



# MNIST漫谈

2017-2018学年 系统仿真 结课答辩

主讲人 高梓恒 15021202

组长 张奥 15021120

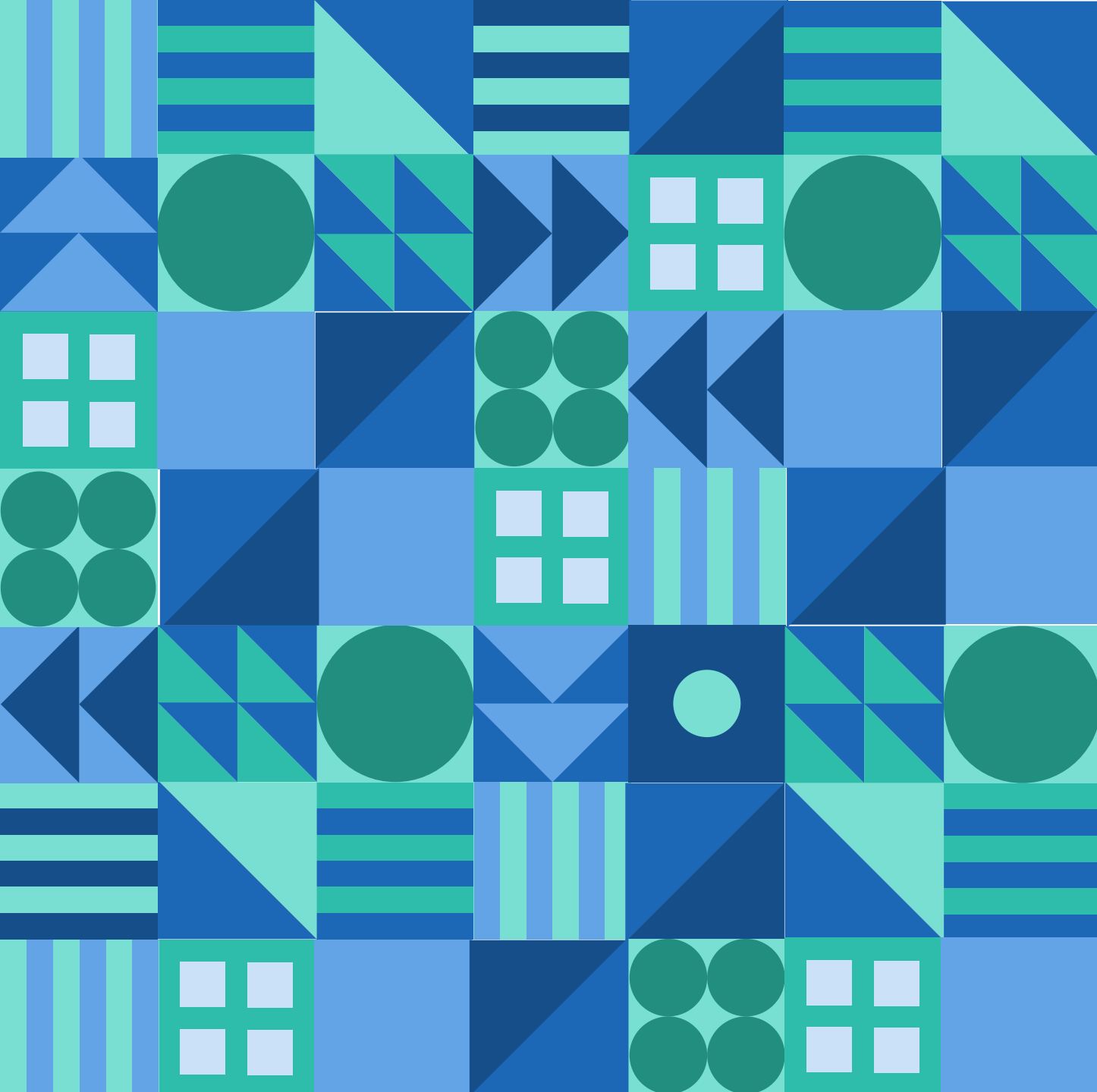


天津科技大学  
Tianjin University of Science & Technology



# CONTENTS

- 01** MNIST 简介
- 02** 识别数字分析
- 03** Matlab 实现
- 04** Google Tensorflow 实现
- 05** 延伸参考



# 01

## MNIST 简介

## 01 MNIST 简介

# MNIST

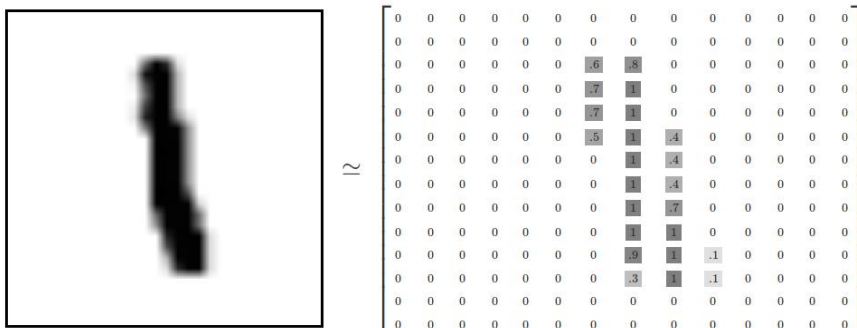
## Modified National Institute of Standards and Technology database of handwritten digits

当我们开始学习编程的时候，第一件事往往是学习打印"Hello World"。就好比编程入门有Hello World，机器学习入门有MNIST。MNIST是一个入门级的计算机视觉数据集，它包含各种手写数字图片；

它也包含每一张图片对应的标签，下载下来的数据集被分成两部分：60000行的训练数据集（mnist.train）和10000行的测试数据集（mnist.test）。这样的切分很重要，在机器学习模型设计时必须有一个单独的测试数据集不用于训练而是用来评估这个模型的性能，从而更加容易把设计的模型推广到其他数据集上（泛化）。

