18.S096 PSET 1

IAP 2023

Problem 0 (4+4+4+4 points)

The hyperbolic Corgi notebook may be found at https://mit-c25.netlify.app/notebooks/1_hyperbolic_corgi. Compute the 2×2 Jacobian matrix for each of the following image transformations from that notebook:

(a) rotate(θ): $(x, y) \to (\cos(\theta)x + \sin(\theta)y, -\sin(\theta)x + \cos(\theta)y)$

(b) hyperbolic_rotate(θ): $(x, y) \to (\cosh(\theta)x + \sinh(\theta)y, \sinh(\theta)x + \cosh(\theta)y)$

(c) nonlin_shear(θ): $(x, y) \to (x, y + \theta x^2)$

(d) warp(θ): $(x, y) \to \text{rotate}(\theta\sqrt{x^2 + y^2})(x, y)$

Problem 1 (5+4 points)

(a) Suppose that $L[x]$ is a linear operation (for x in some vector space V, with outputs $L[x]$ in some other vector space W). If $f(x) = L[x] + y$ for a constant $y \in W$, what is $f'(x)$ (in terms of L and/or y)?

(b) Give the derivatives of $f(A) = A^T$ (transpose) and $g(A) = 1 + \operatorname{tr} A$ (trace) as special cases of the rule you derived in the previous part.

Problem 2 (5+6+5+5 points)

Calculate derivatives of each of the following functions in the requested forms—as a linear operator $f'(x)[dx]$, a Jacobian matrix, or a gradient ∇f —as specified in each part.

(a) $f(x) = x^T(A + \text{diagm}(x))^2 x$, where the inputs $x \in \mathbb{R}^n$ are vectors, the outputs are scalars, $A = A^T$ is a constant *symmetric* $n \times n$ matrix $\in \mathbb{R}^{n \times n}$, and diagm($x$) denotes the $n \times n$ diagonal matrix $\begin{pmatrix} x_1 & & \\ & x_2 & \\ & & \ddots \end{pmatrix}$.

Give the **gradient** ∇f, such that $f'(x)dx = (\nabla f)^T dx$.

(b) $f(x) = (A + yx^T)^{-1}b$, where the inputs x and outputs $f(x)$ are n-component (column) vectors in $\mathbb{R}^n$, y and b are constant vectors $\in \mathbb{R}^n$, and A is a constant $n \times n$ matrix $\in \mathbb{R}^{n \times n}$.

(i) Give $f'(x)$ as a **Jacobian** matrix.

(ii) If you are given A^{-1}, then you can compute $(A + yx^T)^{-1}$ and hence $f(x)$ for any x in $\sim n^2$ scalar-arithmetic operations (i.e., roughly proportional to n^2, or in computer-science terms $\Theta(n^2)$ "complexity"), using the "Sherman–Morrison" formula (Google it). **Explain** how your Jacobian matrix can therefore also be computed in $\sim n^2$ operations for any x given A^{-1} (i.e. give a sequence of computational steps, each of which costs no more than $\sim n^2$ arithmetic).

(c) $f(x) = \frac{xx^T}{x^Tx}$, with vector inputs $x \in \mathbb{R}^n$ and matrix outputs $f \in \mathbb{R}^{n \times n}$. Give $f'(x)$ as a linear operator, i.e. a linear formula for $f'(x)[dx]$.

(d) $g(x) = \frac{xx^T}{x^Tx}b$, with vector inputs $x \in \mathbb{R}^n$ and vector outputs $f \in \mathbb{R}^n$, where $b \in \mathbb{R}^n$ is a constant vector. Give $g'(x)$ as a **Jacobian** matrix.

Problem 3 (5+5+5 points)

(a) Argue briefly that linear functions that map $n \times n$ matrices to $n \times n$ matrices themselves form a vector space V. What is the dimension of this vector space?

(b) Argue briefly that linear functions of $n \times n$ matrices of the form $X \to AX$, where A is $n \times n$, form a vector space. What is the dimension of this vector space?

(c) Argue briefly why it follows that there must be infinitely many linear functions $\in V$ that are not of the form $X \to AX$.