

---

# Correlation of the Jacobian and Probability Flow

---

**John Y. Shin**

Department of Computer Science  
New York University  
New York, NY 10033  
jys308@nyu.edu

## Abstract

### 1 Computing the Jacobian

Suppose we have a Neural Network classifier  $f : \mathbb{R}^d \rightarrow \mathbb{R}^k$ , where  $d$  is the dimensionality of the input and  $k$  is the number of classes. We define the Jacobian matrix of the Neural Network as:

$$J(\mathbf{x}) = \nabla_{\mathbf{x}} f(\mathbf{x}) \in \mathbb{R}^k \times \mathbb{R}^d \quad (1)$$

Or the derivative of the Neural Network with respect to the input. Suppose we have a training dataset of size  $n$ , with examples given as  $(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n)$ , where  $\mathbf{x}_i \in \mathbb{R}^d$ . We can concatenate the  $J(\mathbf{x})$  for each example into a vector:

$$\mathcal{J}_i = \text{flatten}(J(\mathbf{x}_i)) \in \mathbb{R}^{k \times d} \quad (2)$$

We can then form a matrix of these  $\mathcal{J}_i$ 's into a matrix  $\mathcal{J} \in \mathbb{R}^n \times \mathbb{R}^{k \times d}$ , which we will call the Jacobian matrix of the training dataset. Each row corresponds to a training example and the columns are a concatenation of the derivatives of each dimension of the output with each dimension of the input.

As a classifier trained with a cross-entropy loss and with a softmax as the final output of the network, we can interpret the output of the network as a vector of probabilities:

$$f(\mathbf{x}) = (p(y_1|\mathbf{x}), p(y_2|\mathbf{x}), \dots, p(y_n|\mathbf{x})) \in \mathbb{R}^k \quad (3)$$

The Jacobian matrix of the neural network then becomes:

$$J(\mathbf{x}) = \nabla_{\mathbf{x}} f(\mathbf{x}) = (\nabla_{\mathbf{x}} p(y_1|\mathbf{x}), \nabla_{\mathbf{x}} p(y_2|\mathbf{x}), \dots, \nabla_{\mathbf{x}} p(y_k|\mathbf{x})) \in \mathbb{R}^k \times \mathbb{R}^d \quad (4)$$

For conciseness (and abuse) of notation, we can write the flattened  $\mathcal{J}_i$  in vector notation as:

$$\mathcal{J}_i = \nabla_{\mathbf{x}_i} p(\mathbf{y}|\mathbf{x}_i) \in \mathbb{R}^{k \times d} \quad (5)$$

We can then build the correlation matrix of the Jacobian over the training dataset as:

$$M_{ij} = (\mathcal{J} \mathcal{J}^T)_{ij} = \nabla_{\mathbf{x}_i} p(\mathbf{y}|\mathbf{x}_i) \cdot \nabla_{\mathbf{x}_j} p(\mathbf{y}|\mathbf{x}_j) \in \mathbb{R}^{n \times n} \quad (6)$$

## References