COMPUTING GRADIENTS OF TENSOR NETWORKS

I Jacobian and backpropagation

- $f : \mathbb{R}^m \to \mathbb{R}$ Gradient $\nabla_\theta f = \begin{pmatrix} \partial f/\partial \theta_1 \\ \vdots \\ \partial f/\partial \theta_m \end{pmatrix}$ for each $\theta \in \mathbb{R}^m$

- $f : \mathbb{R}^m \to \mathbb{R}^n$ Jacobian $\dfrac{\partial f}{\partial \theta} = \left(\dfrac{\partial f(\theta)_i}{\partial \theta_j} \right)_{i,j} \in \mathbb{R}^{n \times m}$ for each $\theta \in \mathbb{R}^m$

- Backpropagation / Automatic Differentiation (reverse mode).

Computational Graph

$$\begin{array}{ccccccccc} \theta & \to & T^1 & \to & T^2 & \to & T^3 & \to & \mathcal{L} \\ \mathbb{R}^m & \to & \mathbb{R}^{d_1} & \to & \mathbb{R}^{d_2} & \to & \mathbb{R}^{d_3} & \to & \mathbb{R} \end{array}$$

(where $X \to Y$ means that Y is a function of X)

We want to compute $\dfrac{\partial \mathcal{L}}{\partial \theta}$.

Chain rule: $\bar{\theta} \to \underbrace{\dfrac{\partial \mathcal{L}}{\partial \theta}}_{1 \times m} = \underbrace{\dfrac{\partial \mathcal{L}}{\partial T_3}}_{1 \times d_3} \underbrace{\dfrac{\partial T_3}{\partial T_2}}_{d_3 \times d_2} \underbrace{\dfrac{\partial T_2}{\partial T_1}}_{d_2 \times d_1} \underbrace{\dfrac{\partial T_1}{\partial \theta}}_{d_1 \times m}$

We define the adjoint: $\bar{T} = \dfrac{\partial \mathcal{L}}{\partial T}$

$\bar{T}_3 = \dfrac{\partial \mathcal{L}}{\partial T_3}$

$$\bar{T}_2 = \frac{\partial \mathcal{L}}{\partial T_2} = \frac{\partial \mathcal{L}}{\partial T_3} \frac{\partial T_3}{\partial T_2} = \bar{T}_3 \frac{\partial T_3}{\partial T_2}$$

$$\bar{T}_1 = \frac{\partial \mathcal{L}}{\partial T_1} = \frac{\partial \mathcal{L}}{T_2} \frac{\partial T_2}{\partial T_1} = \bar{T}_2 \frac{\partial T_2}{\partial T_1}$$

$$\bar{\theta} = \cdots = \bar{T}_1 \frac{\partial T_1}{\partial \theta}$$

$$\boxed{\bar{T}^j = \bar{T}^{j+1} \frac{\partial T^{j+1}}{\partial T_j}}$$

More complex graphs : $\theta \to T_1 \to T_2 \to T_4 \to \mathcal{L}$, $T_1 \to T_3 \to T_4$

$$\boxed{\overline{T_i} = \sum_{j:\text{child of } i} \overline{T_j}\,\frac{\partial T_j}{\partial T_i}}$$

ex: $\overline{T_1} = \frac{\partial \mathcal{L}}{\partial T_1} = \frac{\partial \mathcal{L}}{\partial T_2}\frac{\partial T_2}{\partial T_1} + \frac{\partial \mathcal{L}}{\partial T_3}\frac{\partial T_3}{\partial T_1}$

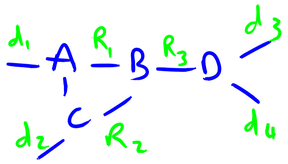
$= \overline{T_2}\frac{\partial T_2}{\partial T_1} + \overline{T_3}\frac{\partial T_3}{\partial T_1}$