注：**NIST**=美国国家标准技术研究所

数据集官网 <http://yann.lecun.com/exdb/mnist/>



## Yann LeCun, Professor

## The Courant Institute of Mathematical Sciences

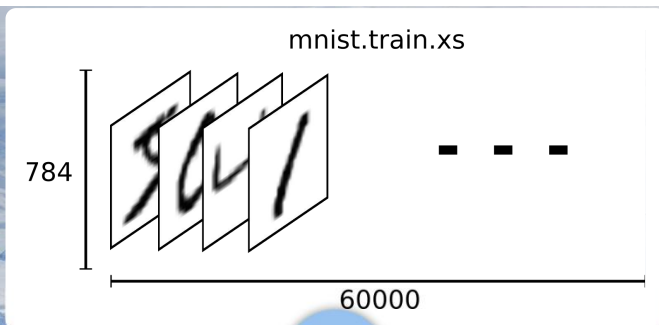
# New York University

### Corinna Cortes, Research Scientist

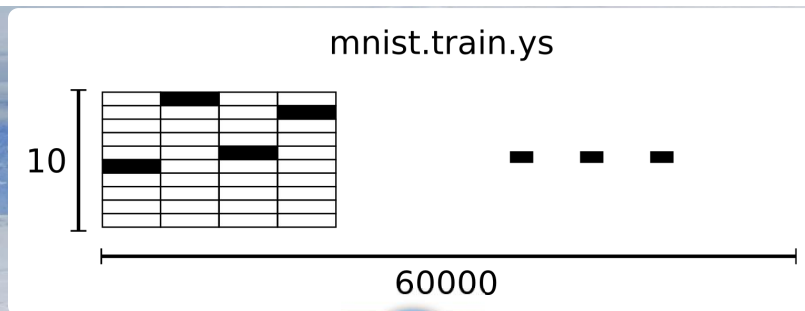
## Google Labs, New York

**corinna at google dot com**

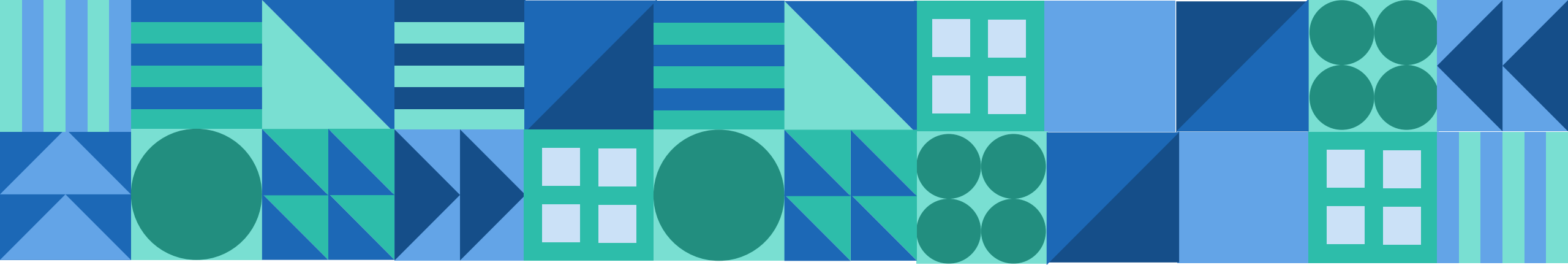
# 01 MNIST 简介



每一张图片包含28像素X28像素。我们可以用一个数字数组来表示这张图片，我们把这个数组展开成一个向量，长度是  $28 \times 28 = 784$ 。只要保持各个图片采用相同的方式展开。从这个角度来看，MNIST数据集的图片就是在784维向量空间里面的点，并且拥有比较复杂的结构。因此，在MNIST训练数据集中，`mnist.train.images` 是一个形状为 `[60000, 784]` 的张量。

