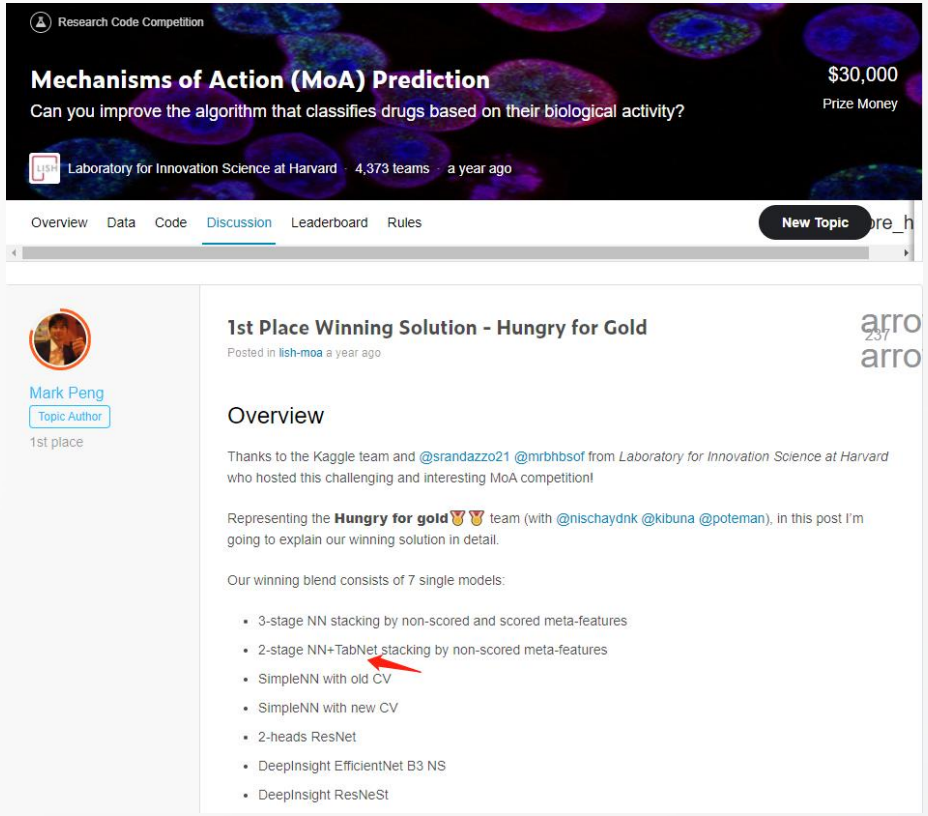
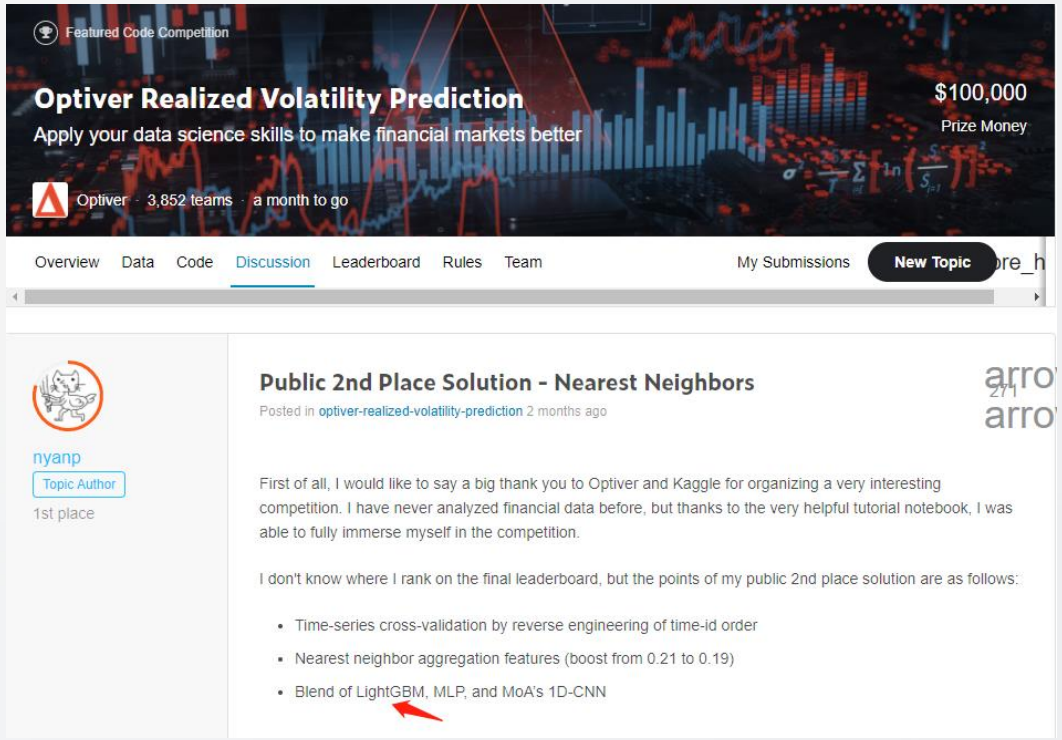
清華大學
Tsinghua University

TabNet: Attentive Interpretable Tabular Learning

汇报人：赵越 蔡紫宴 吴定俊

目录


Research Code Competition
Mechanisms of Action (MoA) Prediction
Can you improve the algorithm that classifies drugs based on their biological activity?
$30,000
Prize Money
Laboratory for Innovation Science at Harvard · 4,373 teams · a year ago
Overview Data Code Discussion Leaderboard Rules
New Topic
1st Place Winning Solution - Hungry for Gold
Posted in lish-moa a year ago
Mark Peng
Topic Author
1st place
Overview
Thanks to the Kaggle team and @srandazzo21 @mrbhbsof from Laboratory for Innovation Science at Harvard who hosted this challenging and interesting MoA competition!
Representing the Hungry for gold team (with @nischaydnk @kibuna @poteman), in this post I'm going to explain our winning solution in detail.
Our winning blend consists of 7 single models:
3-stage NN stacking by non-scored and scored meta-features
2-stage NN+TabNet stacking by non-scored meta-features
SimpleNN with old CV
SimpleNN with new CV
2-heads ResNet
DeepInsight EfficientNet B3 NS
DeepInsight ResNeSt


Featured Code Competition
Optiver Realized Volatility Prediction
Apply your data science skills to make financial markets better
$100,000
Prize Money
Optiver · 3,852 teams · a month to go
Overview Data Code Discussion Leaderboard Rules Team
My Submissions
New Topic
Public 2nd Place Solution - Nearest Neighbors
Posted in optiver-realized-volatility-prediction 2 months ago
nyanp
Topic Author
1st place
First of all, I would like to say a big thank you to Optiver and Kaggle for organizing a very interesting competition. I have never analyzed financial data before, but thanks to the very helpful tutorial notebook, I was able to fully immerse myself in the competition.
I don't know where I rank on the final leaderboard, but the points of my public 2nd place solution are as follows:
Time-series cross-validation by reverse engineering of time-id order
Nearest neighbor aggregation features (boost from 0.21 to 0.19)
Blend of LightGBM, MLP, and MoA's 1D-CNN