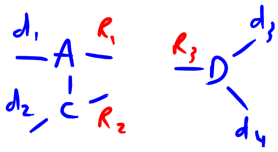
- $f : \mathbb{R}^{m_1 \times m_2 \times \dots \times m_N} \to \mathbb{R}^{h_1 \times h_2 \times \dots \times h_p}$ $\left(\cong \mathbb{R}^{m_1 m_2 \cdots m_N} \to \mathbb{R}^{h_1 h_2 \cdots h_p}\right)$

Jacobian Tensor $\frac{\partial f(T)}{\partial T} = \left(\frac{\partial f(T)_{i_1 \dots i_p}}{\partial T_{j_1 \dots j_N}}\right)_{i_1 \dots i_p j_1 \dots j_N} \in \mathbb{R}^{h_1 \times \dots \times h_p \times m_1 \times \dots \times m_N}$

II Jacobian of Tensor Networks

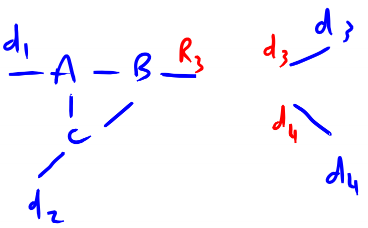
Let W be a tensor given as a tensor network where G is a core tensor appearing only once. Then $\frac{\partial W}{\partial G}$ is simply obtained by deleting G in the tensor network.

Ex: • $\underset{d_1 d_2 d_3 d_4}{W} =$ 

$\frac{\partial W}{\partial B} =$ 

$d_1 \times d_2 \times d_3 \times d_4 \times R_1 \times R_2 \times R_3$

Jacobian of $\underset{R_1 \times R_2 \times R_3}{B} \mapsto \underset{d_1 \times d_2 \times d_3 \times d_4}{W}$

$\frac{\partial W}{\partial D} =$ 

$d_1 \times d_2 \times d_3 \times d_4 \times R_3 \times d_3 \times d_4$

- Rederiving classical matrix / vector gradients :

$\frac{\partial \langle u, v\rangle}{\partial u} = \partial(u - v)/\partial u = -v = v$

$\frac{\partial Ax}{\partial x} = \partial(-A-x)/\partial x = -A- = A$

$\frac{\partial x^T A x}{\partial A} = \partial(x-A-x)/\partial A = x- \quad -x = xx^T$

$\frac{\partial \operatorname{Tr}(A)}{\partial A} = \partial(A)/\partial A = \quad = I$

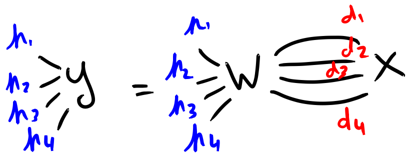
$\frac{\partial Ax}{\partial A} = \partial(-A-x)/\partial A = - \quad -x \cong I \circ x$

If the core tensor G appears k times in the tensor network of W, then $\frac{\partial W}{\partial G}$ is given the sum of k copies of W where a different occurrence of G is deleted in each copy.

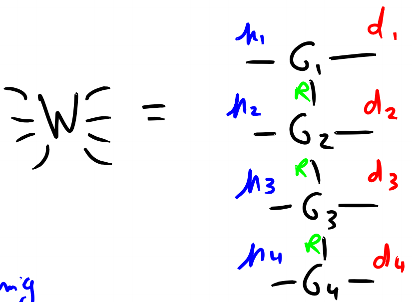
$$\underline{\underline{Ex}}: \quad \frac{\partial\, x^T A x}{\partial x} = \partial\left(x \,{}_1\!-\!A\!-_2\, x\right)/\partial x = {}_1\!-\!A\!-_2\, x + x \,{}_1\!-\!A\!-_2 = (A + A^T)\, x$$

Applications

TENSORIZING NEURAL NETWORKS :



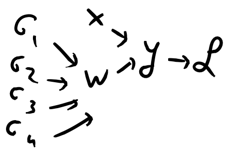
where



↳ We want to compute $\frac{\partial \mathcal{L}}{\partial G_2} = \frac{\partial \mathcal{L}}{\partial y}\boxed{\frac{\partial y}{\partial G_2}}$

