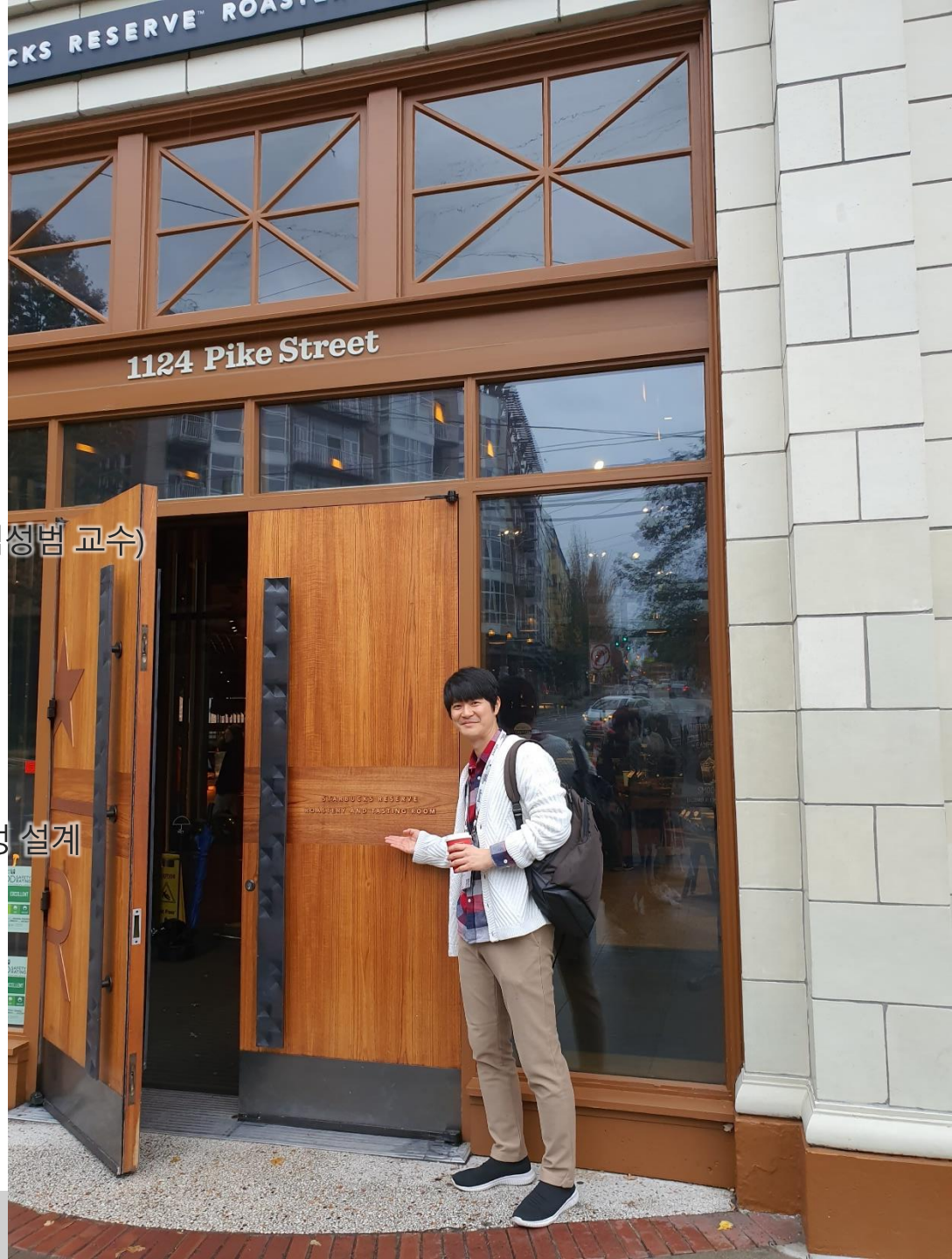
Deep Learning for Tabular Dataset

DMQA Open Seminar

2021.7.30

발표자 소개

- 강성현 (Seonghyeon Kang)
  - 고려대학교 산업경영공학과
  - Data Mining & Quality Analytics Lab (김성범 교수)
  - 박사과정 (2021.3 ~)
- 관심 연구분야
  - 머신러닝을 활용한 무선 통신시험 공정 설계
  - Anomaly detection, Explainable AI



