression is the expected number of times feature \( f_i \) is true given current set of parameters \( \theta \). This requires inference over the model (sum-product inference).

A simple gradient ascent procedure:

- Start with random parameters.
- At each iteration \( t \), update each \( \theta^t_i \) using:

  $$\theta^{t+1}_i = \theta^t_i + \eta \left( \mathbb{E}_D[f_i(x_i)] - \mathbb{E}_{P_\theta}[f_i(x_i)] \right)$$

- Stop when converged.