IGA Works BIG DATA COMPETITION

igaworks

contents

contents

01 분석 방향 및 시각화

분석 목적

클릭률(CTR) 예측

- 광고 관련 데이터
- 사용자 데이터

→ 주어진 광고를 클릭할 확률

클릭률 예측모델의 지향점

예측력

적정가격제시를통한
매출극대화

속도

AutoBidding시스템에적용될수있는신
속한분석

수집 정보 활용

안정적인결과를얻기위해
수집된데이터200%사용하기

igaworks

01 분석 방향 및 시각화

분석 목적

클릭률 예측모델의 지향점

예측력

적정가격제시를 통한
매출극대화

속도

Auto Bidding 시스템에
적용될 수 있는 신속한 분석

수집 정보 활용

안정적인 결과를 얻기 위해
수집된 데이터 200% 사용하기

본 분석의 기조

데이터에 적합한 모델 선택

데이터 차원 축소

정보 손실 유의

igaworks

01 분석 방향 및 시각화

데이터 특성

주어진 데이터셋

데이터셋
Train: 각 bidding 별 정보 (5500000개)
Audience: 소비자 별 정보

종속변수
Click: 클릭했을 경우 1, 안 했을 경우 0

Click	사용자 정보 유	사용자 정보 무	Total
1	9.7%	9.2%	10%
0	90.3%	90.8%	90%
Total	2,232,520	3,267,480	5,500,000

- 전체 5,500,000개 데이터 중 40%에 소비자 정보가 있다.
- 소비자 정보 유무에 따른 클릭률 차이가 크지 않다.

전체 데이터 중 click이 1인 경우가 10%이므로 unbalanced 데이터이다.

igaworks

01 분석 방향 및 시각화

데이터 특성

Train 데이터

Train 데이터

각 bidding 별 정보.
예) 광고 타입, 광고주 아이디, 미디어 아이디, 기기 통신사, 광고 아이디, 매체사 아이디, 기기 연결방식 등

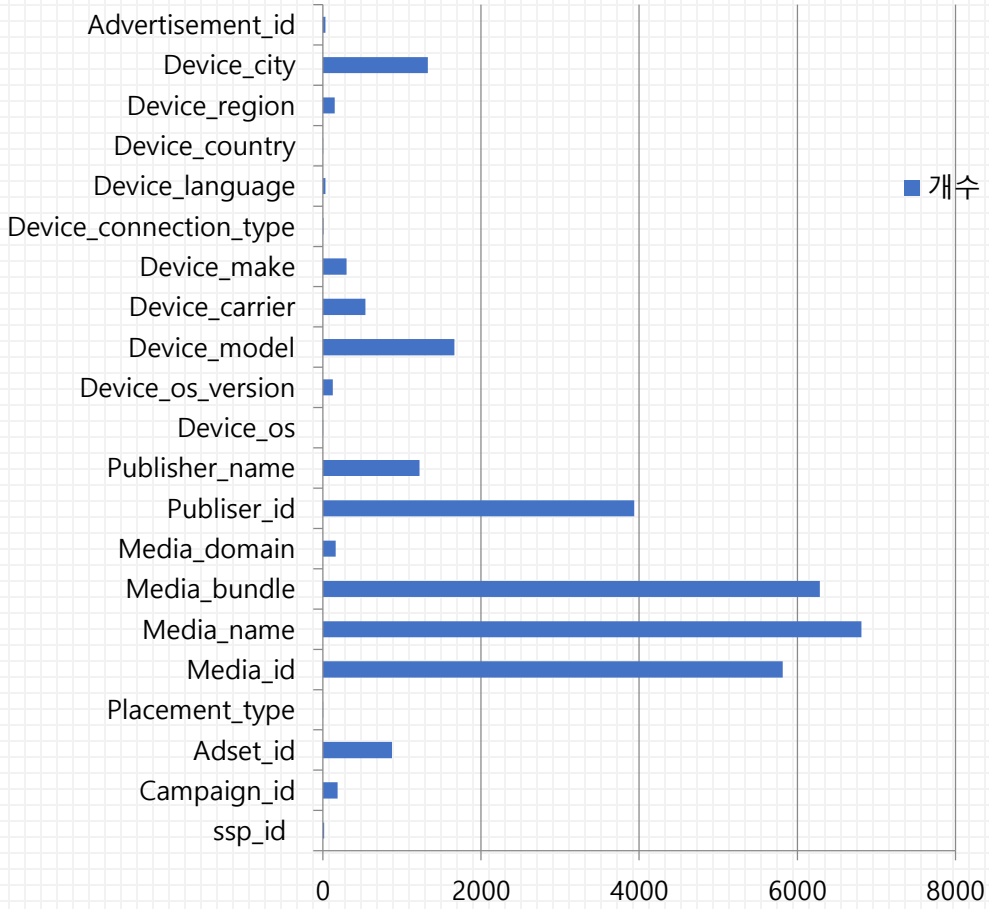
설명 변수 개수

총 22개

- media_name에는 총 6810개의 범주가 있다.
- 소비자 정보 유무에 따른 클릭률 차이가 크지 않다.