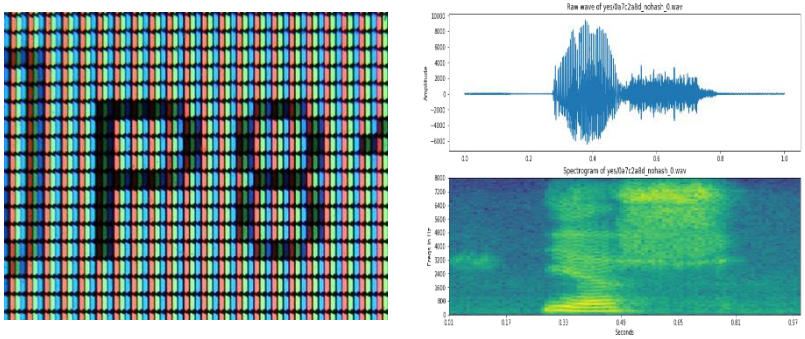
목차

개요 : Tabular Data 정의

❖ Feature는 Column, Sample는 Row 방향으로 결합된 테이블 형태의 Dataset을 의미

- RDMS(관계형 데이터베이스)는 하나 이상의 테이블로 이루어져 있으며, 각 테이블은 Key와 Value의 관계를 통해 표현하고 싶은 대상을 추상화 함
- RDBMS의 보편화로 인해 현존하는 IT System을 통해 빈번히 경험할 수 있는 데이터 구조

Unstructured Dataset



Tabular Dataset

Primary key

Foreign key

Table Name: **EMP**

EMP NO	ENAME	JOB	DEPT NO
7839	KING	PRESIDENT	10
7698	BLAKE	MANAGER	30
7782	CLARK	MANAGER	10
7566	JONES	MANAGER	20

Primary key

Table Name: **DEPT**

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

개요 : Tabular Data 정의

Unstructured Dataset

Tabular Dataset

ID	시침 각도	분침 각도	초침 각도	시간
1	300	60	0	10시 10분
2	180	0	0	06시 00분
⋮	⋮	⋮	⋮	

개요 : 딥러닝의 필요성

❖ Tabular Data에서 딥러닝 연구와 현장 사이의 "Gap"

- 딥러닝 성능은 이미지, 음성, 언어와 같은 비정형 데이터 대비 정형 데이터에서 인상적이지 못함
- 또한, 학습 비용(시간, 인프라 비용) 대비 경제성 부족과 해석이 어려운 문제가 있음
- 연구는 비정형 데이터를 활용한 딥러닝 vs 현업 프로젝트는 feature engineering에 집중하는 모순

Research ML		Production ML
Offline, 불변	데이터 특징	Online, Streaming
SOTA (Accuracy, RMSE 등)	평가 방법	모델성능, Inference 속도, 해석
독창성과 높은 성능	중요 요소	경제성 및 안정적인 운영
모델 변경 시	학습 특성	데이터 변경
기반 기술 확보	목표	서비스 제공

* https://zzsza.github.io/mlops/2018/12/28/mlops 일부 내용 수정

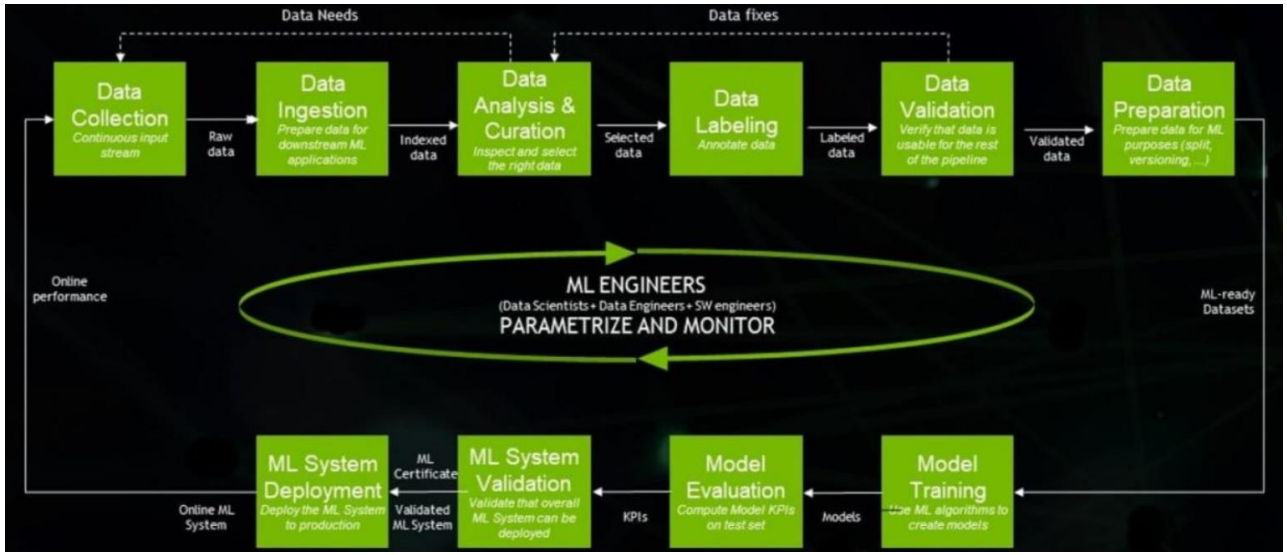
개요 : 딥러닝의 필요성

❖ 딥러닝의 점진적 학습 특성과 사전 학습 가능성은 새로운 기회가 될 수 있음

- Incremental Learning : streaming data에서 지속적인 학습이 가능함
- Pretraining : 사전 학습으로 학습 시간 단축 및 적은 데이터 사용으로 성능 향상 가능
- Capacity : 복잡하고 다양한 패턴을 수용, 데이터가 누적됨에 따라 지속적인 성능 향상 기대

▶ 해석까지 할 수 있다면 ?

