tab2img

A tool to convert tabular data into images for CNN. Inspired by the [DeepInsight]() paper.

Installation

```
pip install tab2img
```

Background

In the [paper]() “*DeepInsight: A methodology to transform a non-image data to an image for convolution neural network architecture*” the autors propose a method to convert tabular data into images, in order to utilize the power of convolutional neural network (CNN) for non-image structured data.

Features to image mapping

The Figure illustrates the main idea: given a training dataset $X \in \mathbb{R}^{m \times n}$, with m samples and n features, we are required to find a function $M : \mathbb{R}^{m \times n} \to \mathbb{R}^{m \times d \times d}$, where $d = \lceil \sqrt{n} \rceil$.
There are numerous ways to choose M. In this implementation, the features are organized with respect to the correlation vector $\rho(X, Y)$, where $Y \in \mathbb{R}^{1 \times m}$ is the target vector. Given X and Y as

$$X = \begin{pmatrix} x_1^{(1)} & \cdots & x_n^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(m)} & \cdots & x_n^{(m)} \end{pmatrix}, \quad Y = \begin{pmatrix} y_1 \\ \vdots \\ y_m \end{pmatrix},$$

the vector $\rho(X, Y) = (\rho_1, \dots, \rho_n)$ express the Pearson correlation coefficient [1]

$$\rho(x, y) = \frac{\operatorname{cov}(x, y)}{\sigma(x)\sigma(y)},$$

where

$$\rho_i = \rho(X_i, Y), \quad X_i = \begin{pmatrix} x_i^{(1)} \\ \vdots \\ x_i^{(m)} \end{pmatrix}.$$

At this point $\rho(X, Y)$ is sorted from the greatest to the smallest, generating the vector of indices $\mathbf{J} = (J_k \in \mathbb{N} : \rho_{J_k} \geq \rho_{J_{k-1}},\ k \in [1, ..., n])$.
Eventually, the final tensor M is

$$M = \begin{pmatrix} X_{J_1} & X_{J_2} & X_{J_{10}} & \cdots \\ X_{J_3} & X_{J_4} & X_{J_7} & \cdots \\ X_{J_6} & X_{J_8} & X_{J_9} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}.$$

The function that maps k (J_k) to the right row and column (r, c) of M is

$$(r,c)_k = \begin{cases} (\sqrt{k}, \sqrt{k}) & \text{if } \sqrt{k} \in \mathbb{N} \\ (\lceil\sqrt{k}\rceil, \lceil\sqrt{k}\rceil - \frac{1}{2}(\lceil\sqrt{k}\rceil^2 - k)) & \text{if } \sqrt{k} \notin \mathbb{N} \text{ and } \lceil\sqrt{k}\rceil^2 - k = 0 \mod 2 \\ (\lceil\sqrt{k}\rceil - \lceil\frac{1}{2}(\lceil\sqrt{k}\rceil^2 - k)\rceil, \lceil\sqrt{k}\rceil) & \text{if } \sqrt{k} \notin \mathbb{N} \text{ and } \lceil\sqrt{k}\rceil^2 - k \neq 0 \mod 2 \end{cases}$$

---

1. In this case, being X a sample, the coefficient is implemented as

$$\rho(x, y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

. ↩