周记（四）

19200210 孙禾嘉

经过前几阶段的工作，我们成功建立了能够复现人类反应时特点的FSNet，这一阶段的工作中，主要关注于对模型的具体结构进行分析。

我们将主要使用PCA方法，在机器学习的研究中，研究者可以通过分析网络内部参数的时间和空间分布来研究网络的内部结构。主成分分析法(principal component analysis, PCA)是一种在机器学习和数据分析中广泛使用的降维方法。PCA指的是通过将数据点投影到一个超平面，并选取数个主成分，要求这些主成分能够解释尽可能大的方差并互不相关，使数据点在超平面上的投影尽可能分开的一种特征提取算法(刘敬浩, 孙晓伟, 金杰, 2020)。PCA能够提取影响数据分布形态的最主要特征，并且在尽可能保留原始信息的前提下，使数据精简，便于分析和研究。

心理学研究中，PCA常被用于抽取统计数据的因子，以此来验证数据的主要维度是否与预期的理论维度一致。而在机器学习的应用中，PCA可以对神经网络内部的隐藏层进行分析，对隐藏层的向量进行降维，进而来研究网络层中的潜在结构。

我们将使用PCA方法对FSNet中的RNN网络进行分析，抽取有效因子，并对因子在时间维度上的变化情况进行分析。RNN网络的运算基于隐藏层的向量，隐藏层在每一时刻得到更新，并在下一时刻作为“记忆”输入，隐藏层包含神经网络是如何运算的关键信息。因此，本研究将使用PCA对隐藏层的活动状态进行分析，尝试分析网络活动过程中的主要特征因子。

模型训练和测试方法

使用人类被试数据划分出的训练数据对FSNet进行训练，采用Adam优化器和MSELoss均方损失函数对网络的参数进行优化，设定loss阈值为0.034，达到阈值后停止模型的训练。为保证分析的模型不存在训练偏差，将该模型输出的反应时均值与分布偏度与研究1中40次训练的分布做t检验，结果显示并不存在显著差异，该模型能够代表普遍的训练结果。

采用人类被试数据划分出的测试数据集对FSNet进行测试，使用该测试集对模型进行72次输出，记录每一次输出中网络隐藏层的变化情况，将72对隐藏层变化序列拼接，得到后续分析需要的网络活动情况数据。

## 结果

PCA聚类结果

通过python程序语言的sklearn包中的PCA模块对拼接后的RNN网络的隐藏层向量进行分析，得到网络隐藏层的主要因子，因子解释率如图1。为了确保因子能够更多的解释网络的活动特性同时保证因子的概括性，参考了Yang等人(2019)使用ANN在直觉决策任务上的筛选标准，最终选取3个因子。此时方差解释率超过85%。

图表, 折线图

描述已自动生成

图1 隐藏层活动的因子累计解释率

对原始的网络输出数据根据上述4个抽取的因子进行转换，得到这4个因子在时间维度上的变化曲线情况，见图2。每一条曲线代表一个试次中网络的活动情况。其中，红色曲线代表72次输出中“不同”刺激试次的网络变化情况，蓝色曲线代表72次输出中“相同”刺激试次的因子变化情况。绿色虚线代表做出相同判断的平均时间点，棕色虚线代表做出不同判断的平均时间点，黑色虚线分割不同的实验阶段，0-1为注视阶段，2-6为刺激呈现阶段，7-21为反应阶段

图表, 折线图

描述已自动生成

图2 各因子在时间维度上的变化情况

因子1的解释率为0.437，绘制成最上方的子图，因子时间变化曲线显示，从刺激呈现的时间点开始， 相同和不同判断的试次均出现一个较大峰值，这一峰值在7时间步，刺激程序完成开始反应的时间点开始下降，对于“不同刺激”条件的试次，试次中的因子1的活动差异较大(*mean\_var* = 0.0021)，对于“相同刺激”条件的试次，试次之间活动差异较小(*mean\_var* = 0.0038)。这说明，“相同刺激”条件下，不同输入刺激与因子1的活动水平并没有显著关联性，而“不同刺激”条件下，输入刺激会引起因子1的明显活动。“相同刺激”和“不同刺激”在因子1层面上表现出明显异质的活动。

因子2的解释率为0.158，绘制成中间的子图，因子时间变化曲线显示，从刺激呈现的时间点开始，相同刺激和不同刺激试次均出现波动，相同试次的波动波幅较小(*mean\_var* = 0.032)，而不同刺激的波动较大(*mean\_var* = 2.2)且逐渐增大，在反应时间点附近达到峰值。因子2中“相同刺激”与“不同刺激”也表现出明显的活动异质性，

因子3的解释率为0.143，因子时间变化曲线显示，绘制成下方的子图，因子3在刺激呈现阶段出现一个负峰值，且“相同刺激”的曲线间差异(*mean\_var* = 0.22)比“不同刺激”的曲线间差异(*mean\_var* = 0.066)较大，这可能说明相同刺激在因子3在刺激呈现阶段对输入刺激活动更敏感，活动水平更高。在反应阶段，因子3出现了类似因子1的活动情况，“不同刺激”条件的试次中活动差异较大(*mean\_var* = 0.0021)， “相同刺激”条件试次之间活动差异较小，但因子1的波动起始时间要较为提前。