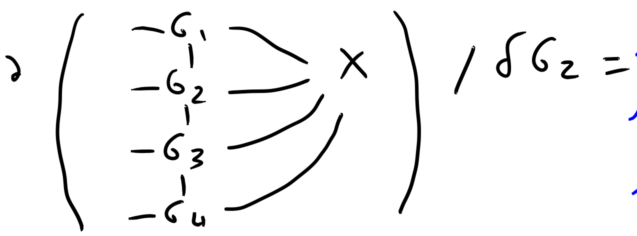
Some loss depending on W

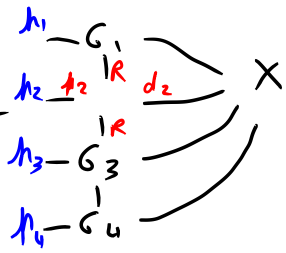
Comp. Graph



We need to compute $\frac{\partial y}{\partial G_2} \in \mathbb{R}^{h_1 \times h_2 \times h_3 \times h_4 \times R \times h_2 \times d_2 \times R}$. We have :

$$\frac{\partial y}{\partial G_2} = \partial$$





IV Generalized CP decomposition

GENERALIZED CANONICAL POLYADIC TENSOR DECOMPOSITION*

DAVID HONG†, TAMARA G. KOLDA‡, AND JED A. DUERSCH§

↳ Likelihood maximization & generalized linear model.

Observations/Data : $\mathcal{X} \in \mathbb{R}^{m_1 \times \cdots \times m_d}$

Parameterized PDF : $x_{i_1 \dots i_d} \sim p(x_{i_1 \dots i_d} \mid \theta_{i_1 \dots i_d})$

where $\ell(\theta_{i_1 \dots i_d}) = m_{i_1 \dots i_d}$

$\ell : \mathbb{R} \to \mathbb{R}$ is an invertible link function connecting the model parameters $m_{i_1 \dots i_d}$ to the "natural" parameter of the distribution.

Model parameters : $\mathcal{M} \in \mathbb{R}^{m_1 \times \cdots \times m_d}$

Goal : Find $\mathcal{M}$ that maximizes the likelihood of the observed entries of $\mathcal{X}$.

↳ $\Omega \subseteq [m_1] \times \cdots \times [m_d]$

$$\max_{\mathcal{M}} \mathcal{L}(\mathcal{M}; \mathcal{X}, \Omega) \equiv \prod_{(i_1 \dots i_d) \in \Omega} p(x_{i_1 \dots i_d} \mid \theta_{i_1 \dots i_d})$$

$$\text{where } \ell(\theta_{i_1 \dots i_d}) = m_{i_1 \dots i_d}$$

→ We only have (at most) one observation for each entry of $\mathcal{X}$!

We will assume a low rank structure of $\mathcal{M}$ (interdependence between the $\theta_{i_1 \dots i_d}$'s)

↳ low CP rank of $\mathcal{M}$.

Reformulation of optimization problem :

$$\boxed{\begin{aligned} &\min_{\mathcal{M}} F(\mathcal{M}; \mathcal{X}, \Omega) = \sum_{(i_1 \dots i_d) \in \Omega} f(x_{i_1 \dots i_d}, m_{i_1 \dots i_d}) \quad \text{s.t. } \operatorname{rank}_{CP}(\mathcal{M}) \leq R \\ &\qquad \text{where } f(x, m) = -\log p(x \mid \ell^{-1}(m)) \end{aligned}}$$

↳ "loss function" (assumed differentiable)

Examples of data distribution

① Gaussian distribution (Gaussian MLE $\Longleftrightarrow$ Mean squared error)

$$X_{i_1 \cdots i_d} = M_{i_1 \cdots i_d} + \varepsilon_{i_1 \cdots i_d} \quad \text{where } \varepsilon_{i_1 \cdots i_d} \sim \mathcal{N}(0, \underbrace{\sigma^2}_{\text{constant}})$$