ML Lifecycle for a Production (Nvidia)



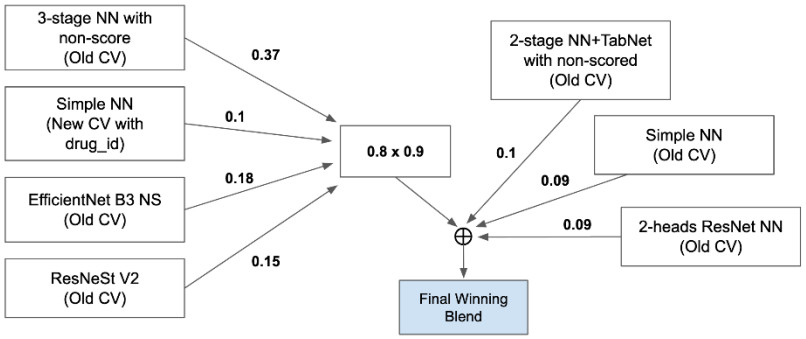
* https://blogs.nvidia.co.kr/2020/09/11/what-is-mlops

TabNet : 알고리즘 소개

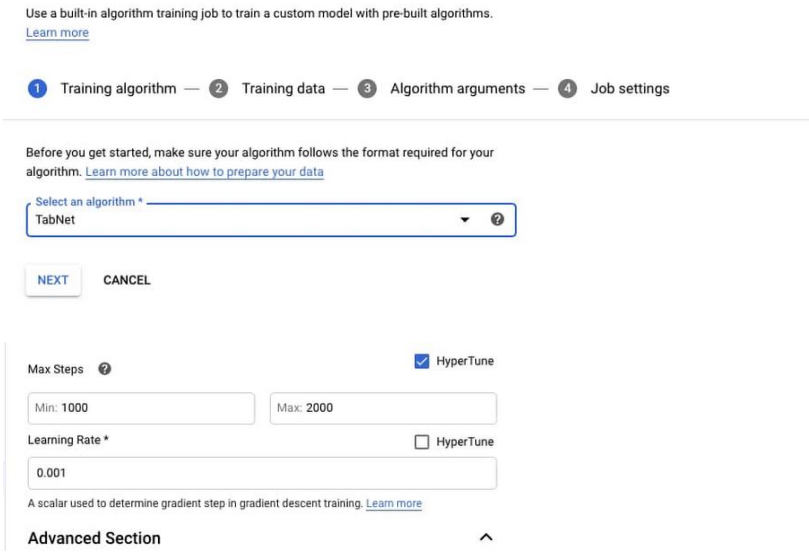
❖ TabNet on AI Platform: High-performance, Explainable Tabular Learning

- 캐글 MoA(Mechanisms of Action Prediction) 대회의 baseline 모델이며, 우승 알고리즘에도 활용됨
  * 900여개 특성, 5000여개 샘플로 200개 Action을 예측하는 multi-label classification 문제
- Google ML Platform인 Vertex AI에 탑재되어 GCP 내에서 즉시 사용 가능

Winning Weighted-Average Blend for MoA



GCP User Guide for Tabular



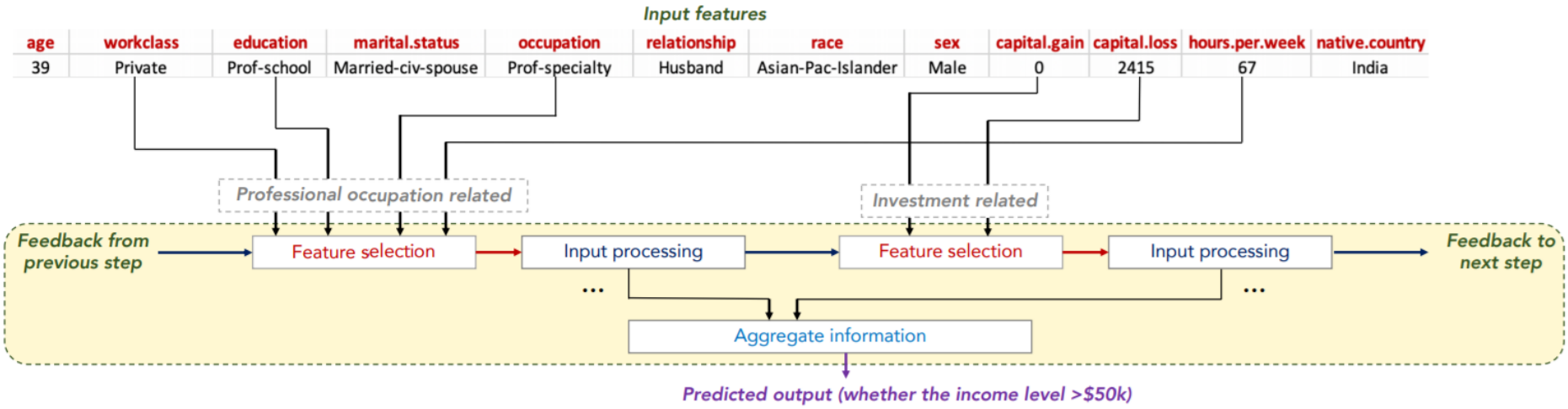
* https://www.kaggle.com/c/lish-moa/discussion/201510
* https://cloud.google.com/blog/products/ai-machine-learning/ml-model-tabnet-is-easy-to-use-on-cloud-ai-platform