1.1 DT-based 和 DNN-based 模型的优点

基于决策树的模型（DT-based）

优点：

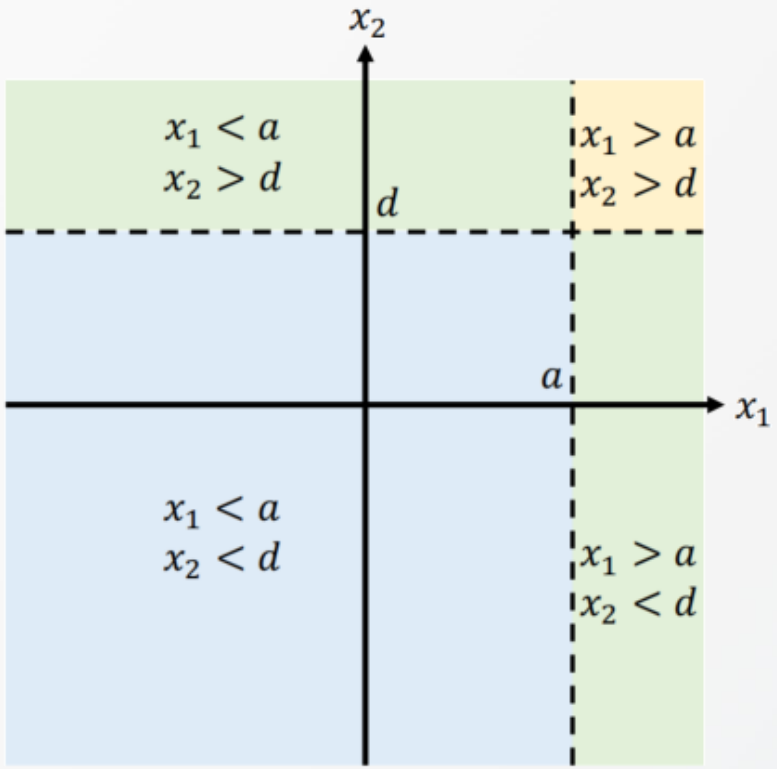
1. 适用性：具有近似超平面的决策流形（decision manifolds），对表格数据友好
2. 可解释性：可以跟踪决策节点，追溯推断过程，可解释性较好
3. 速度：训练时间短

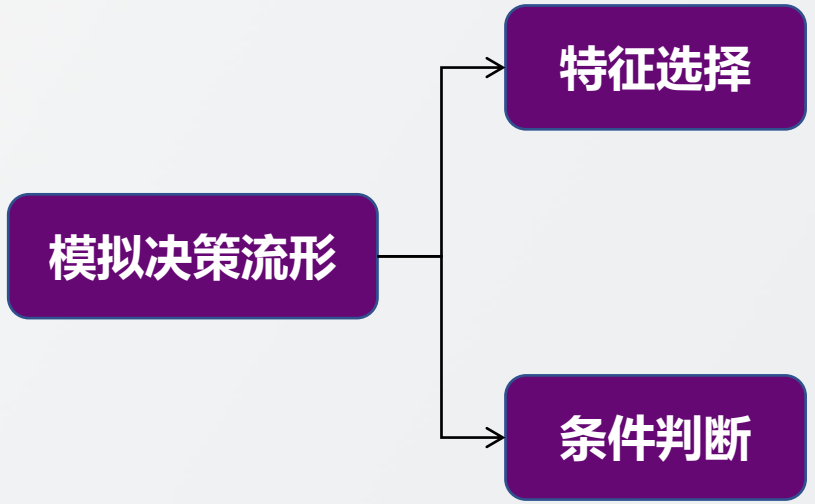
基于深度神经网络的模型（DNN-based）

优点：

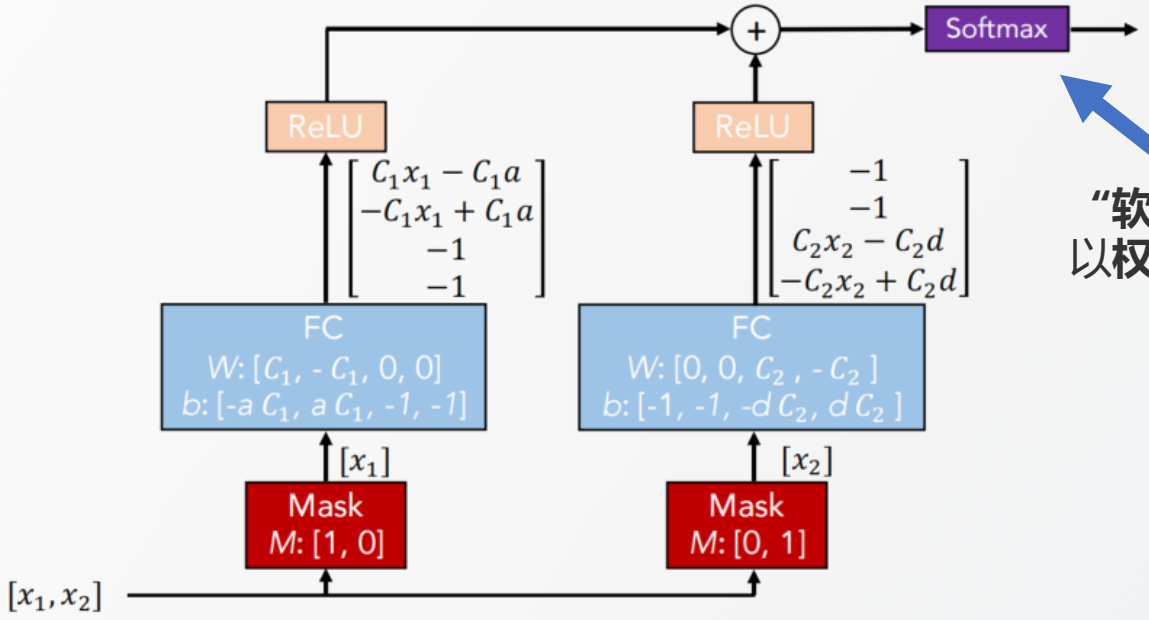
1. 表征方式：可以像图片、文本一样，对表格数据进行编码
2. 自动化：减少对表格数据对特征工程的依赖
3. 表征学习：端对端的模型允许表征学习，能应用在有价值的新应用场景中，包括: 迁移学习 (Data-efficient Domain Adaption)，生成模型 (Generative Modeling) 和半监督学习 (Semi-supervised Learning)。