범주형 변수를 one-hot-encoding했을 때,
데이터의 대다수가 0인 Sparse 데이터이다.

변수별 범주 개수



igaworks

01 분석 방향 및 시각화

데이터 특성

Audience 데이터

Audience 데이터
각 소비자 별 정보.
예) 성별, 연령, 결혼 여부, 사용자가 앱에 부여한 평점

변수 개수
총 8개

cate_code 변수
사용한 앱과 평점에 대한 변수이다.
A:1이라면 A앱에 대하여 1점을 부여한 것이므로 점수와 앱 이름을 분리한다.
이를 one-hot-encoding한 결과는 우측과 같다.

cate_code 변수를 one-hot-encoding했을 때,
데이터의 대다수가 0인 Sparse 데이터이다.

One-hot-encoding 결과 예시

App1	App2	App3	.	.	App_242
0	0	5			0
0	0	0			1
0	0	0			0
0	0	0			0
0	0	0			0
0	0	0			2
5	0	0			0
0	3	0			0
0	0	4			0
0	0	0			0
0	0	0			0
0	4	0			0
0	0	0			0
2	0	0			1
0	0	0			0
0	3	0			0
0	0	4			0
0	0	0			0
0	0	0			0
0	0	0			2

01 분석 방향 및 시각화

데이터 특성에 따른 전처리와 모델링

Sparse Data

차원이 크며, 대부분의 값이 0이다.

전처리

Feature Selection
- 각 변수 내 범주를 줄여준다

모델

xDeepFM (extreme Deep Factorization machine)
- 범주형과 연속형 데이터를 모두 반영
- Train의 AutoEncoder결과값을 x로 둠
- Logloss값을 추출

Audience data의 cate_code

사용한 앱이 컬럼으로 평점이 값으로 들어 있다.

전처리

1. 평점 존재 / 평점 존재 하지 않음
2. 앱에 대한 좋은 평가 / 낮은 평가 혹은 평가하지 않음

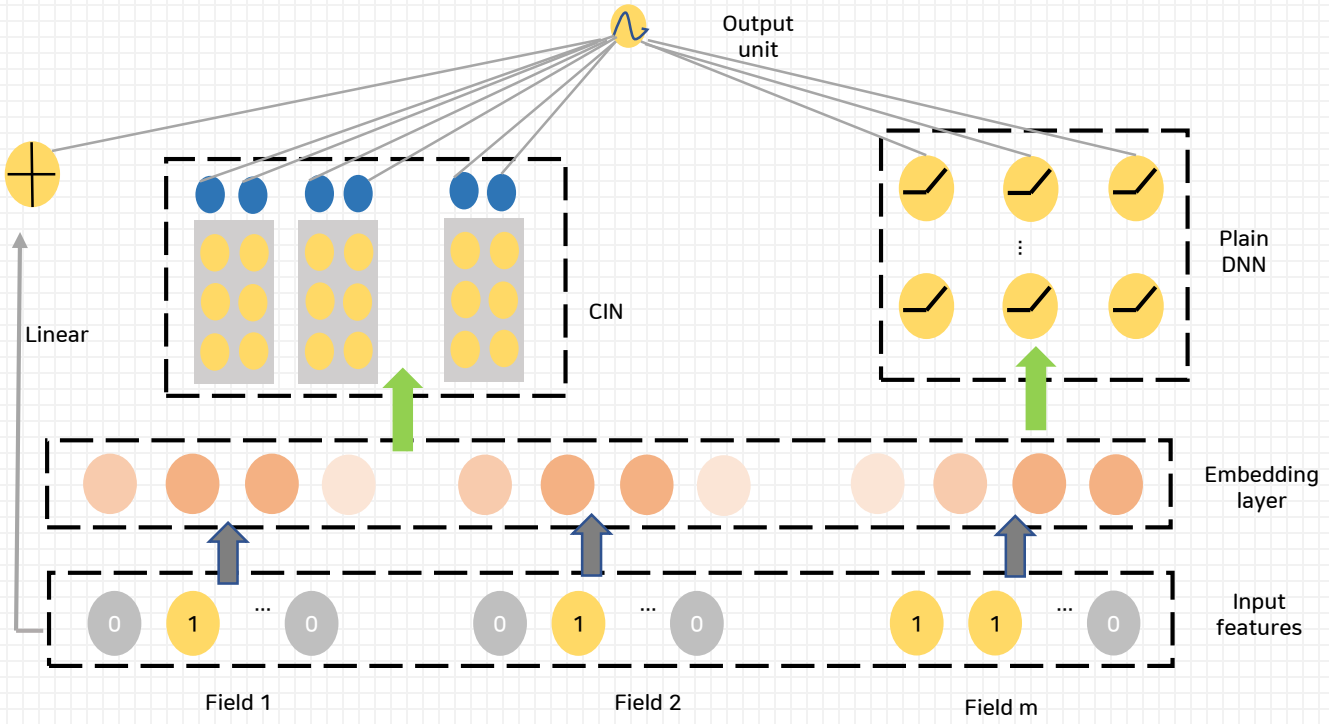
모델

AutoEncoder
- 모델 예측력이 높을 때, hidden layer가 sparse데이터의 정보를 압축적으로 가지고 있다고 가정

igaworks

01 분석 방향 및 시각화

모델링 - xDeepFM



xDeepFM

xDeepFM이란?