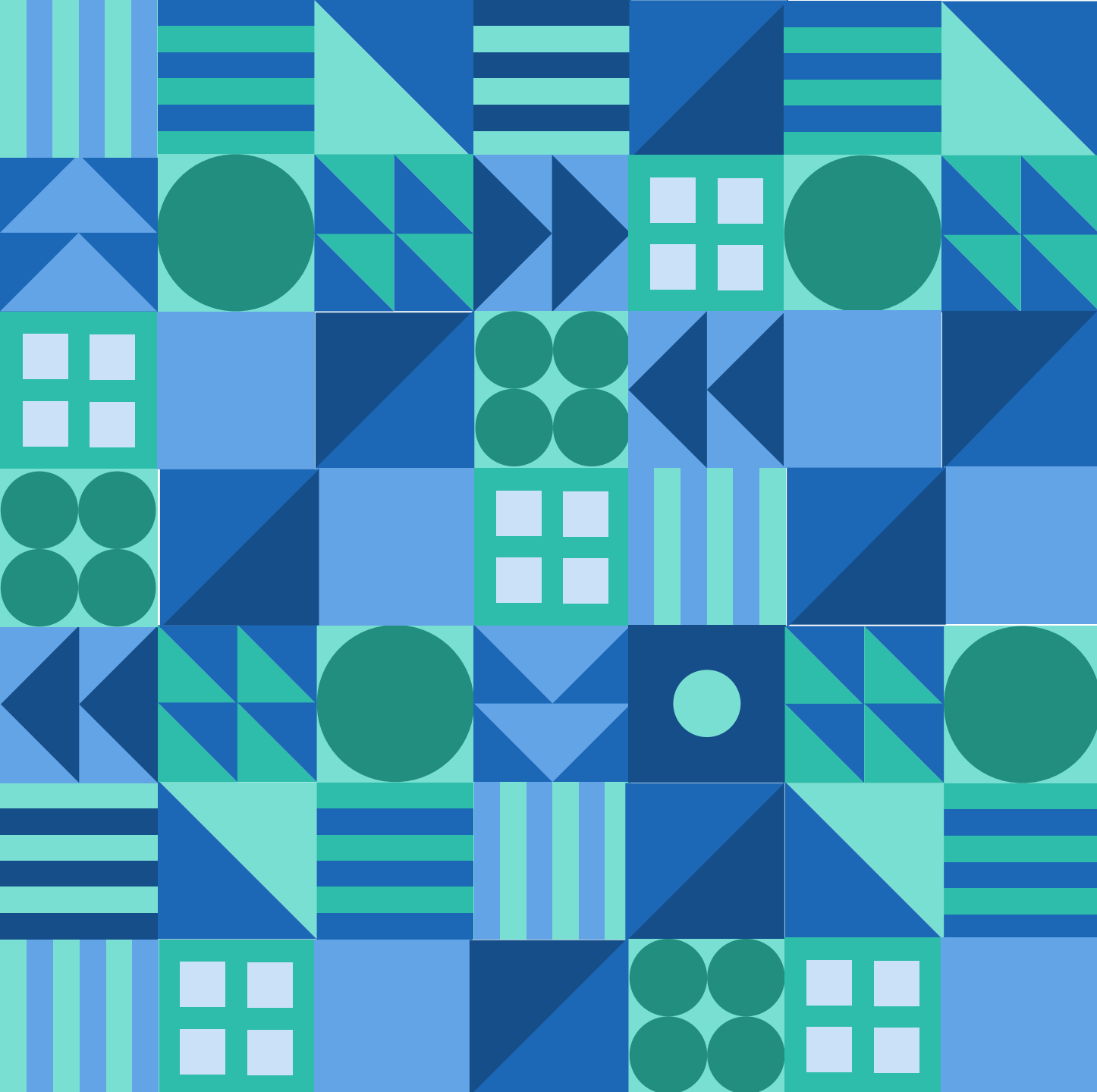


相对应的MNIST数据集的标签是介于0到9的数字，用来描述给定图片里表示的数字。一个one-hot向量除了某一位的数字是1以外其余各维度数字都是0。数字n将表示成一个只有在第n维度（从0开始）数字为1的10维向量。如：标签0将表示成 `([1,0,0,0,0,0,0,0,0,0])`。因此，`mnist.train.labels` 是 `[60000, 10]` 数字矩阵。



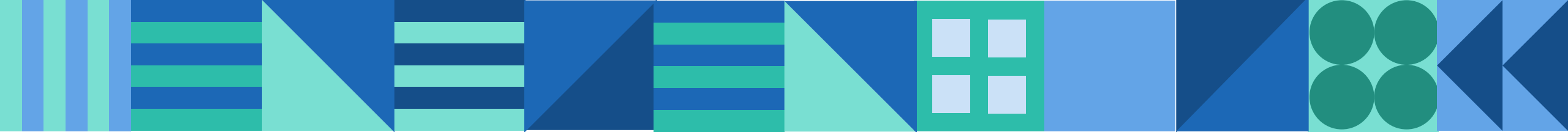
先上个例子





# 02

## 识别数字分析



## 02 识别数字分析

### Softmax 回归 ( softmax regression )

我们知道MNIST的每一张图片都表示一个数字，从0到9。我们希望得到给定图片代表每个数字的概率。比如说，我们的模型可能推测一张包含9的图片代表数字9的概率是80%但是判断它是8的概率是5%（因为8和9都有上半部分的小圆），然后给予它代表其他数字的概率更小的值。

这是一个使用softmax回归 ( softmax regression ) 模型的经典案例。softmax模型可以用来给不同的对象分配概率。即使在之后，我们训练更加精细的模型时，最后一步也需要用softmax来分配概率。

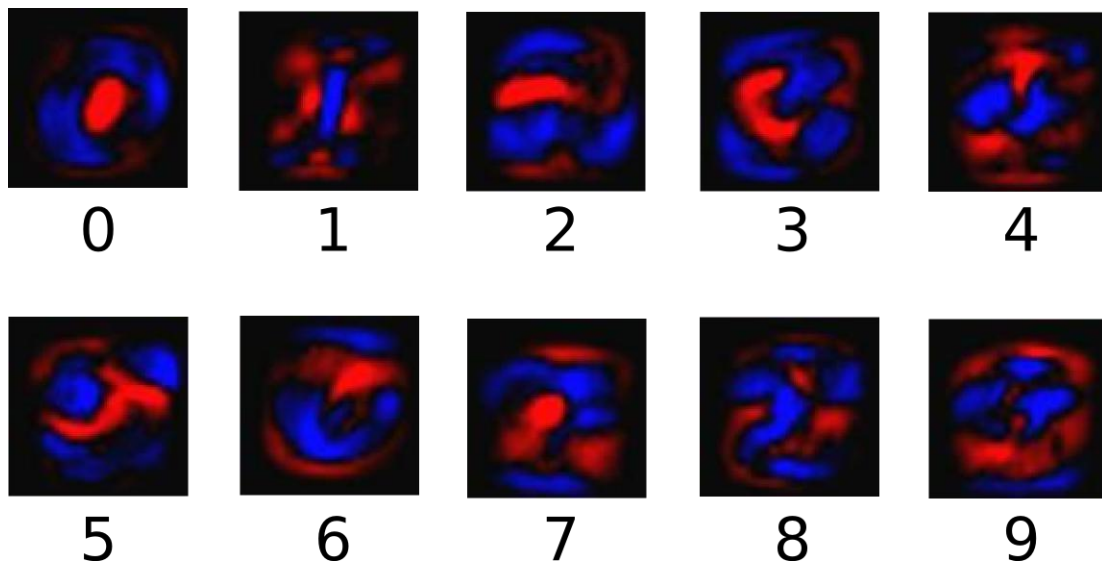




## softmax 回归 第一步

为了得到一张给定图片属于某个特定数字类的证据（**evidence**），我们对图片像素值进行加权求和。如果这个像素具有很强的证据说明这张图片不属于该类，那么相应的权值为负数，相反如果这个像素拥有有利的证据支持这张图片属于这个类，那么权值是正数。