2.1 树的决策流形





2.2 TabNet设计思想



“软性” 的特征选择：
以**权重**形式输出

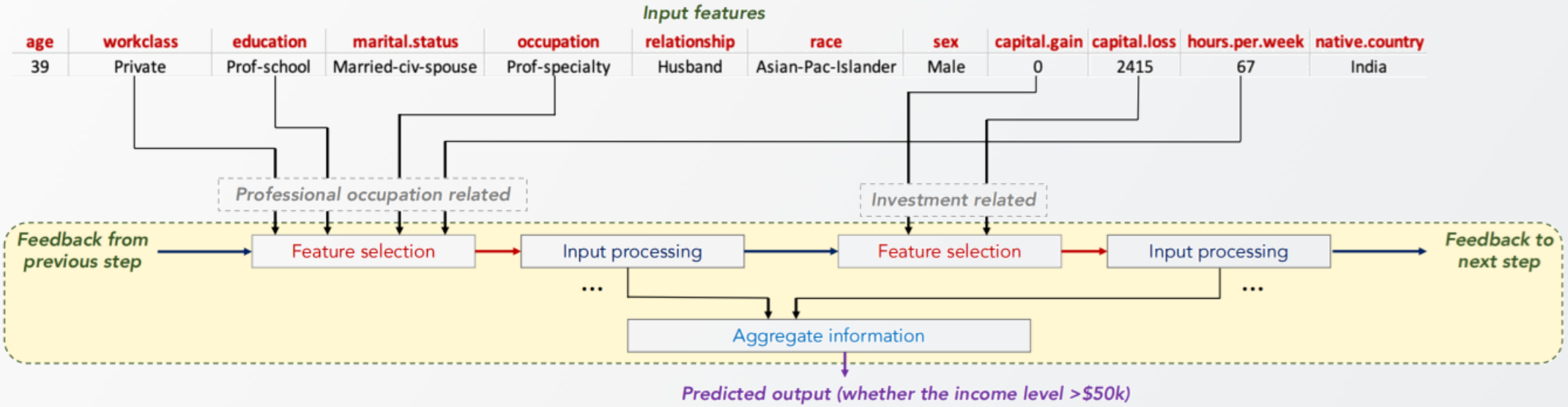
TabNet结构（简单版）

Mask：特征选择
FC+ReLU：条件判断
两棵“基本决策树”->结果相加
Softmax计算权重：[0.1, 0.4, 0.3, 0.2]

1. $Mask$：
$x:(x_1,x_2)->Mask()->Mask(x)$
$eg.\ M:[1,0]->[x_1]$
2. $FC\ Layer$：
$W_1 \cdot x_1 + b1$
$=[C_1,-C_1,0,0]+[-aC_1,aC_1,-1,-1]$
$=[C_1(x_1-a),C_1(a-x_1),-1,-1]$
$=[f(x_1),-f(x_1),-1,-1]$
3. $ReLU\ Layer$：
$\begin{cases}[f(x_1),0,0,0], x1 > a\ and\ C_1 > 0\\ [0,-f(x_1),0,0], x1 < a\ and\ C_1 > 0\end{cases}$
4. 区域相加->决策流形
$[x_1 > a, x_1 < a, 0, 0]$
$[x_1 > a, x_1 < a, x_2 > d, x_2 < d]$
5.$Softmax$->权重：
$[0.1, 0.4, 0.2, 0.3]$

简单计算

2.3 稀疏性的特征选择

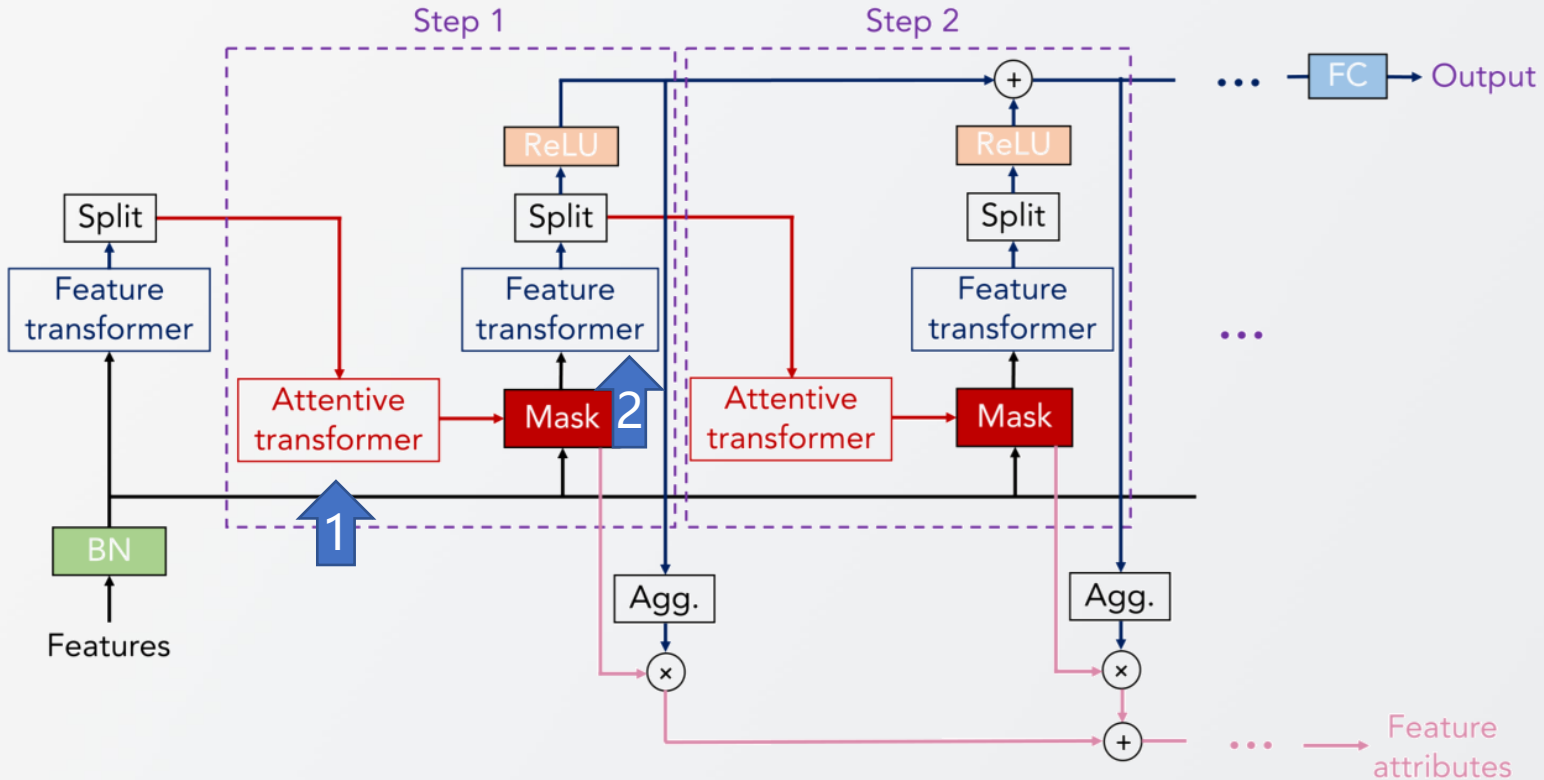


成人人口普查收入预测为例(Dua and Graff 2017)