- extreme Deep Factorization Machine

  deepFM에서 개선된 모델로 명시적인 방식과 벡터단위로 feature interaction을 계산한다

장점

1) 범주형과 연속형을 다룰 수 있다
2) 다차원의 sparse data에 대해서도 feature interaction 계산이 가능하다

Why xDeepFM?

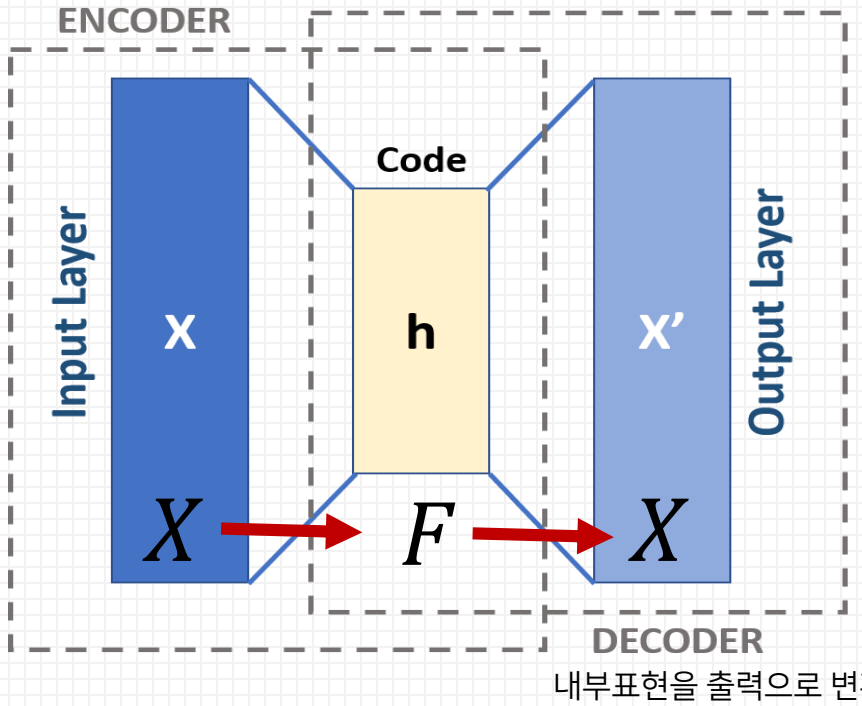
우리 데이터는 범주형과 연속형을 모두 가진 다차원의 sparse data이므로 기존의 FM, deepFM에 비해서 노이즈에 강하고 성능이 좋은 xdeepFM을 사용

igaworks

01 분석 방향 및 시각화

모델링 - AutoEncoder

입력을 내부 표현으로 변환



내부표현을 출력으로 변환

Algorithm

1. 입력값 x로부터 중요한 정보를 압축해 놓은 h값 찾기
2. h값을 이용해서 또 다시 x'값을 예측
3. 정답인 입력값 x와 비교

AutoEncoder

AutoEncoder란?

✓ 입력값과 출력값을 같게 만든다

장점

1) **비지도학습 + 지도학습**
   y를 설정하지 않고도 x를 예측할 수 있다 ➔ 비지도학습
   예측한 x'값을 실제 x값과 비교할 수 있다 ➔ 지도학습
2) **data-specific하다**
3) **입력 데이터에 대하여 자동적으로 학습**

Why AutoEncoder?

속도를 줄이기 위해 데이터 차원을 줄이면서 동시에 정보 손실을 최소화 할 수 있는 방법이기 때문에 우리 데이터에 적합함

igaworks

01 분석 방향 및 시각화

변수 특성

	variable	relative_importance	scaled_importance	percentage
0	publisher_name	382205.562500	1.000000	0.261094
1	adset_id	197057.015625	0.515579	0.134614
2	publisher_id	121026.023438	0.316652	0.082676
3	device_city	121011.304688	0.316613	0.082666
4	device_model	106226.843750	0.277931	0.072566
5	device_ifa	104630.531250	0.273755	0.071476
6	ssp_id	69146.375000	0.180914	0.047236
7	device_carrier	64813.175781	0.169577	0.044275
8	device_os_version	59275.515625	0.155088	0.040493
9	campaign_id	56900.812500	0.148875	0.038870
10	media_bundle	38511.964844	0.100762	0.026308
11	media_name	34882.199219	0.091266	0.023829
12	device_region	26734.177734	0.069947	0.018263
13	media_id	16268.852539	0.042566	0.011114
14	advertisement_id	16230.002930	0.042464	0.011087
15	placement_type	15265.708008	0.039941	0.010428
16	device_make	13872.188477	0.036295	0.009476
17	device_connection_type	12142.248047	0.031769	0.008295
18	device_language	6166.703125	0.016135	0.004213
19	media_domain	1422.255615	0.003721	0.000972
20	device_os	72.542709	0.000190	0.000050