下面的图片显示了一个模型学习到的图片上每个像素对于特定数字类的权值。红色代表负数权值，蓝色代表正数权值。



$$\text{evidence}_i = \sum_j W_{i,j} x_j + b_i$$

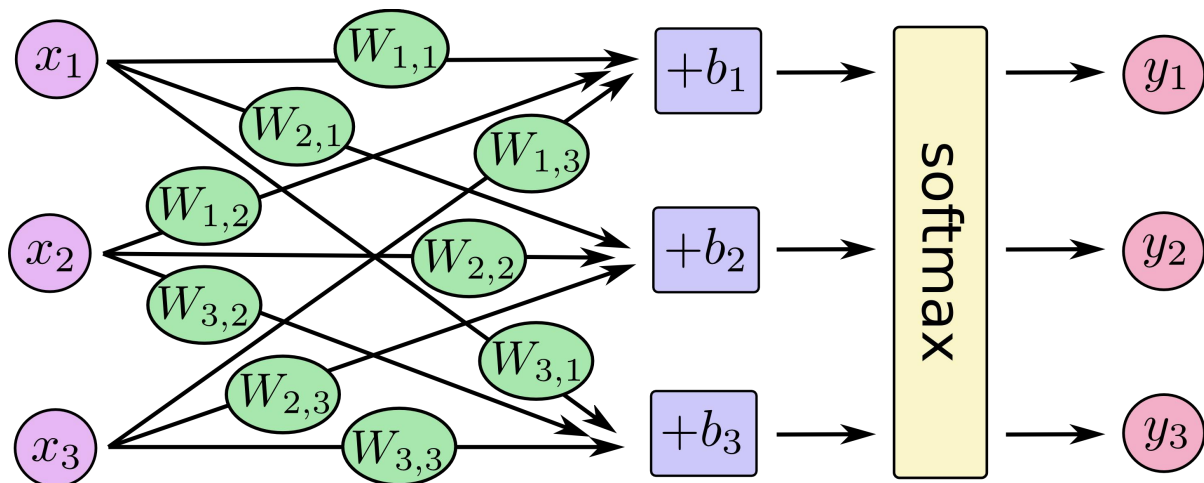
$$y = \text{softmax}(\text{evidence})$$

$$\text{softmax}(x) = \text{normalize}(\exp(x))$$

$$\text{softmax}(x)_i = \frac{\exp(x_i)}{\sum_j \exp(x_j)}$$

## softmax 回归 第二步

$$\text{softmax}(x)_i = \frac{\exp(x_i)}{\sum_j \exp(x_j)}$$



$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \text{softmax} \left( \begin{bmatrix} W_{1,1} & W_{1,2} & W_{1,3} \\ W_{2,1} & W_{2,2} & W_{2,3} \\ W_{3,1} & W_{3,2} & W_{3,3} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \right)$$

对于**softmax**回归模型可以用图解释，对于输入的**x**s加权求和，再分别加上一个偏置量，最后再输入到**softmax**函数中

**softmax**可以看成是一个激励（**activation**）函数或者链接（**link**）函数，把我们定义的线性函数的输出转换成我们想要的格式，也就是关于**10**个数字类的概率分布。因此，给定一张图片，它对于每一个数字的吻合度可以被**softmax**函数转换成为一个概率值。



## 02 识别数字分析

### 训练模型

#### 成本函数(cost)

为了训练我们的模型，首先需要定义一个指标来评估这个模型是好的。

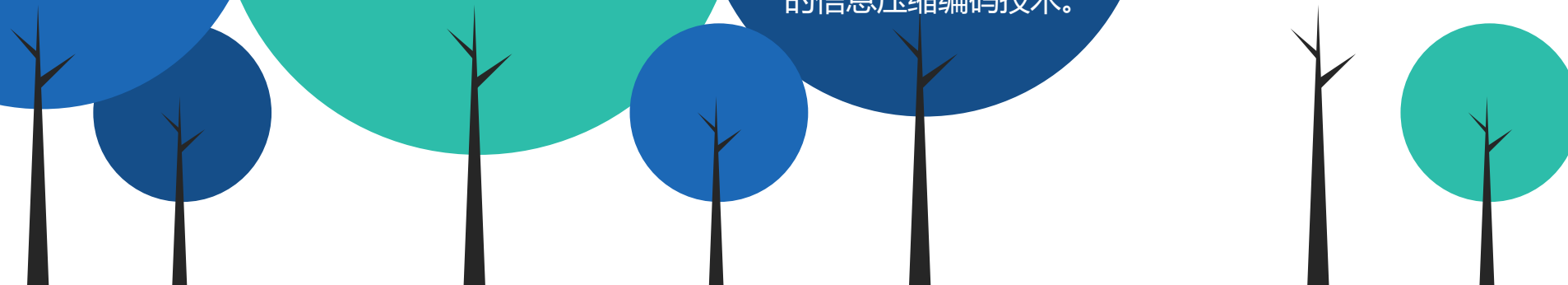
#### 损失函数(loss)

