

Notation

Numerical Methods for Deep Learning

Data

- ▶ n - number of examples
- ▶ n_f - dimension of feature vector
- ▶ n_c - dimension of prediction (e.g., number of classes)
- ▶ $\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n \in \mathbb{R}^{n_f}$ - input features
- ▶ $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n] \in \mathbb{R}^{n_f \times n}$ - feature matrix
- ▶ $\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_n \in \mathbb{R}^{n_c}$ - output observations
- ▶ $\mathbf{C} = [\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_n] \in \mathbb{R}^{n_c \times n}$ - observation matrix

Neural Networks

- ▶ $f(\mathbf{y}, \theta) = \mathbf{c}$ - model represented by neural net
- ▶ $\theta \in \mathbb{R}^{n_p}$ - parameters of model
- ▶ $\theta^{(1)}, \theta^{(2)}, \dots$ - parts of weights. Division clear from context. Examples
 1. $\theta^{(j)}$ are weights of j th layer.
 2. $\theta^{(1)}$ are weights for convolution kernel, $\theta^{(2)}$ are weights for bias
- ▶ N - number of layers
- ▶ \mathbf{K} - linear operator applied to features
- ▶ b - bias
- ▶ $\sigma : \mathbb{R} \rightarrow \mathbb{R}$ - activation function

Optimization and Loss

- ▶ $E(\mathbf{Y}, \mathbf{C}, \mathbf{W})$ - loss function parameterized by weights \mathbf{W}
- ▶ $\phi : \mathbb{R}^k \rightarrow \mathbb{R}$ - generic objective function
- ▶ θ^* - minimizer of a function, i.e.,

$$\theta^* = \arg \min_{\theta} \phi(\theta)$$

- ▶ $\theta_1, \theta_2, \dots$ - iterates
- ▶ \mathbf{d}, \mathbf{D} - search directions
- ▶ α - step size
- ▶ λ - regularization parameter
- ▶ $\nabla_{\mathbf{x}} F$ - gradient, if $F : \mathbb{R}^k \rightarrow \mathbb{R}^l$, then $\nabla F(\mathbf{x}) \in \mathbb{R}^{k \times l}$
- ▶ $\mathbf{J}_{\mathbf{x}} F$ - Jacobian of F with respect to \mathbf{x} , $\mathbf{J}_{\mathbf{x}} F = (\nabla_{\mathbf{x}} F)^\top$

Linear Algebra - 1

- ▶ $\mathbf{e}_k \in \mathbb{R}^k$ - vector of all ones
- ▶ \mathbf{I}_k - $k \times k$ identity matrix
- ▶ $\kappa(\mathbf{A})$ - condition number of \mathbf{A}
- ▶ $\sigma_1(\mathbf{A}) \geq \dots \geq \sigma_k(\mathbf{A}) \geq 0$ - singular values of \mathbf{A}
- ▶ $\lambda_1(\mathbf{A}), \dots$ - eigenvalues of \mathbf{A}
- ▶ $\text{tr}(\mathbf{A})$ - trace of square matrix, i.e., sum of diagonal elements

Linear Algebra - 2

- ▶ \odot - Hadamard product

$$\mathbf{C}_{ij} = \mathbf{A}_{ij} \cdot \mathbf{B}_{ij}, \quad \text{for } \mathbf{B}, \mathbf{A} \in \mathbb{R}^{k \times l}$$

MATLAB: $\mathbf{C} = \mathbf{A}.*\mathbf{B}$

- ▶ \otimes - Kronecker product

$$\mathbf{A} \otimes \mathbf{B} = \begin{pmatrix} \mathbf{A}_{11}\mathbf{B} & \mathbf{A}_{12}\mathbf{B} & \dots & \mathbf{A}_{1l}\mathbf{B} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{A}_{k1}\mathbf{B} & \mathbf{A}_{k2}\mathbf{B} & \dots & \mathbf{A}_{kl}\mathbf{B} \end{pmatrix}$$

MATLAB: $\mathbf{C} = \text{kron}(\mathbf{A}, \mathbf{B})$

- ▶ $\text{vec}(\mathbf{A})$ - reshape matrix \mathbf{A} into vector (column-wise).

$$\text{Example: } \text{vec} \left(\begin{pmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{pmatrix} \right) = \begin{pmatrix} \mathbf{A}_{11} \\ \mathbf{A}_{21} \\ \mathbf{A}_{12} \\ \mathbf{A}_{22} \end{pmatrix}$$

MATLAB: $\mathbf{a} = \mathbf{A}(:)$

Linear Algebra - 3

- ▶ $\text{mat}(\mathbf{v}, k, l)$ - reshape vector $\mathbf{v} \in \mathbb{R}^{kl}$ into matrix. k, l omitted when dimension clear from context. Note

$$\text{mat}(\text{vec}(\mathbf{A})) = \mathbf{A}.$$

MATLAB: $V = \text{reshape}(v, k, l)$.

- ▶ $\text{diag}(\mathbf{v})$ - diagonal matrix with elements of $\mathbf{v} \in \mathbb{R}^k$ on diagonal

MATLAB: $V = \text{diag}(v(:))$

- ▶ $\text{diag}(\mathbf{A})$ - diagonal matrix obtained by vectorizing \mathbf{A}

Acronyms

- ▶ CG - Conjugate Gradient Method
- ▶ VarPro - Variable Projection
- ▶ SD - Steepest Descent
- ▶ SGD - Stochastic Gradient Descent
- ▶ SA - Stochastic Approximation
- ▶ SAA - Stochastic Average Approximation
- ▶ SPD - symmetric positive definite
- ▶ SPSD - symmetric positive semi-definite
- ▶ CV - Cross Validation