Equivalently: $X_{i_1 \cdots i_d} \sim \mathcal{N}(\theta_{i_1 \cdots i_d}, \sigma^2)$ where $\theta_{i_1 \cdots i_d} = M_{i_1 \cdots i_d}$

the link function is the identity $\ell(\theta) = \theta$

Simple derivation:

$$f(x, m) = -\log \mathcal{N}(x; m, \sigma^2) = \frac{(x-m)^2}{2\sigma^2} + \frac{1}{2}\log(2\pi\sigma^2)$$

$\hookrightarrow$ ignoring σ^2 and constants $\boxed{f(x, m) = (x-m)^2}$

$\Longrightarrow$ ☆ is then equivalent to CP decomposition.

② Bernoulli distribution.

If $X_{i_1 \cdots i_d} \in \{0,1\}$ (binary data), we can assume $X_{i_1 \cdots i_d} \sim \mathcal{B}(\theta_{i_1 \cdots i_d})$

$$p(x|\theta) = \theta^x (1-\theta)^{1-x} \qquad , x \in \{0,1\},\ \theta \in [0,1]$$

link function?

- identy $\rightsquigarrow$ we need to constrain $m \in [0,1]$
- odds ratio: $m = \ell(\theta) = \dfrac{\theta}{1-\theta}$

  $\rightsquigarrow$ only need to enforce $m \geq 0$
- log odds ratio:

  $$m = \ell(\theta) = \log\left(\frac{\theta}{1-\theta}\right)$$

  $\hookrightarrow \theta = \dfrac{e^m}{1+e^m}, \quad 1-\theta = \dfrac{1}{1+e^m}$

$$\cdots \quad f(x, m) = -\log p(x|\theta) = \underbrace{-x\log\theta - (1-x)\log(1-\theta)}_{\text{cross-entropy}}$$

$$\boxed{f(x, m) = \log(1+e^m) - xm}$$

③ Other distributions

Table 1: Statistically-motivated loss functions. Parameters in blue are assumed to be constant. Numerical adjustments are indicated in red.

Distribution	Link function	Loss function	Constraints
$\mathcal{N}(\mu, \sigma)$	$m = \mu$	$(x-m)^2$	$x, m \in \mathbb{R}$
Gamma(k, σ)	$m = k\sigma$	$x/(m+\epsilon) + \log(m+\epsilon)$	$x > 0, m \geq 0$
Rayleigh(θ)	$m = \sqrt{\pi/2}\,\theta$	$2\log(m+\epsilon) + (\pi/4)(x/(m+\epsilon))^2$	$x > 0, m \geq 0$
Poisson(λ)	$m = \lambda$	$m - x\log(m+\epsilon)$	$x \in \mathbb{N}, m \geq 0$
	$m = \log\lambda$	$e^m - xm$	$x \in \mathbb{N},\ m \in \mathbb{R}$
Bernoulli(ρ)	$m = \rho\,/\,(1-\rho)$	$\log(m+1) - x\log(m+\epsilon)$	$x \in \{0, 1\}, m \geq 0$
	$m = \log(\rho\,/\,(1-\rho))$	$\log(1+e^m) - xm$	$x \in \{0, 1\},\ m \in \mathbb{R}$
NegBinom(r, ρ)	$m = \rho\,/\,(1-\rho)$	$(r+x)\log(1+m) - x\log(m+\epsilon)$	$x \in \mathbb{N},\ m \geq 0$

Gradient based optimization

(if Ω is the set of observed entries:
$W_{i_1 \cdots i_d} = 1$ if $(i_1, \cdots, i_d) \in \Omega$ and 0 o.w.)