2.4 TabNet Encoder完整结构

输入：(B, D)，B 是batch size，D 是feature的维数

2大模块
- Attentive transformer
- Feature transformer



(a) TabNet encoder architecture

2.5 TabNet组成部分 - Attentive Transformer

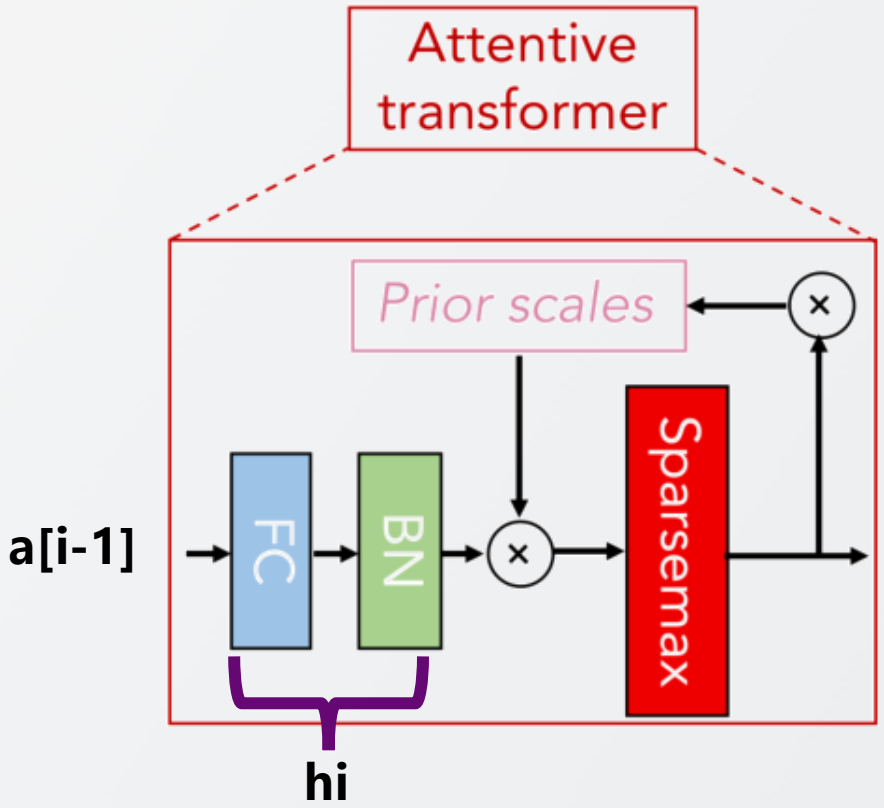
a[i-1]，历史已经处理好的特征（作为输入）
P[i]：先验尺度项 (Prior Scale Term)，用于告知模型某个特征在历史训练里被使用的程度

$$M[i] = sparsemax(P[i-1] \cdot h_i(a[i-1]))$$

$$P[i] = \prod_{j=1}^{i}(\gamma - M[j])$$

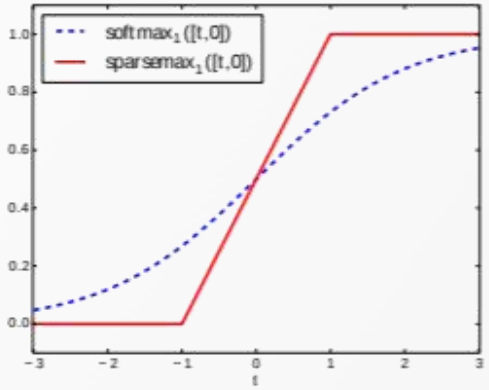
作用：M[i]· f，对特征进行遮掩

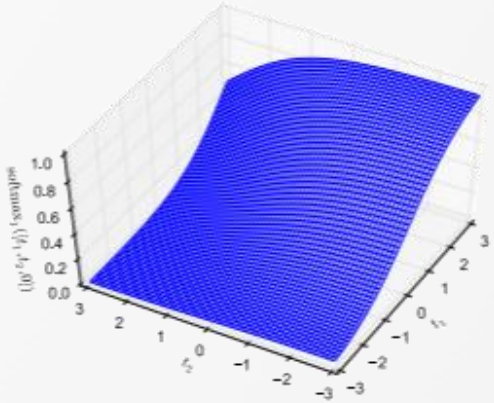
P[0] 是被全部初始化为1的矩阵，即1∈(B, D)

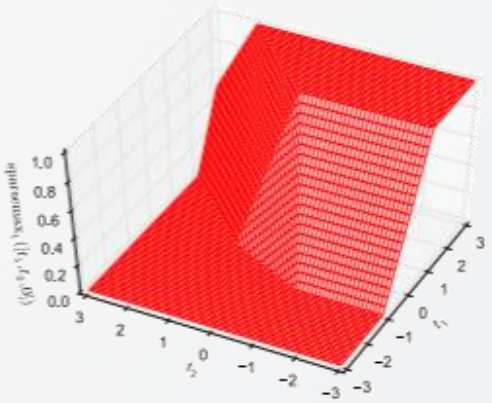


负责特征选择的Attentive Transformer

2.6 sparsemax







Sparsemax的性质有 $\sum_{j=1}^{D} M[i]_{b,j} = 1$，因此 M[i] 可以理解为模型在当前step上，对于batch样本的**注意力权重分配。** 对于不同的样本，Attentive transformer层输出的注意力权重也不同（论文中称**instance-wise**）

$$M[i] = sparsemax(P[i-1] \cdot h_i(a[i-1])) \qquad \text{(B, D)}$$

2.7 稀疏正则项

$$L_{sparse} = \sum_{i=1}^{N_{steps}} \sum_{b=1}^{B} \sum_{j=1}^{D} \frac{-M_{b,j}[i]}{N_{steps} \cdot B} log(M_{b,j}[i] + \epsilon)$$