랜덤포레스트 변수중요도

- 범주가 많은 변수들이 대부분 상위권에 있다
- 유사도가 높은 변수들이 존재할 경우,
  중요한 변수들의 중요도가 일부 과소평가 될 수 있다
- media_bundle, media_name, media_id의 경우,
  매우 유사한 의미를 갖는 변수들이며, 범주의 개수가 많은 편임에도 변수중요도가 높지 않다

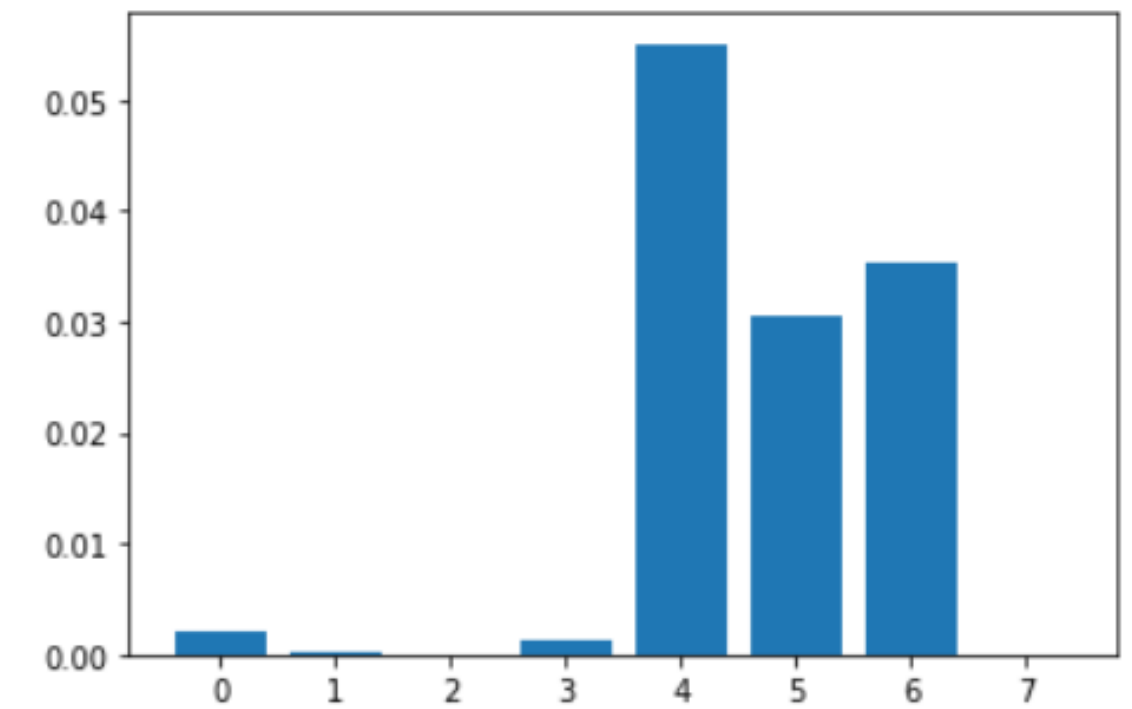
범주 개수에 비해 변수중요성이 낮으며 media_id와 유사한 media_bundle과 media_name은 분석에서 제외한다