在机器学习，我们通常会定义指标来表示一个模型是坏的。

#### 交叉熵

如右图，一个非常常见的，非常漂亮的成本函数，产生于信息论里面的信息压缩编码技术。

$$H_{y'}(y) = - \sum_i y'_i \log(y_i)$$





## 02 识别数字分析

第一层级

Softmax 回归分析

第二层级

反向传播算法

交叉熵（成本函数）

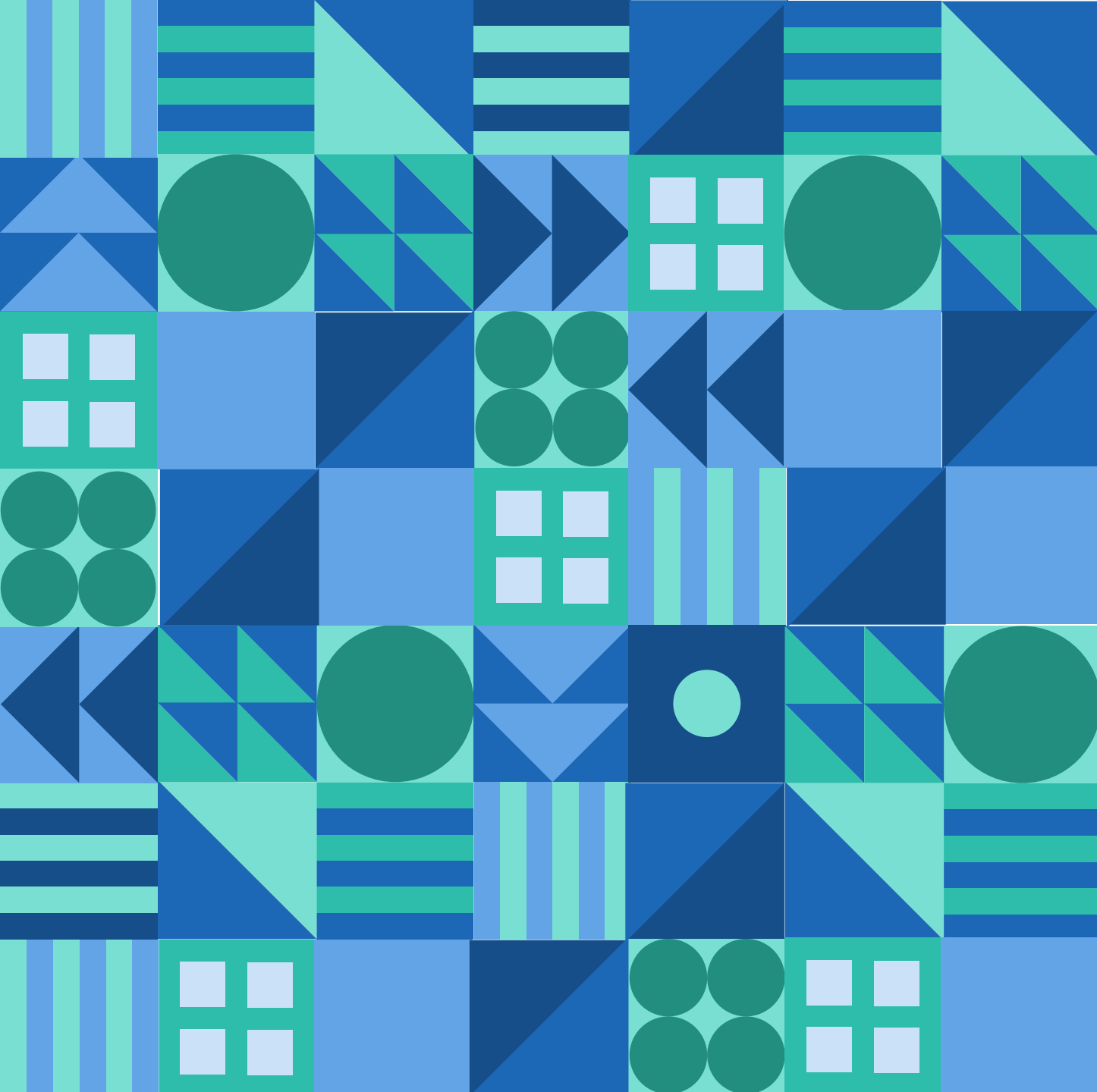
第三层级

多层卷积网络  
( CNN )

聚类分析

循环卷积网络  
( RNN )