TabNet : Main Concept

❖ Tree 기반 모델의 변수선택 특징을 네트워크 구조에 반영한 딥러닝 모델

- 가공하지 않은 Raw Data에서 Gradient를 기반한 최적화를 사용함으로써 End-to-End 학습을 실현
- Sequential attention mechanism을 사용하여 모델의 성능과 해석 용이성을 향상

Sparse Feature Selection 동작 원리



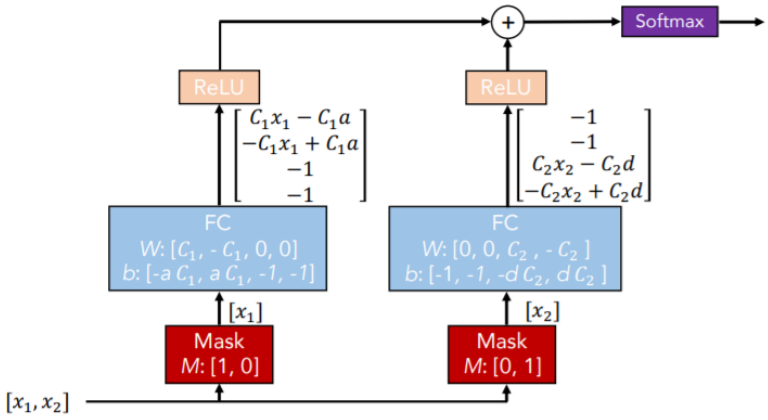
* 논문 인용

TabNet : Feature Selection

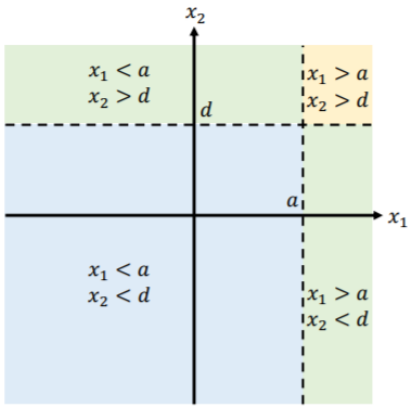
❖ (관련 연구) Conventional DNN 블록으로 Tree와 유사한 형태의 결정 경계를 생성

- sparse **instance-wise** feature selection learned from data
- constructs a **sequential** multi-step architecture, where each step contributes to a portion of the decision based on the selected features
- improves the learning capacity via nonlinear processing of the selected features

Conventional DNN 블록과 Tree의 결정 경계 비교



Conventional DNN 블록



Tree 알고리즘

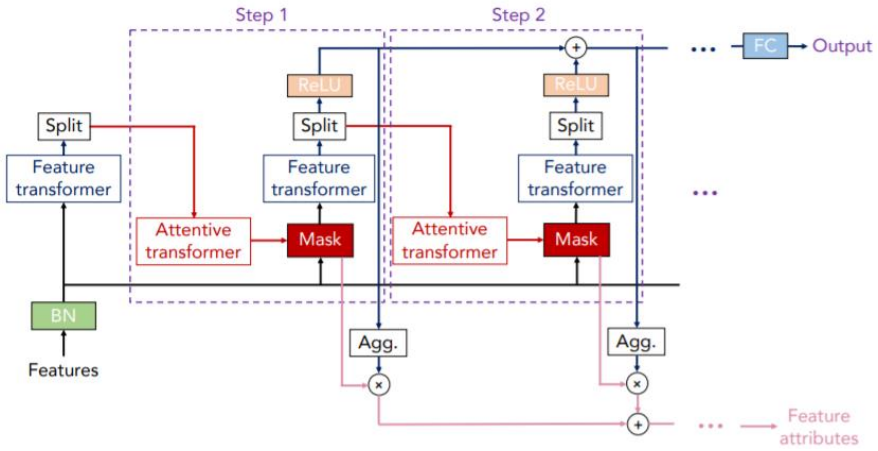
* 논문 인용

TabNet : Encoding architecture

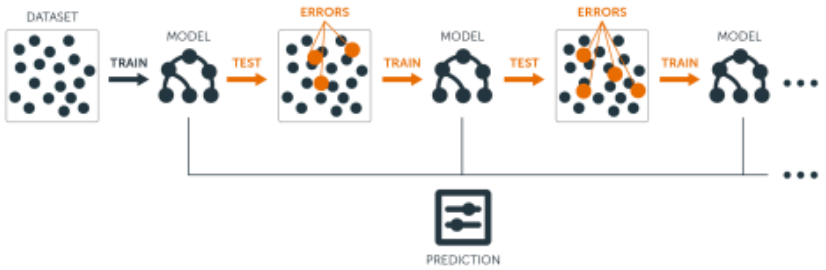
❖ 이전 단계 학습 결과가 다음 단계 Mask 학습에 영향을 주는 연결 구조