igaworks

01 분석 방향 및 시각화

변수 특성

Train/Test data – device_connection_type



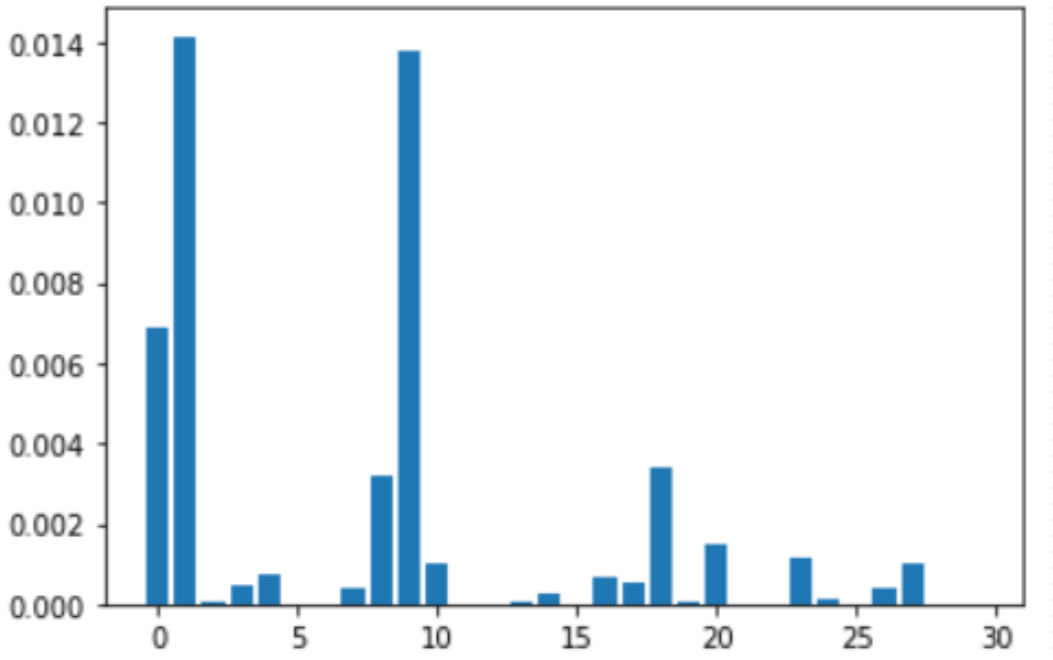
ID	importance
2xyglifxY3C	0.002167
4sFc3rM27t	0.000198
55rBbmtMui	0.000000
6xAYOHll8l	0.001321
Hx7e3TE5mu	0.055077
WCK2G73H3A	0.030665
aEmZFzgDfq	0.035373
aze5oiXm1V	0.000000