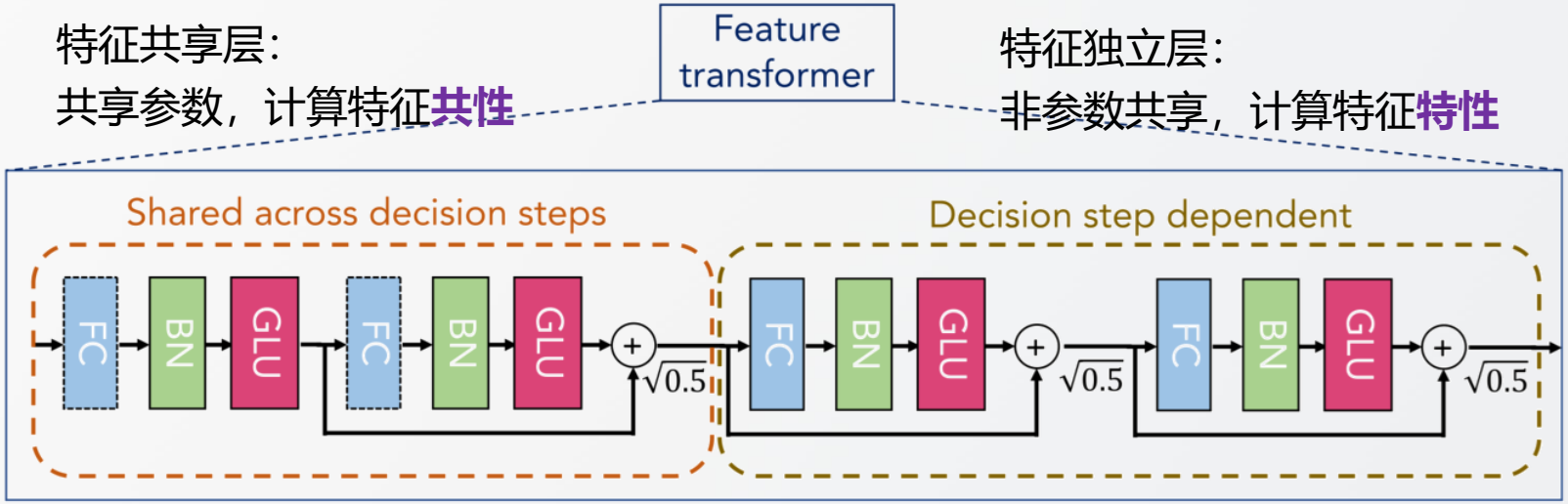
ε 是个小的数值

2.8 TabNet组成部分 - Feature Transformer

特征共享层：
共享参数，计算特征**共性**

Feature transformer

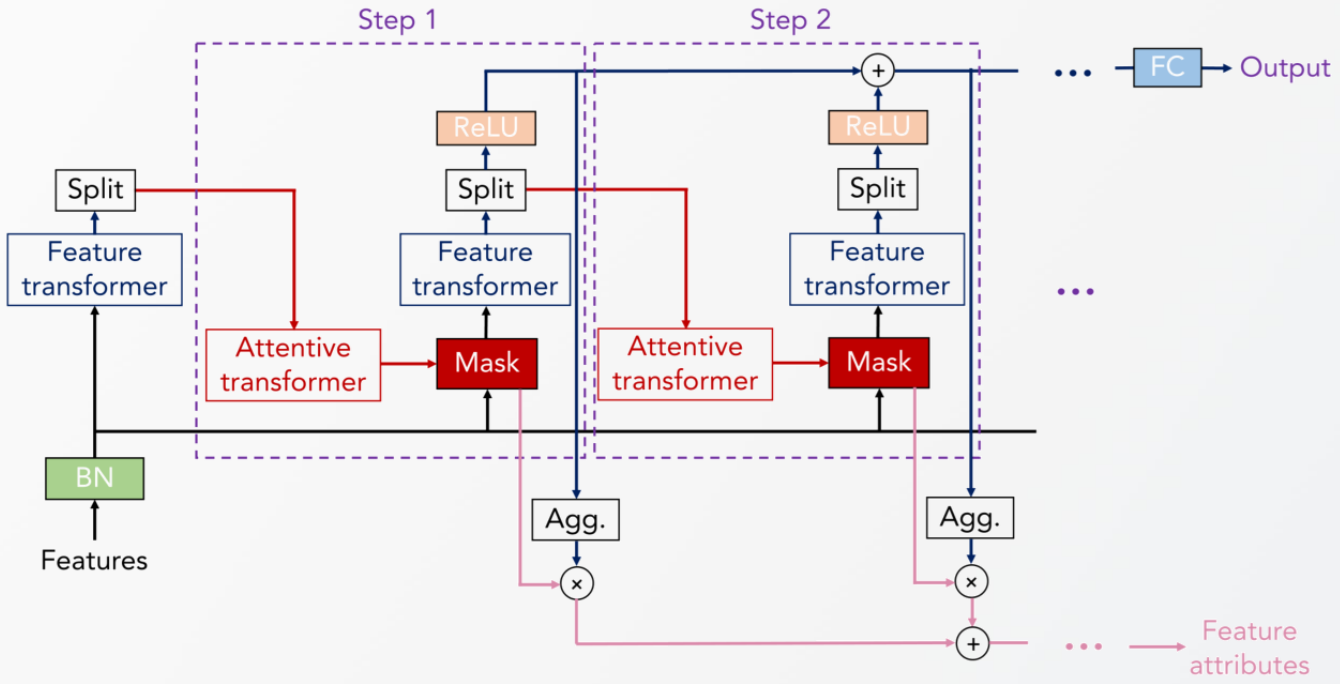
特征独立层：
非参数共享，计算特征**特性**



负责特征处理的 Feature Transformer

GLU 门控: $h(X) = (W * X + b) \otimes \sigma(V * X + c)$

2.9 Split 层



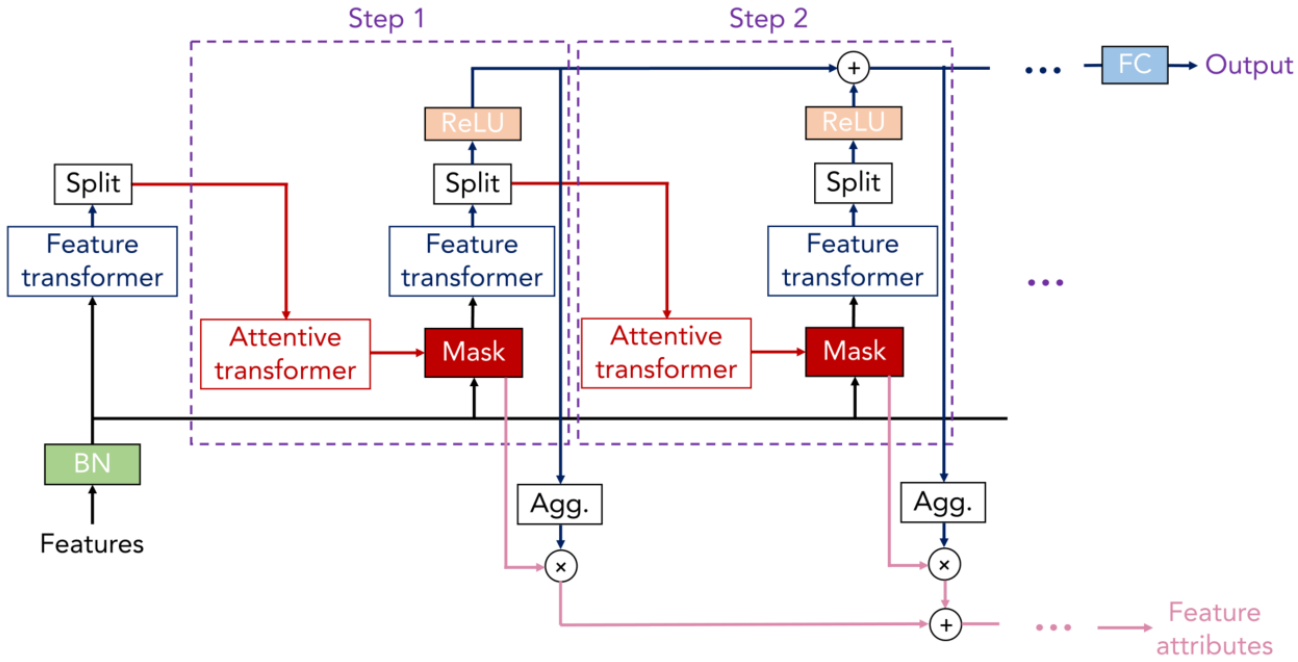
(a) TabNet encoder architecture

Feature Transformer的输出结果经过split，得到

$$[\mathbf{d}[\mathbf{i}], \mathbf{a}[\mathbf{i}]] = \mathbf{f}_i(\mathbf{M}[\mathbf{i}] \cdot \mathbf{f})$$