对因子的解释

根据对三个因子在时间维度上活动的描述，整体来看，FSNet处理“不同刺激”条件的试次与“相同刺激”条件的试次的过程中，活动水平和模式有着明显的差异，这种差异不仅表现在时间先后上，也表现在活动水平上，这可能说明FSNet对相同的刺激对和不同的刺激对进行处理时可能采用的是不同的算法。

从信息加工的角度来讲，RNN网络的隐藏层承担“记忆”的功能，如果随着时间输入的信息使隐藏层发生改变，则可以认为输入的新信息激发了网络的运算，改变了隐藏层存储的“记忆”，反之，如果随着时间隐藏层的值矩阵较为稳定，则可以认为外界输入的新信息没有被网络处理并改变记忆。因此，隐藏层矩阵的数据在某一时段趋于稳定，可能说明网络并没有在这一时间段上进行运算。网络隐藏层活动的平滑度，可以作为衡量网络对输入刺激以及已有信息的处理过程的活跃程度。

在因子1和因子3中，“相同刺激”的试次均在反应期出现了稳定的平滑段，在因子2中，也出现类似的平滑段，但其稳定性比因子1和因子3低，不过总体来看，要远高于“不同刺激”的试次。这表明这一时期网络可能活动水平较低，而“不同刺激”的试次在三个因子中均有明显的波动，这说明网络隐藏层在持续参与运算并改变值矩阵，也就是说，网络有较高的活动水平。根据这一解释，可以推测网络在反应期的活动特点:当输入相同刺激时，网络在反应期的活动水平较低，当输入不同刺激时，网络在反应期活动水平较高。

在因子3中的刺激呈现期中，出现了一个波，此时“相同”和“不同”两种条件下的因子3均有活动，但比起“不同”刺激，“相同”刺激的活动要更为强烈。这可能说明“相同”刺激的输入使得网络在刺激呈现期有着更多的活动。值得注意的是，相同刺激在刺激呈现阶段产生的活动波的结束时间步和不同刺激在反应期呈现的活动波开始时间步是相近的，这可能说明这两个活动是先后进行的。

在因子1的刺激呈现期中，“相同”和“不同”条件的刺激激发了网络相同的活动状况，因子1的值在刺激呈现后持续增长，并在呈现结束后迅速下跌，并且增长和下跌的过程都较为单调。由于对于RNN来说，持续的刺激呈现代表着将相同的刺激数据在一段时间步內持续输入到网络中参与运算，而由于重复运算某个刺激带来的值矩阵变化很可能是单调的。因此，刺激呈现期因子1的活动很可能说明网络识别并存储输入刺激的过程。

快同效应的认知机制

根据对因子活动特点的描述和解释，我们能够结合现有的理论来证明本文的研究假设。

现有的理论认为，不论是异同判断的单过程模型还是双过程模型，都存在一个慢速比较器和一个同一性指示器来给出判断。争论的焦点在于这一过程是否是同时进行或者顺序进行的(Farell, 1985; 张之光, 禤宇明, 傅小兰, 2003)。

根据本研究对PCA分析结果的解释，FSNet在加工相同刺激和加工不同刺激时，有明显异质的加工过程，在相同刺激加工过程中，模型活动主要集中在刺激呈现阶段，而在不同刺激加工过程中，模型活动主要集中在刺激呈现后的反应阶段，而在刺激呈现阶段也有一定水平的活动。这两个加工过程与现有理论中的双过程模型的假设一致。这支持本研究的假设2。

FSNet在刺激呈现阶段，对于相同刺激和不同刺激均有加工，但相同刺激的加工比之不同刺激的活动水平较高，这可能说明不同刺激在刺激呈现之后有额外的加工过程。根据慢速比较器和同一性指示器的特性，相同刺激在呈现阶段的高活动水平可能是由于编码和存储输入的刺激与同一性指示器的活动带来的。而不同刺激在刺激呈现之后仅进行编码和存储，而不进行额外的加工过程。

在反应期，FSNet对于不同刺激有着更明显的加工，而对相同刺激的加工则几乎停止，并且对于不同刺激的加工过程是紧随着相同刺激在刺激呈现阶段的活动之后的。不同刺激在反应期的活动与双过程模型中假设的慢速比较器的活动规律是一致的。FSNet对相同刺激处理和不同刺激处理这两个过程在时间上的顺序性，说明在异同判断任务中，可能基于刺激是相同或不同的而存在两个先后的加工过程，这支持了慢速比较器和同一性指示器是顺序进行加工的这一假说。

综上，根据相关理论和对模型活动的分析，可以对异同判断的过程做出解释，并以此解释快同效应的原理：

对于异同判断任务来说，加工很有可能是双过程的，并且其中的慢速比较器和同一性指示器是顺序加工的，对于输入的刺激，同一性指示器先进行处理，如果判断结果为相同，则直接输出结果而不继续后续加工，如果判断结果为不同，则使用慢速比较器后续进一步进行加工，最后输出判断结果。由于对相同刺激来说，只需要进行同一性指示器一个加工过程，而不同刺激不仅要进行同一性指示器的加工，还要调用后续的慢速比较器，需要更长的加工流程，进而导致了更长的反应时。因此，加工过程的不同导致了异同判断任务中快同效应的出现。