igaworks

01 분석 방향 및 시각화

변수 특성

Train/Test data – advertisement_id



ID	importance
7dyzy9aZoJ	0.014139
K67ZwviDgnB	0.013796

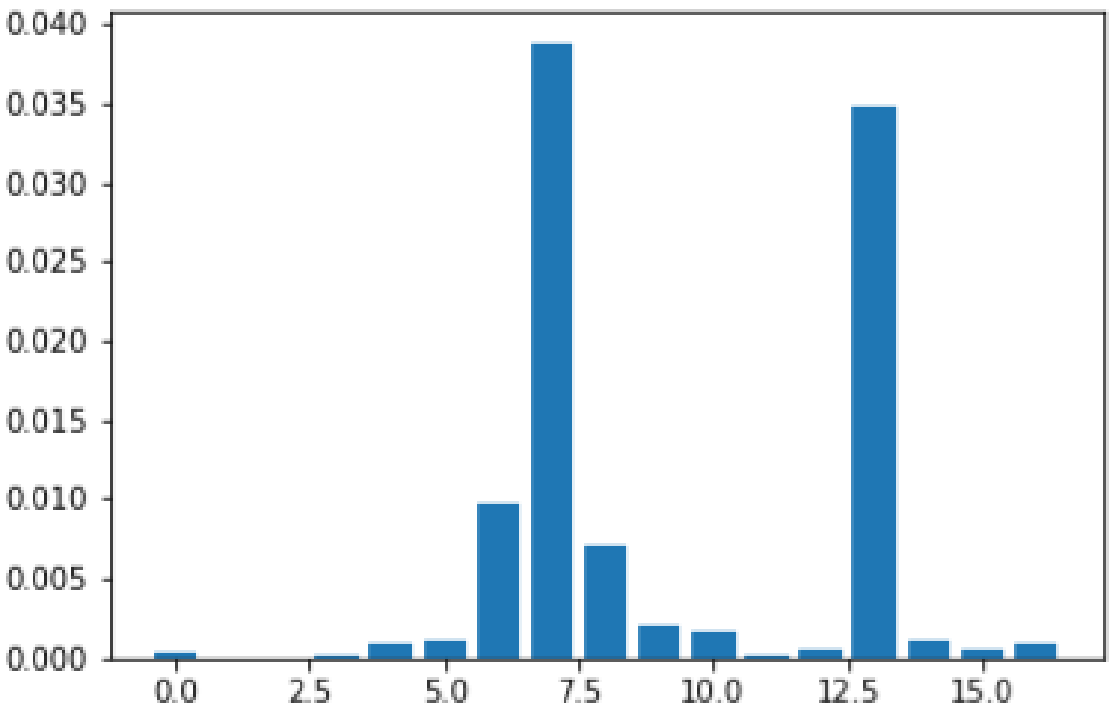
변수 내 범주별 발생횟수

- 범주별 발생횟수 차이가 현저하다
- 발생횟수가 적은 범주들은 영향력이 적다

igaworks

01 분석 방향 및 시각화

변수 특성



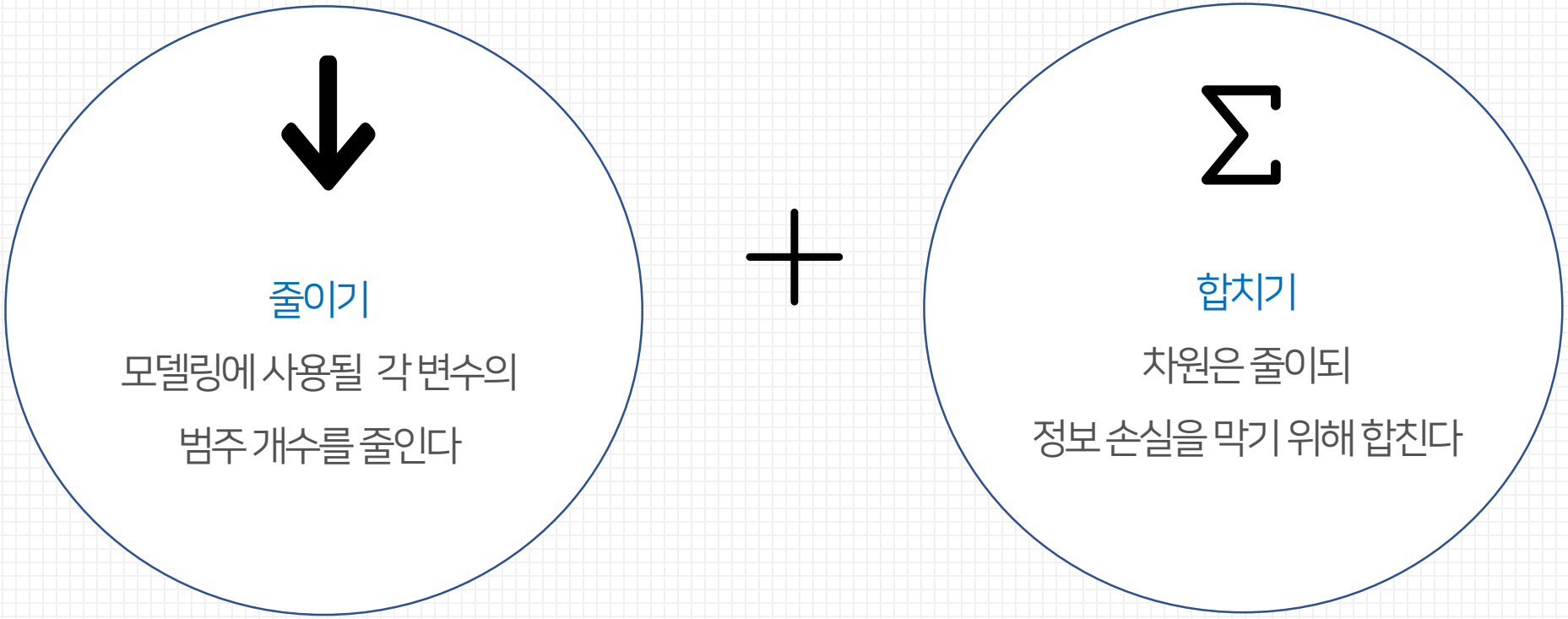
변수 내 범주 별 중요도

- 범주별 중요도 차이가 현저함
- 발생횟수가 적은 범주들은 중요도도 적은 경우가 많음
- 여러가지 구하는 방법 중 information gain으로 산출
  (Chi-square로 구한 경우, outlier빼고 다 거의 0임)

igaworks

contents

02 전처리 방향



우리의 데이터는 범주형 자료가 대부분, one-hot-encoding시 차원이 크게 늘어나는 문제 발생.
따라서 줄이기와 합치기 전략으로 전처리를 진행함

igaworks

02 전처리

줄이기 | 합치기

1 Train/Test data

변수	변수
ssp_id	device_os
campaign_id	device_os_version
adset_id	device_model
placement_type	device_carrier
media_id	device_make
media_name	device_connection_type
media_bundle	device_country
media_domain	device_language
publiser_id	device_region
publisher_name	device_city
device_ifa	advertisement_id
	event_datetime

01 media_name, media_bundle
: media id와 유사한 의미를 가지는 변수이며 범주의 개수가 많음에도 변수 중요도가 높지 않으므로 제거