其中 d[i] 将用于计算模型的最终输出，而 a[i] 则用来计算下一个step的Mask层。

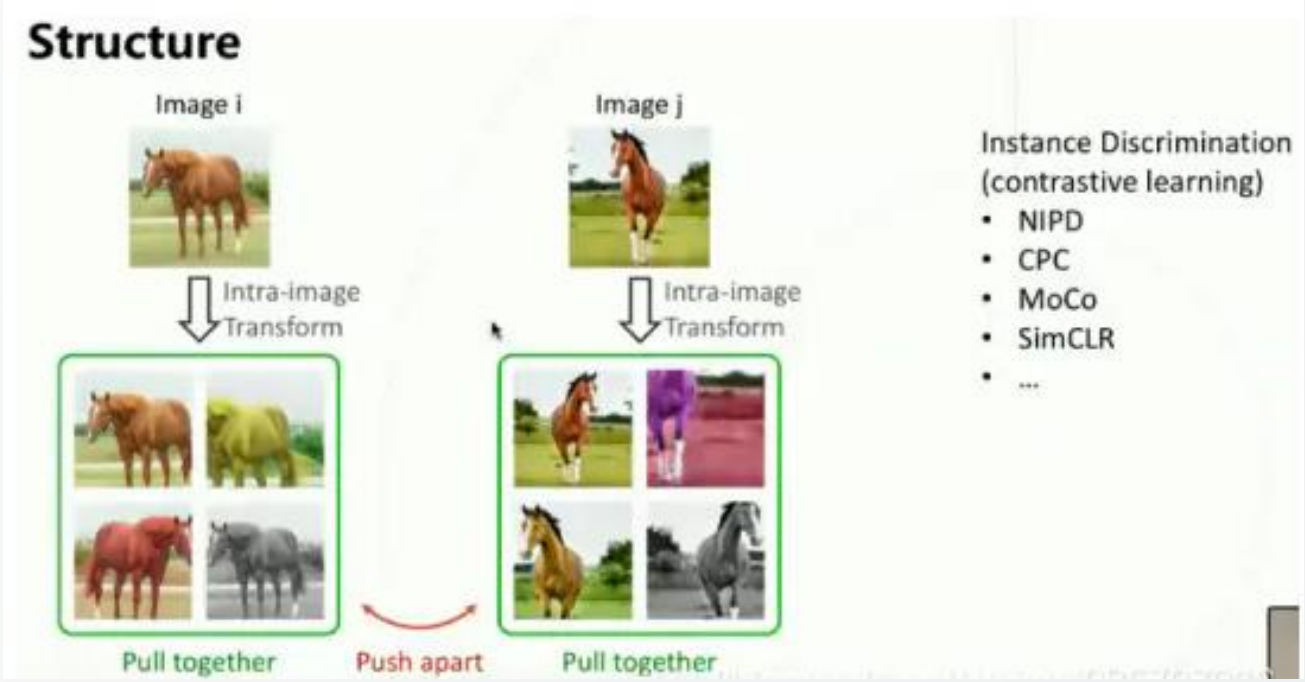
2.9 最终输出



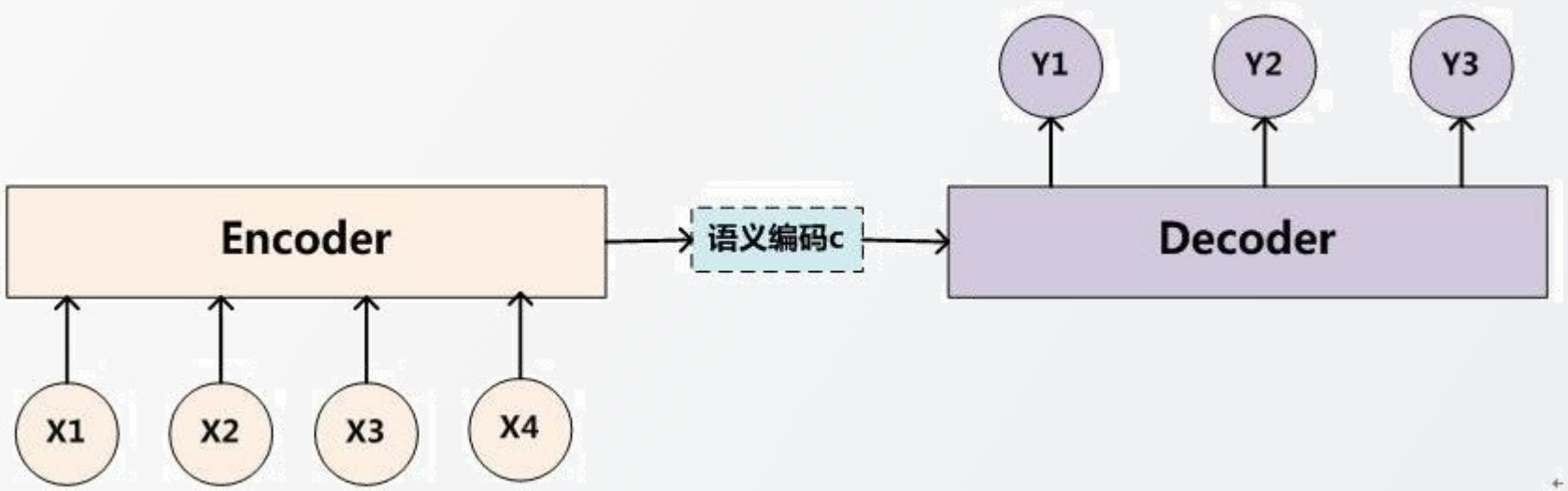
(a) TabNet encoder architecture

$$\mathbf{d_{out}} = \sum_{i=1}^{N_{steps}} \mathrm{ReLU}(\mathbf{d[i]})$$

自监督学习主要是利用辅助任务（pretext）从大规模的无监督数据中挖掘自身的监督信息，通过这种构造的监督信息对网络进行训练，从而可以学习到对下游任务有价值的表征。



encoder-decoder模型，又叫做编码-解码模型，应用于seq2seq问题。



Unsupervised pre-training

Age	Cap. gain	Education	Occupation	Gender	Relationship
53	200000	?	Exec-managerial	F	Wife
19	0	?	Farming-fishing	M	?
?	5000	Doctorate	Prof-specialty	M	Husband
25	?	?	Handlers-cleaners	F	Wife
59	300000	Bachelors	?	?	Husband
33	0	Bachelors	?	F	?
?	0	High-school	Armed-Forces	?	Husband

TabNet encoder

TabNet decoder

Age	Cap. gain	Education	Occupation	Gender	Relationship
		Masters			
		High-school			Unmarried
43					
	0	High-school		F	
			Exec-managerial	M	
			Adm-clerical		Wife
39				M	