- Sequential Approach: 모델을 반복 연결하여 잔차를 보완하는 gradient boosting이 연상되는 구조
- Feature Selection: feature transformer와 attentive transformer 블록을 통과하여 최적 mask를 학습함

Tabnet Encoding architecture



Gradient Boost



* 논문 인용
* https://bradleyboehmke.github.io/HOML/gbm.html

TabNet : Encoding architecture

❖ Feature Transformer: 선택된 feature로 정확히 예측하기 위한 embedding 기능

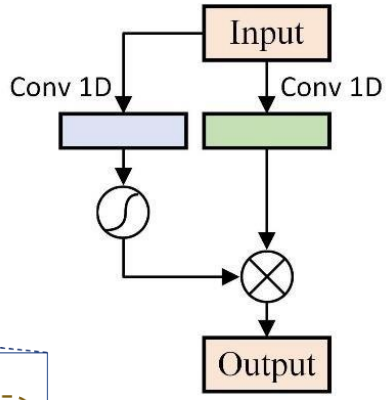
- Ghost Batch Norm(BN로 표기함)

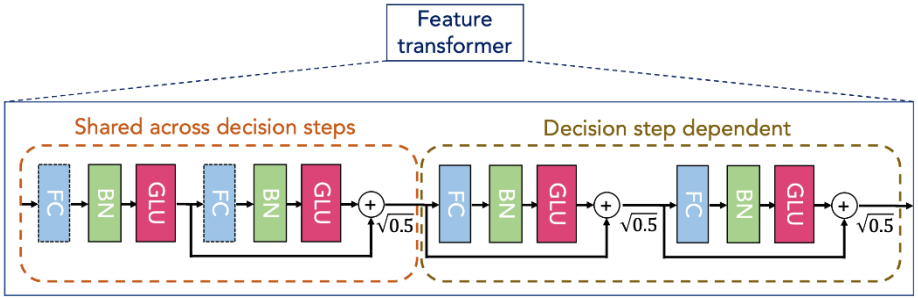
  : batch를 분할한 nano batch 사용으로 잡음 추가 → 지역 최적화 예방 → large batch size로 학습속도 향상

- Gated Linear Unit(GLU): 이전 layer에서 전달되는 정보 크기를 제어하는 역할

Ghost Batch Norm

Gated Linear Unit





* 논문 인용

TabNet : Encoding architecture

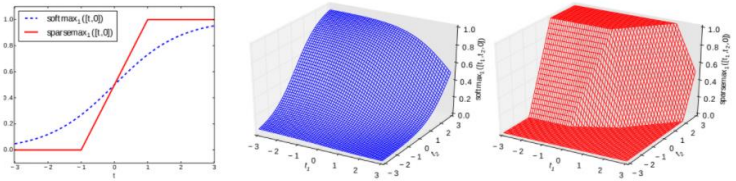
❖ Attentive Transformer: 변수 선택 기능

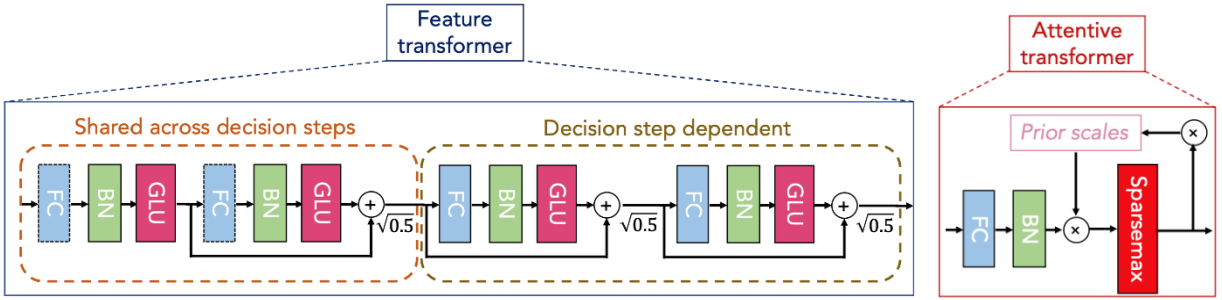
- Prior Scales: i번째 단계에서 변수의 중복 반영 여부를 결정하는 factor로써,
  선택된 변수의 반영률이 점차 낮아지는 측성
- Sparsemax: softmax 함수 대비 sparsity가 높은 함수로 attention layer 등에서 효과적임