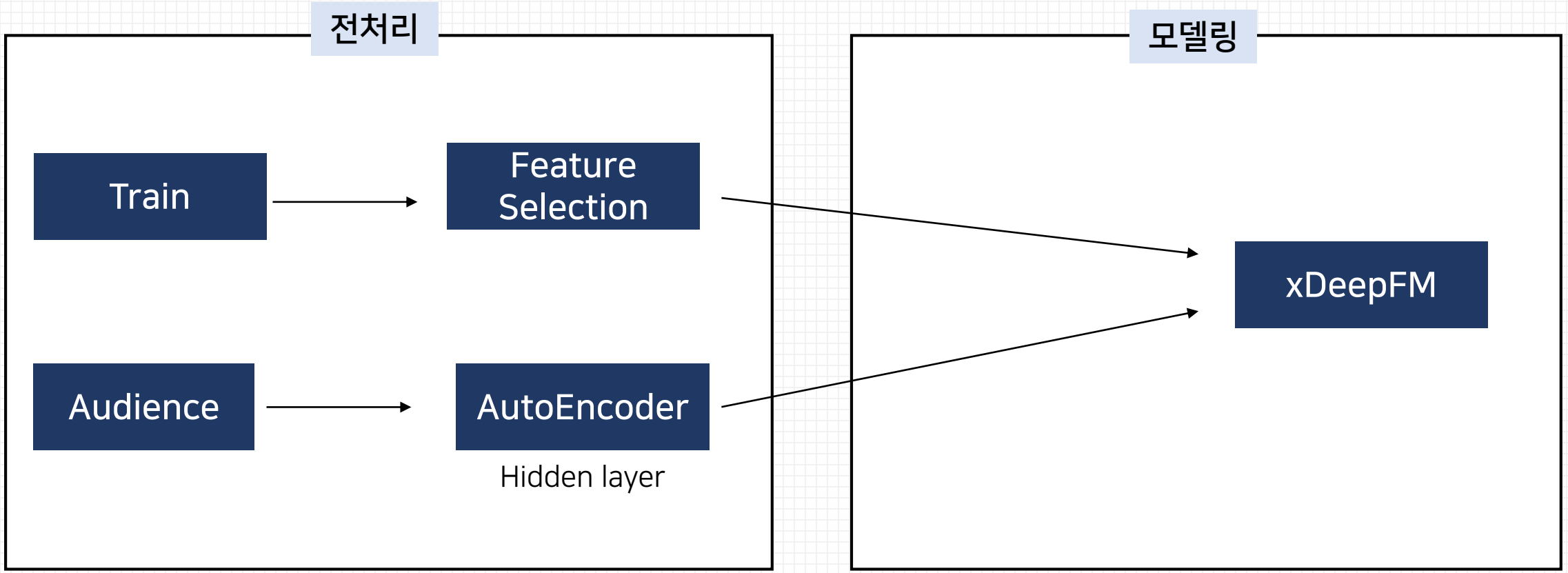
02 device_country
: 대한민국(1개 국가)에 대한 정보, 모두 같으므로 유의미하지 않다고 판단하여 제거

03 event_datetime
: 년, 월, 일, 분, 초 의 정보는 필요 없다고 판단하여 시간(hour)과 요일(day of week)만 사용

igaworks

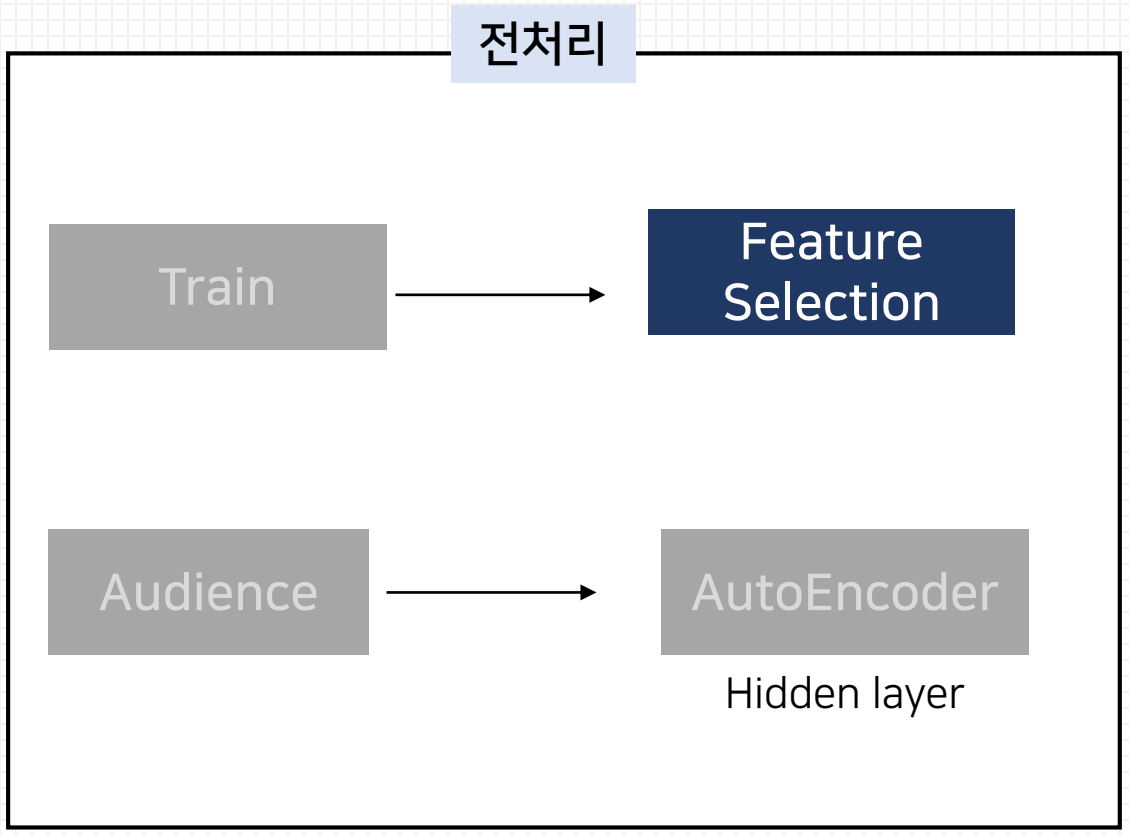
02 전처리 방향

줄이기


전처리
Train
Feature Selection
Audience
AutoEncoder
Hidden layer
모델링
xDeepFM

02 전처리 방향

줄이기



Feature Selection

- Information Gain
- 발생횟수

모델링에 사용할 변수의 범주를 줄이기 위해 information gain과 발생횟수를 고려하여 feature selection 후,
선택하지 않은 범주는 묶어서 기타 처리 하였다.

igaworks

02 전처리 방향

줄이기 | 합치기

Why Information Gain?