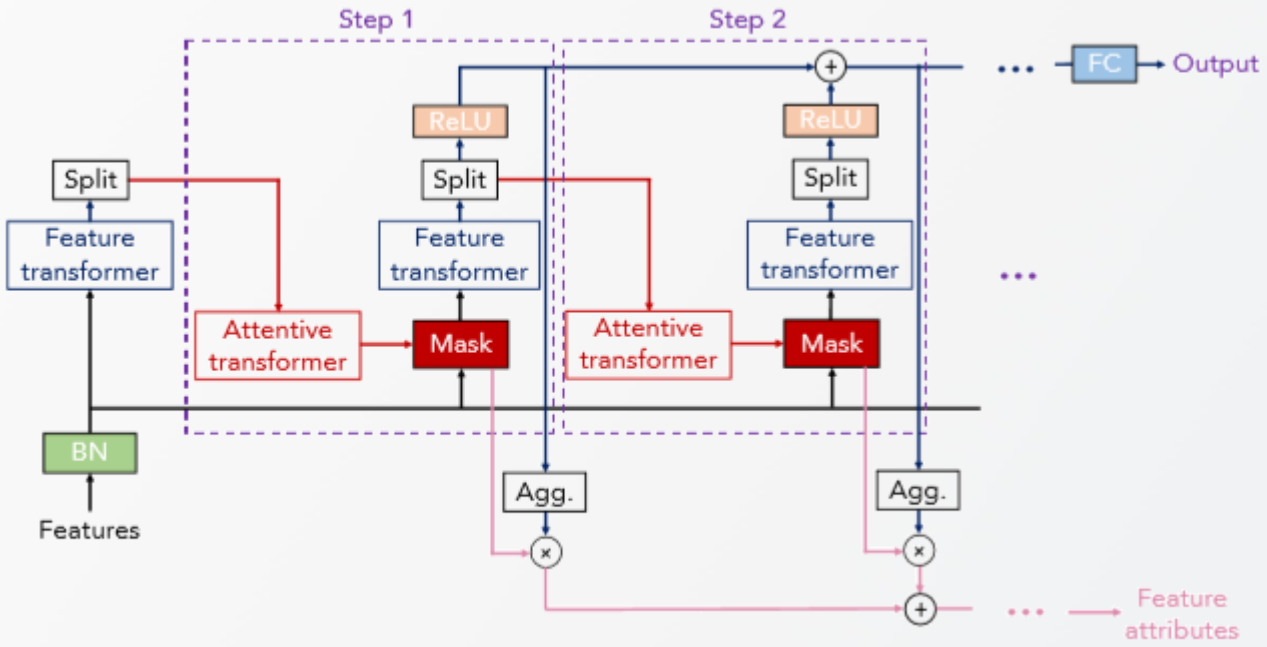
Supervised fine-tuning

Age	Cap. gain	Education	Occupation	Gender	Relationship
60	200000	Bachelors	Exec-managerial	M	Husband
23	0	High-school	Farming-fishing	M	Unmarried
45	5000	Doctorate	Prof-specialty	M	Husband
23	0	High-school	Handlers-cleaners	F	Wife
56	300000	Bachelors	Exec-managerial	M	Husband
38	10000	Bachelors	Prof-specialty	F	Wife
23	0	High-school	Armed-Forces	M	Husband

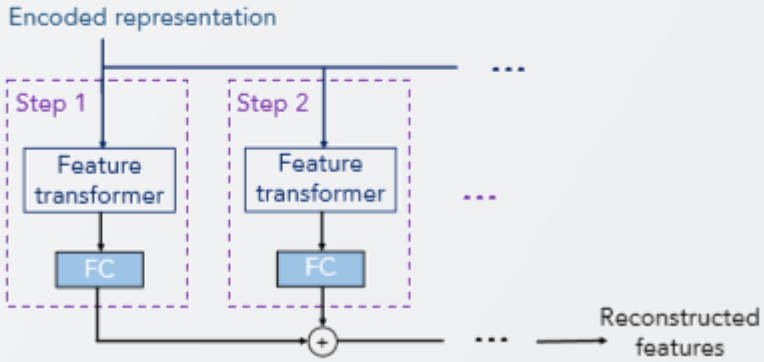
TabNet encoder

Decision making

Income > $50k
True
False
True
False
True
True
False



(a) TabNet encoder architecture



(b) TabNet decoder architecture

设在一开始对feature做mask的矩阵是 $S \in \{0,1\}^{B\times D}$ ，特征数据是 f ，则encoder的输入是 $(1-S)\cdot f$ ，若最后decoder的输出是 $\hat{f}$ ，那么自监督学习就是减小真实值 $S\cdot f$ 与重构值 $S\cdot \hat{f}$ 之间的差别，考虑到不同的feature的量级不一定相同，因此采用正则化后的MSE作为loss，形式如下：

$$\sum_{b=1}^{B}\sum_{j=1}^{D}\left|\left(\hat{f}_{b,j}-f_{b,j}\right)\cdot S_{b,j}/\sqrt{\sum_{b=1}^{B}\left(f_{b,j}-1/B\sum_{b=1}^{B}f_{b,j}\right)^2}\right|^2$$

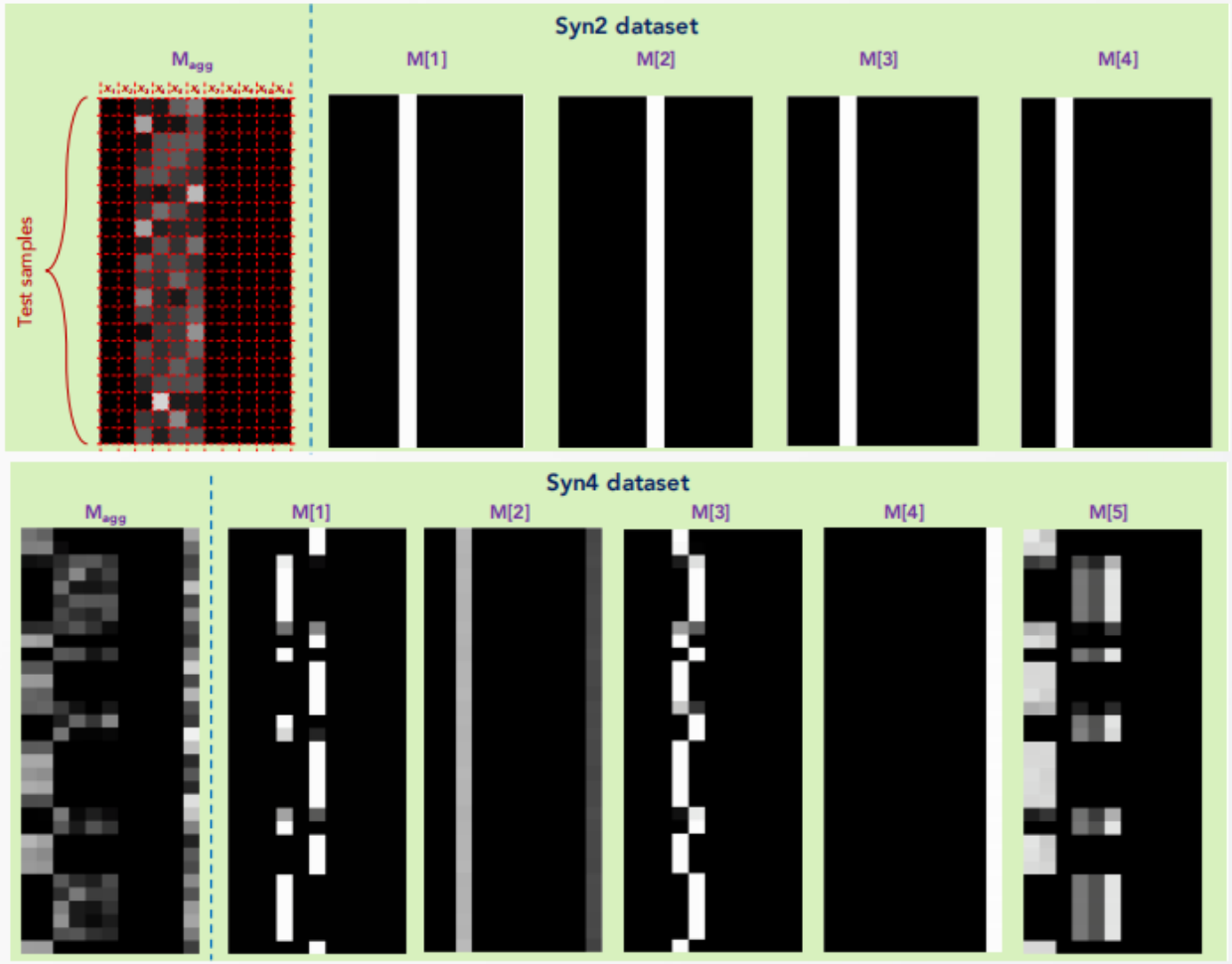
4.1 Instance-wise feature selection

Dataset：6 tabular datasets from (Chen et al. 2018) (consisting 10k training samples)

For Syn1- Syn3, salient features are same for all instances (e.g., the output of Syn2 depends on features $X3$-$X6$).
For Syn4-Syn6, salient features are instance dependent (e.g., for Syn4, the output depends on either $X1$-$X2$ or $X3$-$X6$ depending on the value of $X11$).