Prior Scales

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

Sparsemax vs Softmax





* 논문 인용

TabNet : Encoding architecture

❖ Sparsemax 함수 실험

- 동일한 범위 내 서로 다른 등간격의 n개 숫자 배열에서 sparsemax 함수의 결과 비교
- 범위가 동일해도 데이터가 많을수록 0 전환 비율이 높아짐

Algorithm 1 Sparsemax Evaluation

Input: $\boldsymbol{z}$

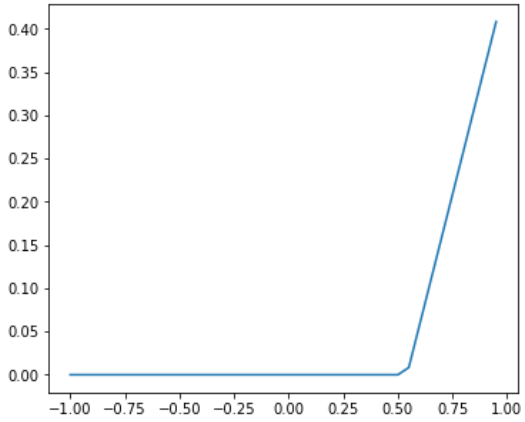
Sort $\boldsymbol{z}$ as $z_{(1)} \geq \ldots \geq z_{(K)}$

Find $k(\boldsymbol{z}) := \max \left\{ k \in [K] \mid 1 + k z_{(k)} > \sum_{j \leq k} z_{(j)} \right\}$

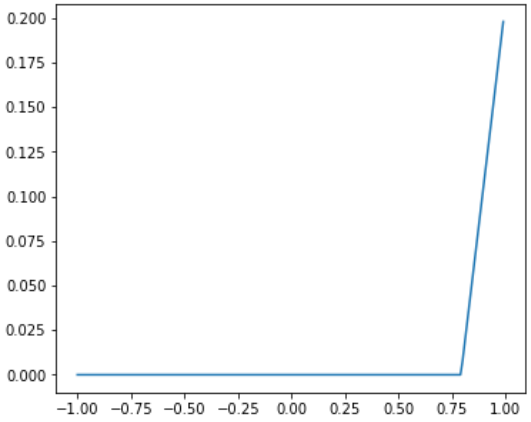
Define $\tau(\boldsymbol{z}) = \frac{(\sum_{j \leq k(\boldsymbol{z})} z_{(j)}) - 1}{k(\boldsymbol{z})}$

Output: $\boldsymbol{p}$ s.t. $p_i = [z_i - \tau(\boldsymbol{z})]_+$.

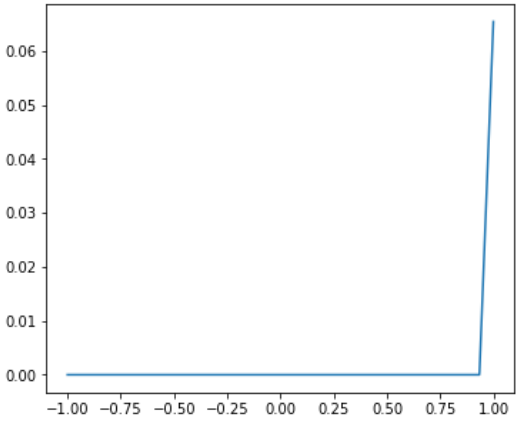
np.arange(-1, 1, 0.05)



np.arange(-1, 1, 0.01)



np.arange(-1, 1, 0.001)



TabNet : Encoding architecture

❖ Entmax 함수

- A sparse family of probability mappings and corresponding loss functions, generalizing softmax / cross-entropy

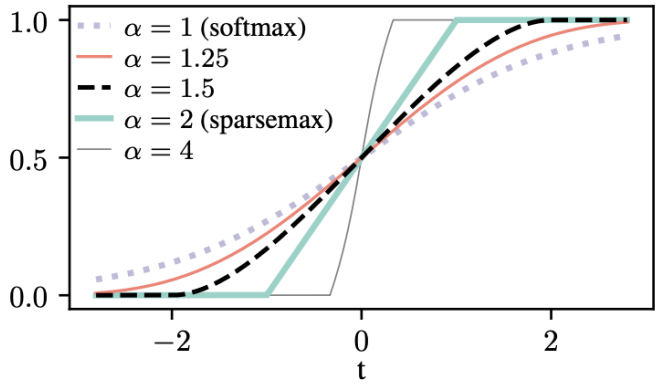


Figure 3: Illustration of entmax in the two-dimensional case α-entmax$([t, 0])_1$. All mappings except softmax saturate at $t = \pm 1/\alpha - 1$. While sparsemax is piecewise linear, mappings with $1 < \alpha < 2$ have smooth corners.

```
In [1]: import torch

In [2]: from torch.nn.functional import softmax

In [2]: from entmax import sparsemax, entmax15, entmax_bisect

In [4]: x = torch.tensor([-2, 0, 0.5])

In [5]: softmax(x, dim=0)
Out[5]: tensor([0.0486, 0.3592, 0.5922])

In [6]: sparsemax(x, dim=0)
Out[6]: tensor([0.0000, 0.2500, 0.7500])

In [7]: entmax15(x, dim=0)
Out[7]: tensor([0.0000, 0.3260, 0.6740])
```