$m_1 \times \cdots \times m_d$

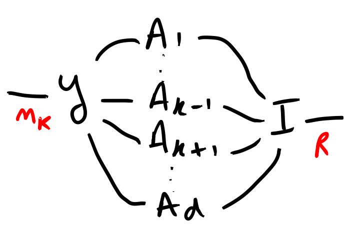
$$\star : \quad \min_{A_1, \cdots, A_d} F(M; X, W) \equiv \sum_{i_1 \cdots i_d} W_{i_1 \cdots i_d}\, f(X_{i_1 \cdots i_d}, M_{i_1 \cdots i_d})$$

$$\text{s.t.} \quad M = [\![\underset{m_1 \times R}{A_1}, \underset{m_2 \times R}{A_2}, \cdots, \underset{m_d \times R}{A_d}]\!]$$

$$\begin{tikzcd}
A_1 \arrow[dr] & X \arrow[dr] & \\
A_2 \arrow[r] & M \arrow[r] & F(M; X, W) \\
\vdots & & \\
A_d \arrow[uur] & W \arrow[uur] &
\end{tikzcd}$$

We want to show

$$\underset{1 \times m_k \times R}{\frac{\partial F(M; X, W)}{\partial A_k}} =$$



where $y \in \mathbb{R}^{m_1 \times \cdots \times m_d}$

$$y_{i_1 \cdots i_d} = W_{i_1 \cdots i_d} \underset{1 \times 1}{\frac{\partial f(X_{i_1 \cdots i_d}, M_{i_1 \cdots i_d})}{\partial M_{i_1 \cdots i_d}}}$$

$\hookrightarrow$ of the same size as X but it is sparse if W is sparse.

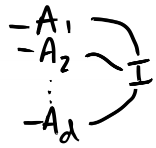
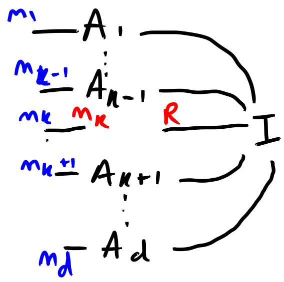
$$= y_{(k)} \left(A_d \odot \cdots \odot A_{k+1} \odot A_{k-1} \odot \cdots \odot A_1 \right)$$

$$\frac{\partial F(M;X,W)}{\partial A_k} = \overbrace{\frac{\partial F(M;X,M)}{\partial M}}^{y} \frac{\partial M}{\partial A_k}$$

$1 \times m_k \times R \qquad 1 \times m_1 \times \cdots \times m_d \qquad m_1 \times \cdots \times m_d \times m_k \times R$

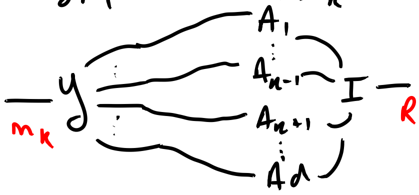
$$\frac{\partial F(M;X,M)}{\partial M}_{i_1,\dots,i_d} = \partial\left(\sum_{j_1\dots j_d} W_{j_1\dots j_d}\, f(X_{j_1\dots j_d}, M_{j_1\dots j_d})\right) \Big/ \partial M_{i_1,\dots,i_d}$$

$$= W_{i_1\dots i_d} \frac{\partial f(X_{i_1\dots i_d}, M_{i_1\dots i_d})}{\partial M_{i_1\dots i_d}} := y_{i_1\dots i_d}$$

$$\frac{\partial M}{\partial A_k} = \partial\left(\right.$$  $$\left.\right) / \partial A_k =$$ 

$m_1 \times \cdots \times m_d \times m_k \times R$

$$\frac{\partial F(M;X,W)}{\partial A_k} = \frac{\partial F(M;X,M)}{\partial M} \frac{\partial M}{\partial A_k}$$

$=$  $= y_{(k)} \left(A_d \odot \cdots \odot A_{k+1} \odot A_{k-1} \odot \cdots \odot A_1\right)$

Follow up paper: **Stochastic Gradients for Large-Scale Tensor Decomposition***

Tamara G. Kolda† and David Hong‡