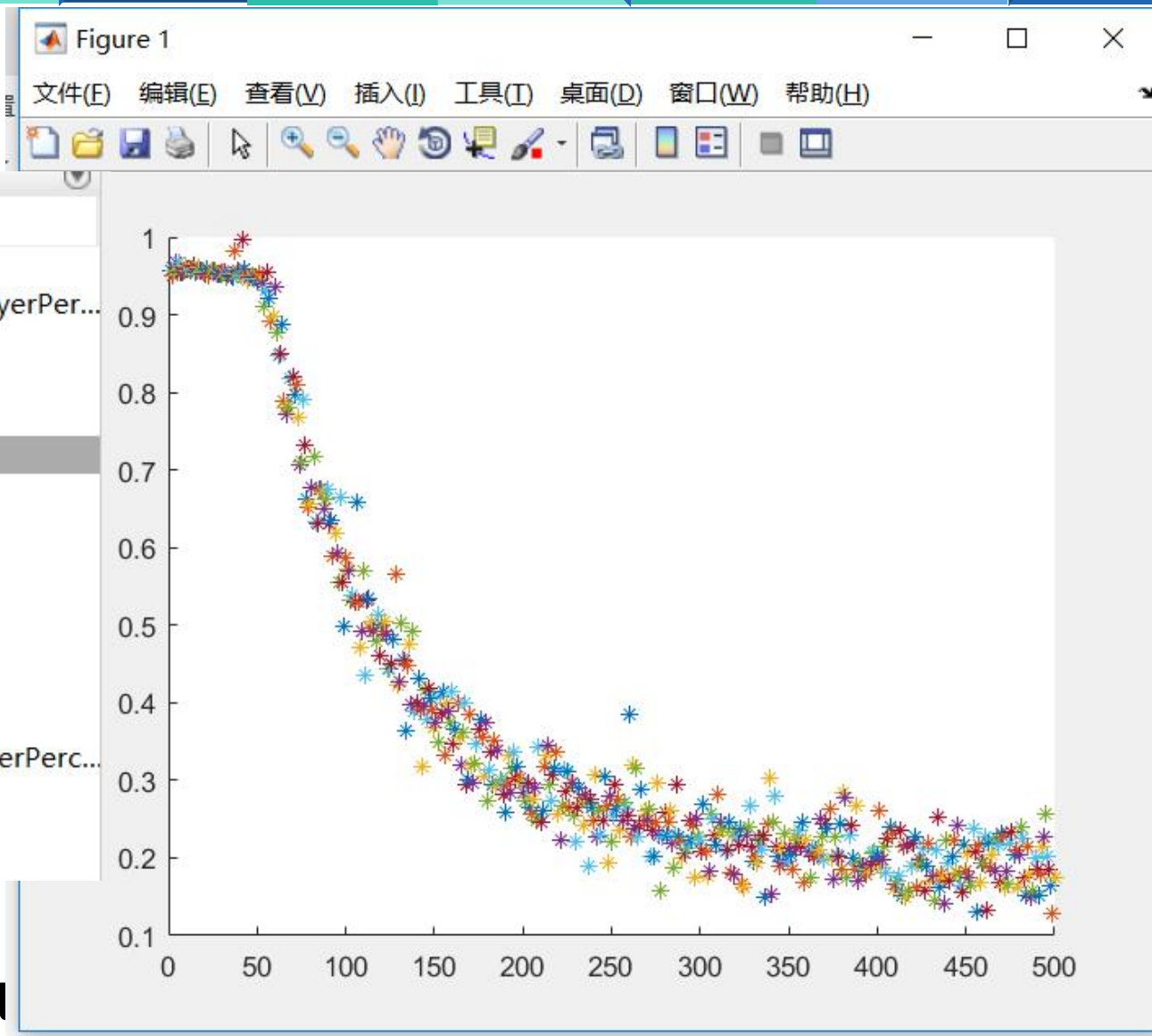
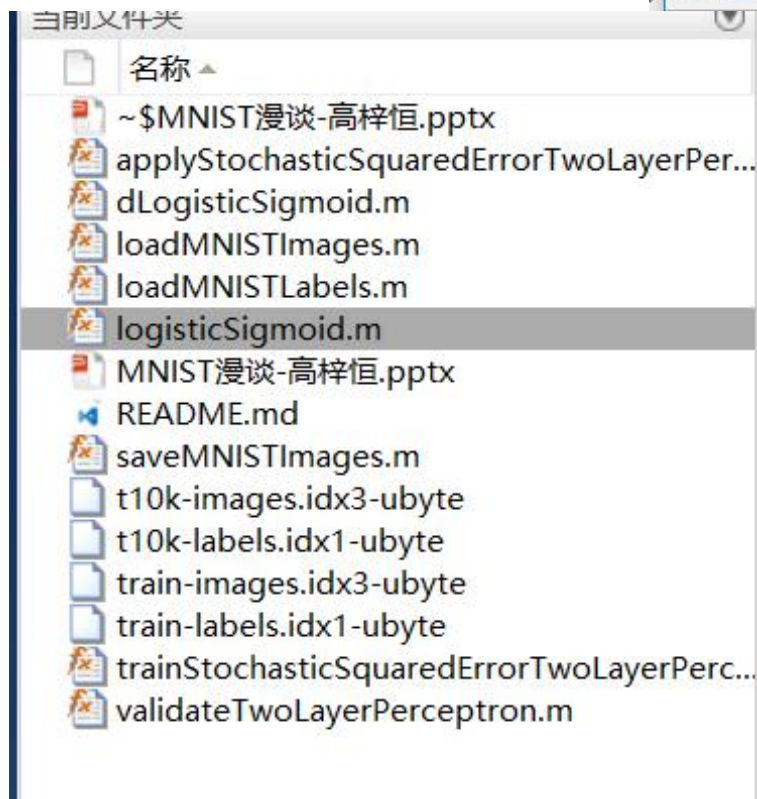
长短时记忆模型  
( LSTM )