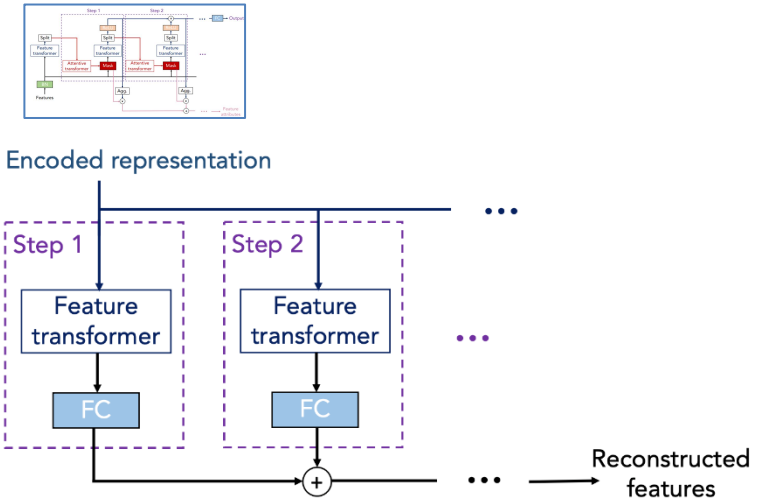
* https://housekdk.gitbook.io/ml/ml/tabular/tabnet-overview
* https://github.com/deep-spin/entmax

TabNet : Decoding architecture

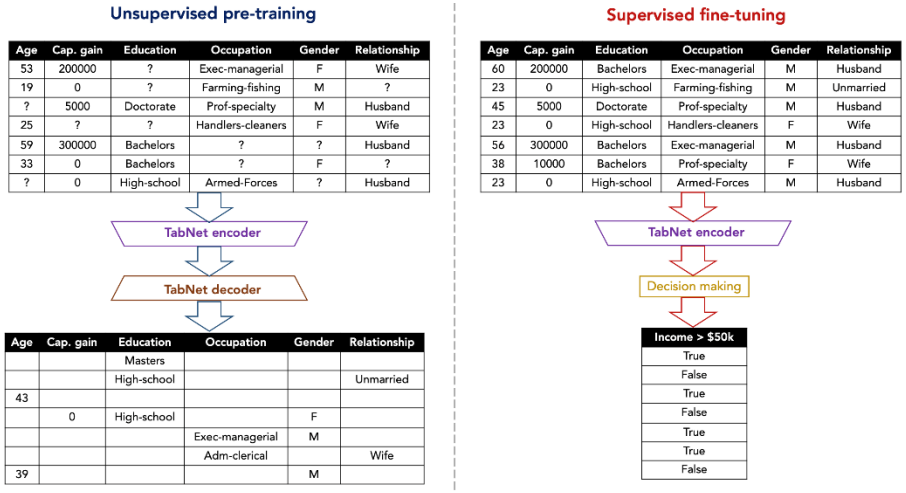
❖ Semi-supervised Learning

- 앞서 소개한 encoder에 decoder를 연결하면 autoencoder와 같은 자기 학습 구조를 생성할 수 있음
- 특정한 영역이 masking된 인코딩 데이터를 원본대로 복원할 수 있도록 학습
- 사전 학습을 통한 예측 성능 향상, 학습 시간 단축 및 결측치에 대한 보간 효과