Model	*Test AUC*					
	Syn1	Syn2	Syn3	Syn4	Syn5	Syn6
No selection	.578 ± .004	.789 ± .003	.854 ± .004	.558 ± .021	.662 ± .013	.692 ± .015
Tree	.574 ± .101	.872 ± .003	.899 ± .001	.684 ± .017	.741 ± .004	.771 ± .031
Lasso-regularized	.498 ± .006	.555 ± .061	.886 ± .003	.512 ± .031	.691 ± .024	.727 ± .025
L2X	.498 ± .005	.823 ± .029	.862 ± .009	.678 ± .024	.709 ± .008	.827 ± .017
INVASE	**.690 ± .006**	.877 ± .003	**.902 ± .003**	**.787 ± .004**	.784 ± .005	.877 ± .003
Global	.686 ± .005	.873 ± .003	.900 ± .003	.774 ± .006	.784 ± .005	.858 ± .004
TabNet	.682 ± .005	**.892 ± .004**	.897 ± .003	.776 ± .017	**.789 ± .009**	**.878 ± .004**

4.1 Instance-wise feature selection



$$\eta_{\mathrm{b}}[\mathrm{i}] = \sum_{c=1}^{N_d} \mathrm{ReLU}(\mathrm{d}_{\mathrm{b,c}}[\mathrm{i}])$$

$$\mathbf{M}_{\mathbf{agg-b,j}} = \sum_{i=1}^{N_{steps}} \eta_{\mathrm{b}}[\mathbf{i}]\mathbf{M}_{\mathbf{b,j}}[\mathbf{i}] / \sum_{j=1}^{D} \sum_{i=1}^{N_{steps}} \eta_{\mathrm{b}}[\mathbf{i}]\mathbf{M}_{\mathbf{b,j}}[\mathbf{i}]$$

- For Syn2, the output depends on *X*3-*X*6 .
- For Syn4, the output depends on either *X*1-*X*2 or *X*3-*X*6 depending on the value of *X*11.

4.2 real-world datasets

Dataset：Forest Cover Type

Task：classification of forest cover type from cartographic variables.

Model	*Test accuracy (%)*
XGBoost	89.34
LightGBM	89.28
CatBoost	85.14
AutoML Tables	94.95
TabNet	**96.99**

Dataset： Rossmann Store Sales

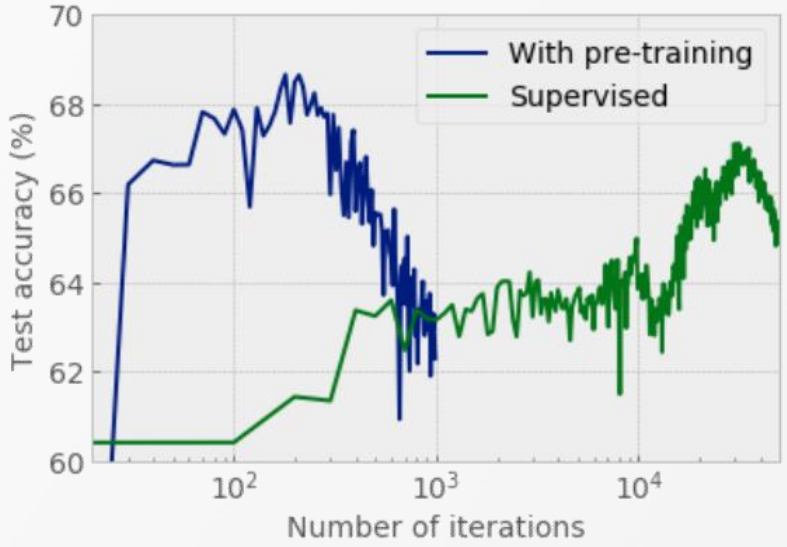
Task： regression of forecasting the store sales from static and time-varying features.

Model	*Test MSE*
MLP	512.62
XGBoost	490.83
LightGBM	504.76
CatBoost	489.75
TabNet	**485.12**

4.3 Higgs Boson (Self-supervised learning)

Dataset：The Physical Realm

Task：distinguishing between a Higgs bosons process vs. background (10.5M instances).



Model	*Test acc. (%)*	*Model size*
Sparse evolutionary MLP	**78.47**	**81K**
Gradient boosted tree-S	74.22	0.12M
Gradient boosted tree-M	75.97	0.69M
MLP	78.44	2.04M
Gradient boosted tree-L	76.98	6.96M
TabNet-S	78.25	81K
TabNet-M	**78.84**	**0.66M**

Training dataset size	*Test accuracy (%)*	
	Supervised	*With pre-training*
1k	57.47 ± 1.78	$\mathbf{61.37 \pm 0.88}$
10k	66.66 ± 0.88	$\mathbf{68.06 \pm 0.39}$
100k	72.92 ± 0.21	$\mathbf{73.19 \pm 0.15}$

4.4 我们的实验（基于线下商店销量预测数据集）

1. 任务：给定商店销量历史相关数据和时间等信息，预测商店对应商品的周销量。

2. 数据说明：数据集由字段shop_id（店铺id）、item_id（商品id）、week（周标识）、item_price（商品价格）、item_category_id（商品品类id）、weekly_sales（周销量）组成。

3. 评价指标：MSE

$$\frac{1}{n}\sum_{n=1}^{n}(y_i-\bar{y_i})^2$$

	shop_id	item_id	week	item_price	item_category_id	weekly_sales	train	weekly_sales_lag_1	weekly_sales_lag_2	weekly_sales_lag_3	weekly_sales_lag_4
0	0	0	4	NaN	0	0.0	1	1.0	2.0	1.0	2.0
1	0	0	5	NaN	0	1.0	1	0.0	1.0	2.0	1.0
2	0	0	6	399.0	0	4.0	1	1.0	0.0	1.0	2.0
3	0	0	7	399.0	0	3.0	1	4.0	1.0	0.0	1.0
4	0	0	8	399.0	0	1.0	1	3.0	4.0	1.0	0.0
...	...	...	...	...	...	...	...	...	...	...	...
468603	31	522	27	NaN	0	1.0	1	0.0	0.0	0.0	0.0
468604	31	522	28	NaN	0	0.0	1	1.0	0.0	0.0	0.0
468605	31	522	29	NaN	0	1.0	1	0.0	1.0	0.0	0.0
468606	31	522	30	NaN	0	0.0	1	1.0	0.0	1.0	0.0
468607	31	522	31	NaN	0	1.0	1	0.0	1.0	0.0	1.0

468608 rows × 11 columns

4. 数据预处理

简单的特征工程：获得商品的历史销量（lag1-4 周），在使用TabNet时对价格空缺值进行了填充

数据集划分：第4-31周作为训练集，第32周作为测试集；5折交叉验证

参数：默认参数

	MSE
LightGBM	1.790 (fast)
TabNet	1.889 (slow)

TabNet

- 针对于表格数据的神经网络
- 加性模型的顺序注意力机制（sequential attention mechanism）
- instance-wise的特征选择
- encoder-decoder框架实现了自监督学习
- 将树模型的可解释性与DNN的表征能力很好地结合

清華大學
Tsinghua University

谢 谢 大 家 ！

汇报人：赵越 蔡紫宴 吴定俊