# 03

## Matlab实现

## 03 Matlab实现



代码：<https://github.com>

NIST

eptron

## 03 Matlab实现

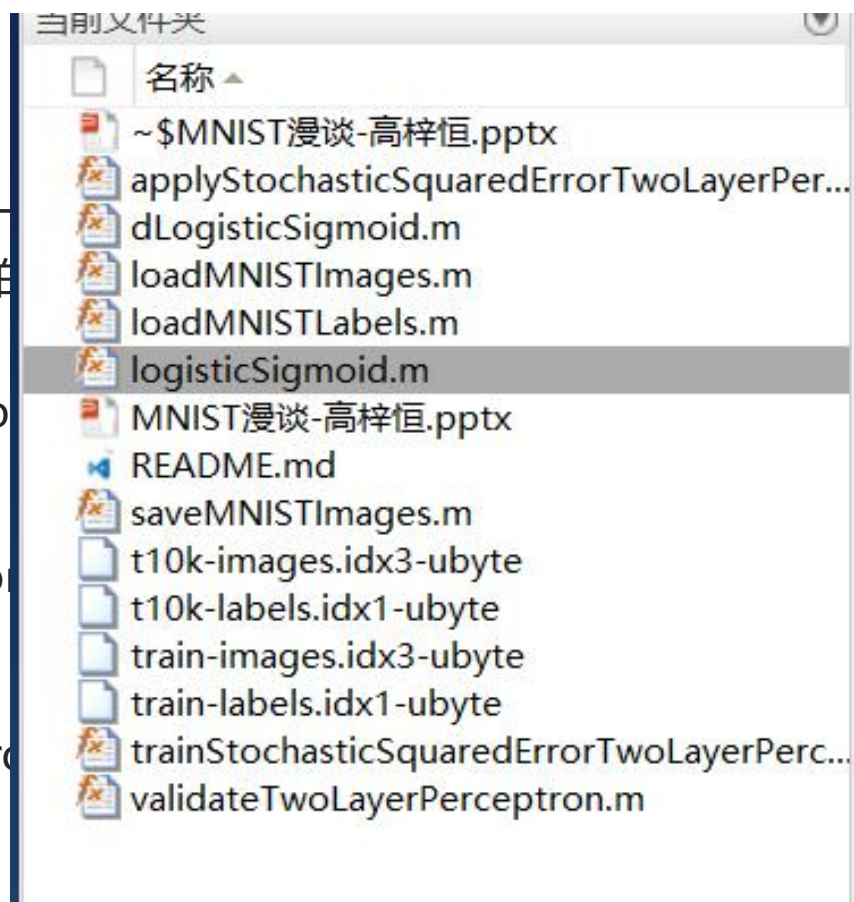


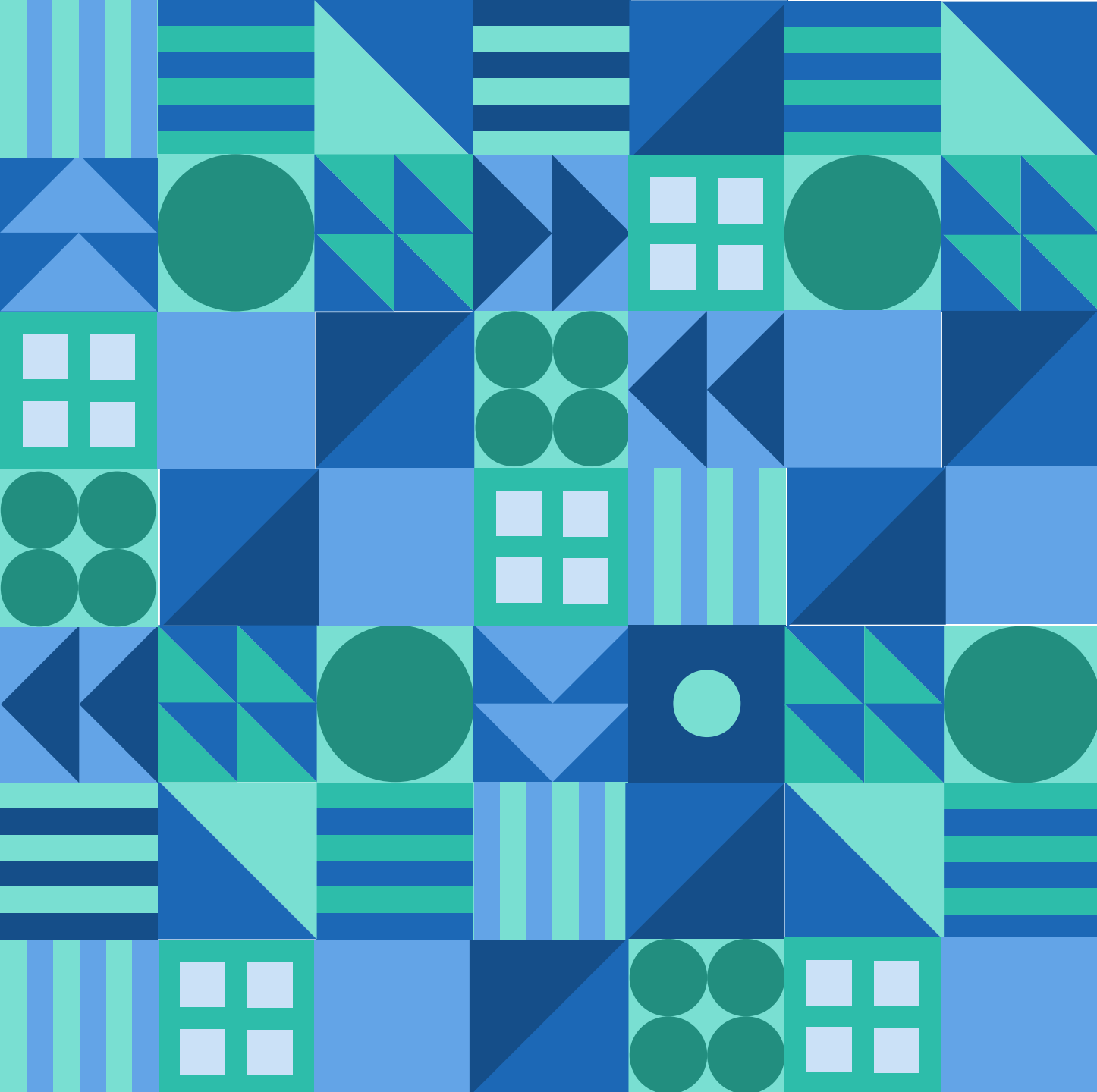
logisticSigmoid 逻辑函数 —  
dLogisticSigmoid 逻辑函数的