Sparsemax 알고리즘



Sparsemax vs Softmax



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

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

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

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

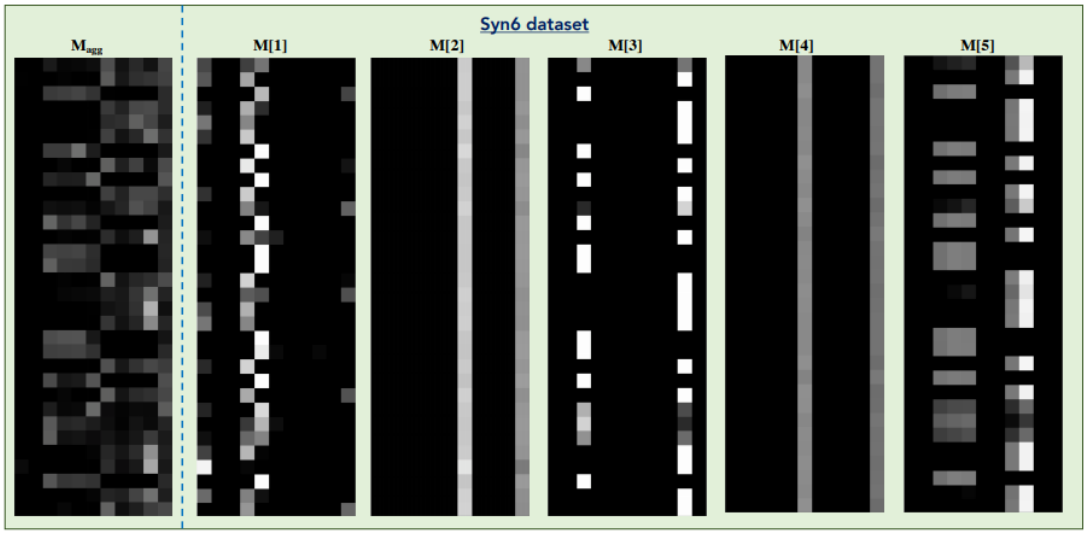
*논문 인용

TabNet : Interpretation

❖ Attentive Transformer의 Mask값을 활용한 변수 중요도 시각화

- M[i]는 모든 검증 데이터에 대해 각 attentive transformer 단계에서 mask 적용 후 활성화 비율을 표현하며 지역적인 특성을 확인할 수 있음
- M_{agg}는 모든 attentive transformer 단계의 활성화 비율을 결합한 것으로 글로벌 특성을 확인할 수 있음

Feature Importance Masks



* 논문 인용

TabNet : Performance

❖ 다수의 Tabular Dataset에서 성능 우세

- 분류(Forest Cover Type) 및 회기 문제(Sarcos)에서 tree모델 대비 성능 우세
- 10.5 M 사이즈의 Higgs Boson dataset에서는 pretraining의 성능을 추가 입증함

Higgs Boston dataset
(Dua and Graff 2017)

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	**61.37 ± 0.88**
10k	66.66 ± 0.88	**68.06 ± 0.39**
100k	72.92 ± 0.21	**73.19 ± 0.15**

Forest Cover Type dataset
(Dua and Graff 2017)

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

Sarcos
(Vijayakumar and Schaal 2000)

Model	*Test MSE*	*Model size*
Random forest	2.39	16.7K
Stochastic DT	2.11	28K
MLP	2.13	0.14M
Adaptive neural tree	1.23	0.60M
Gradient boosted tree	1.44	0.99M
TabNet-S	**1.25**	**6.3K**
TabNet-M	**0.28**	**0.59M**
TabNet-L	**0.14**	**1.75M**

* 논문 인용

TabNet : Limitation

❖ Tabular Data: Deep Learning is Not All You Need

- 동일한 데이터로 4개의 tabular data용 딥러닝 연구 간 성능 비교 결과, fine tuned XGBoost가 딥러닝 대비 준수한 성능을 나타냄
- 딥러닝 모델과 XGBoost의 앙상블 결합 시 가장 우수한 결과를 도출함

Test results on tabular datasets

Name	Rossman	CoverType	Higgs	Gas	Eye	Gesture	YearPrediction	MSLR	Epsilon	Shrutime	Blastchar
XGBoost	490.18	3.13	21.62	2.18	**56.07**	80.64	77.98	55.43	11.12	13.82	20.39
NODE	488.59	4.15	21.19	2.17	68.35	92.12	76.39	55.72	**10.39**	14.61	21.40
DNF-Net	503.83	3.96	23.68	**1.44**	68.38	86.98	81.21	56.83	12.23	16.80	27.91
TabNet	**485.12**	3.01	**21.14**	1.92	67.13	96.42	83.19	56.04	11.92	14.94	23.72
1D-CNN	493.81	3.51	22.33	1.79	67.90	97.89	78.94	55.97	11.08	15.31	24.68
Simple Ensemble	488.57	3.19	22.46	2.36	58.72	89.45	78.01	55.46	11.07	13.61	21.18
Deep Ensemble w/o XGBoost	489.94	3.52	22.41	1.98	69.28	93.50	78.99	55.59	10.95	14.69	24.25
Deep Ensemble w XGBoost	485.33	**2.99**	22.34	1.69	59.43	**78.93**	**76.19**	**55.38**	11.18	**13.10**	**20.18**
		TabNet			DNF-Net			NODE		New datasets	

* 논문 인용

요약

❖ 딥러닝은 제품 수준의 서비스에도 유리한 특성을 지님

- Real Data는 끊임없이 유입되고 변화함 → 한번의 학습으로 영원히 사용할 수 있는 모델은 없음
- 딥러닝의 pretraining, Incremental learning 특성은 지속 학습 가능한 측면에서 좋은 대안임
- 해석 난이도의 개선 필요

❖ Tabnet은 해석이 용이한 딥러닝 모델

- conventional DNN 블록 개념을 도입하여 tree 모델과 같은 해석 용이성은 제공함
- attentive transformer 블록 내에 sparsemax, prior scales를 활용하여 변수의 중복 사용을 제한함으로써 변수 마다 중요도를 학습할 수 있도록 고안

❖ No Free Lunch

- 모든 문제를 해결하는 만능 키는 없음(Deep Learning is Not All You Need 사례)
- Tabular Data에서 딥러닝 활용 가능성을 보여준 사례로 관련 연구를 지속 학습할 계획임

감사합니다