trainStochasticSquaredError

validateTwoLayerPerceptron

applyStochasticSquaredError





# 04

**Google® Tensorflow™**  
**实现**



## 04 Google® Tensorflow™ 实现

### 微信示例

一开始展示的微信示例是使用  
Tensorflow、Flask、HTML5 Canvas、  
WeUI等技术完成的。

### 自动的反向传播优化算法

一个字：准！



### 利用图(Graph)构建网络

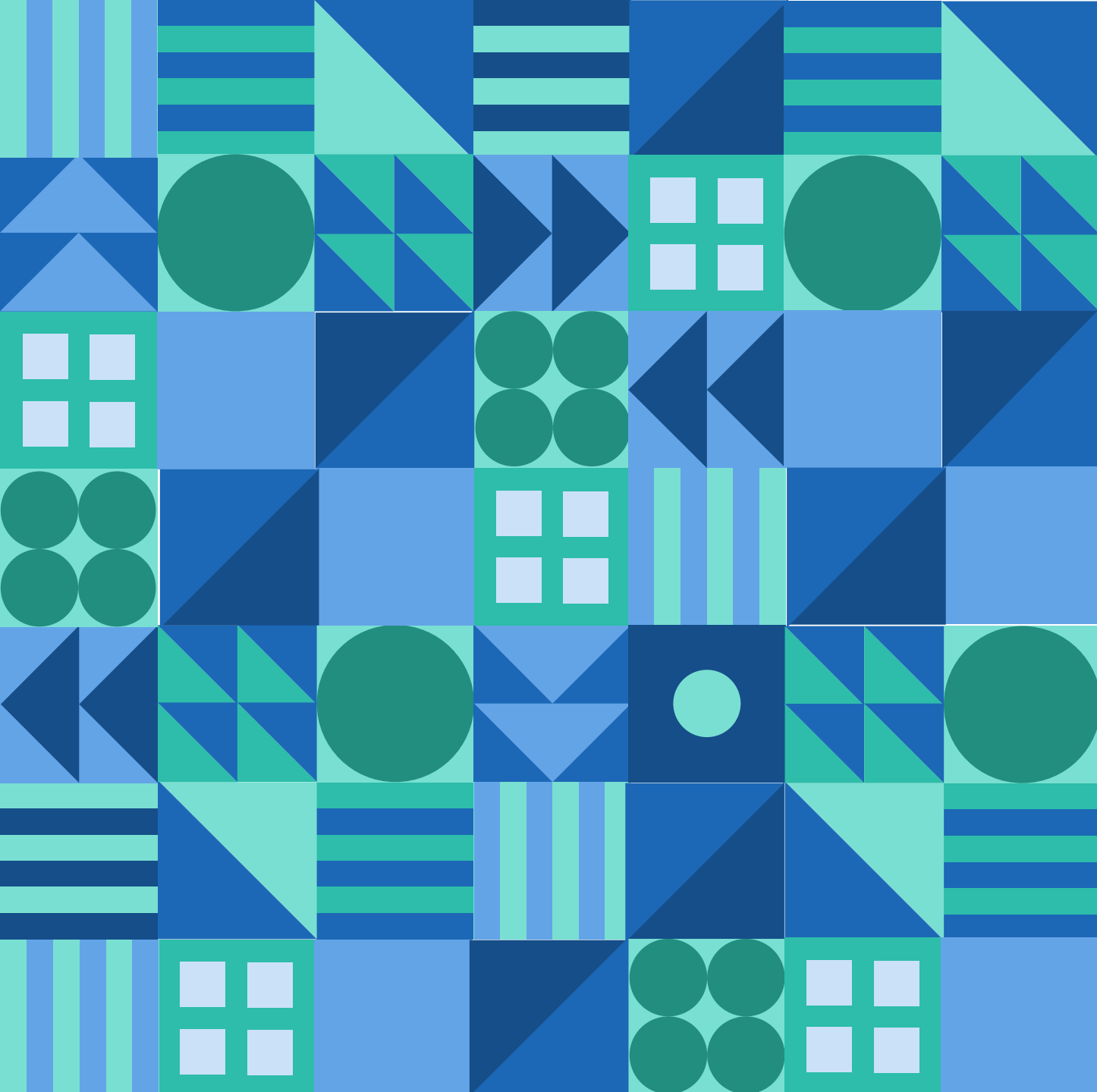
一个字：快！

### 一个简单的视频（2:17）

如时间允许将放映，  
不允许则请大家自行观看

[https://youtu.be/oZikw5k\\_2FM](https://youtu.be/oZikw5k_2FM)

代码托管地址：<https://github.com/noahziheng/mnist-wechat-demo>



# 05

延伸参考



## 05 延伸参考 现代人工智能结构

Google® Alpha Go (Zero)   Alibaba® Taobao   Twitter® Tweet

上层应用

Restful API   泛前端体系 ( Web、iOS、Android )

用户界面

Keras   Caffe   Matconv(剑桥)   私有框架

中层封装

Google® Tensorflow™   Theano   Matlab For Server

底层智能框架



## 05 延伸参考资料来源

**Seminar Paper “Introduction to Neural Networks” By David Stutz**

[//davidstutz.de/seminar-paper-introduction-neural-networks/](http://davidstutz.de/seminar-paper-introduction-neural-networks/)

**Tensorflow™ 官方网站：** [//tensorflow.org](http://tensorflow.org)

**Tensorflow™ 中文社区：** [//tensorflow.cn](http://tensorflow.cn)

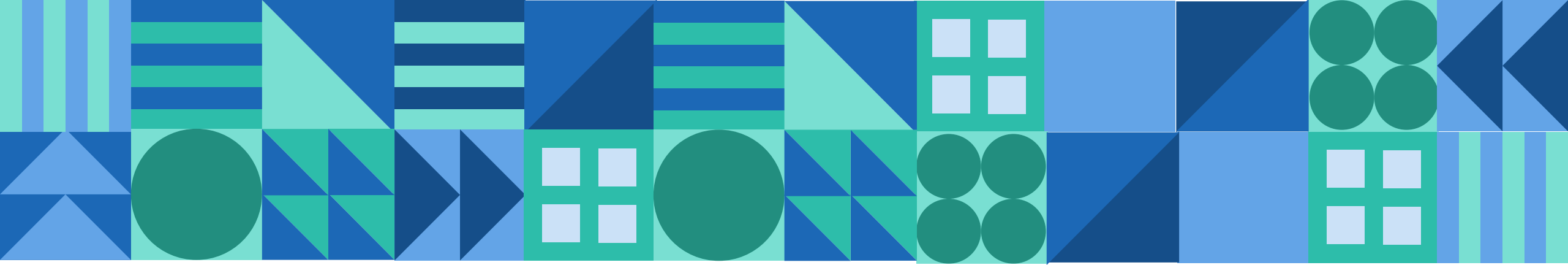
**人工智能与机器学习（南开大学 刘杰博士、教授） From GDG DevFest 2017 Tianjin**

[//github.com/GDGTianjin/DevFest-2017/blob/master/人工智能与机器学习-刘杰.pdf](https://github.com/GDGTianjin/DevFest-2017/blob/master/人工智能与机器学习-刘杰.pdf)

**Google Developers China：** [//developers.google.cn](http://developers.google.cn)

**PPT 下载地址**

<https://github.com/noahziheng/mnist-wechat-demo/tree/master/ppt>



# 组内分工

**张奥 组长**

**高梓恒 演讲 示例编程**

**邱阳 陈国宝 PPT 资料收集**

**于健 张旺 PPT 规划制作**

**刘孟竹 苗玉 MATLAB 编程及仿真**



# 谢谢

水平有限，如有错误，敬请指出

CREATED BY Noah Gao



天津科技大学  
Tianjin University of Science & Technology



OfficePLUS.cn