Feature Selection 방법

1	Chi-squared	Y에 대하여 x(각 설명 변수)가 얼마나 독립인지 확인
2	Random forest	랜덤포레스트 모델링 후 각 변수가 트리빌딩에 얼마나 영향을 주었는지, 변수가 고려되었을 때 평가지표가 얼마나 개선되었는지 비교
3	Information gain	얼마나 y예측에 도움이 되는지

igaworks

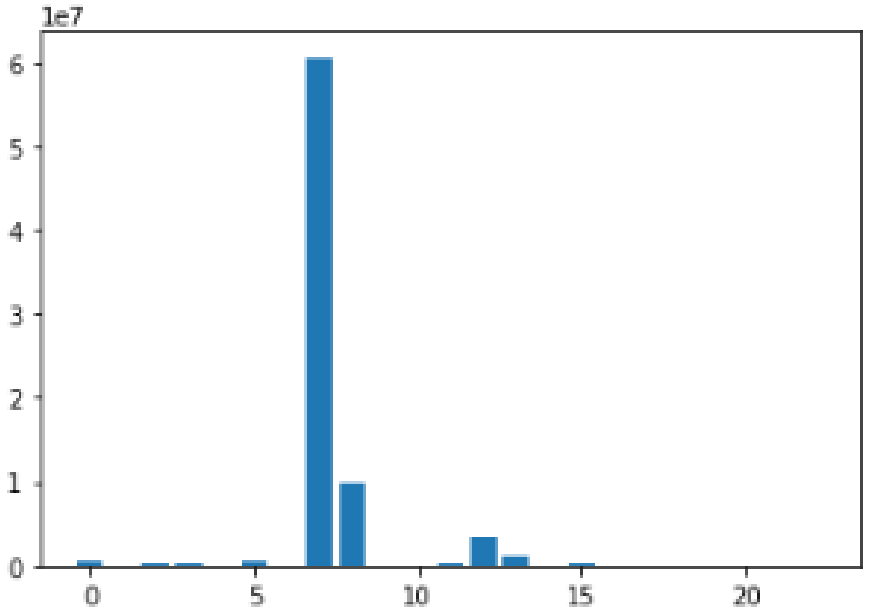
02 전처리 방향

줄이기 | 합치기

1 Chi-squared

y에 대하여 x(각 설명 변수)가 얼마나 독립인지 (종속인 변수일수록 더 좋음)

```
Feature 0: 0.000000
Feature 1: 0.000000
Feature 2: 0.000000
Feature 3: 0.000000
Feature 4: 0.000000
Feature 5: 0.000000
Feature 6: 0.000000
Feature 7: 0.000000
Feature 8: 0.000000
Feature 9: 0.000000
Feature 10: 0.000000
Feature 11: 0.000000
Feature 12: 0.000000
Feature 13: 0.000000
Feature 14: 0.000000
Feature 15: 0.000000
Feature 16: 0.000000
Feature 17: 0.000000
Feature 18: 0.000000
Feature 19: 0.000000
Feature 20: 0.000000
Feature 21: 0.000000
Feature 22: 0.039949
```



chi-squared test 결과 대부분 0이 나왔고,
그래프 상으로도 함께 확인해서 낮은 값만 제거하려 하였으나 거의 모든 값이 낮게 나옴
따라서 chi-squared는 사용 불가

igaworks

02 전처리 방향

줄이기 | 합치기

2 RandomForest

RadomForest 모델 적합 후, variable importance를 확인하여 importance가 낮은 것을 제거하려고 하였으나,
범주의 개수가 많으면 importance가 올라가는 추세를 보임
importance가 낮은 것을 제거할 경우 정보 손실의 우려가 있어 변수를 제거하는 것은 부적합함
따라서 RandomForest를 사용하는 방법도 사용 불가

3 Information Gain

y예측에 얼마나 기여하는가를 나타냄
따라서 Information Gain이 낮은 값을 제거하는 것은 y예측에 도움이 안되는 것을 빼내는 것

information gain과 발생횟수를 고려하여 feature selection 진행

igaworks

02 전처리

줄이기 | 합치기

1 Train/Test data – Feature Selection

변수	범주 개수(전)	범주 개수(후)
ssp_id	17개	15개
campaign_id	186개	90개
adset_id	872개	231개
placement_type	4개	4개
media_id	5815개	5815개
media_domain	162개	7개
publiser_id	3939개	3939개
publisher_name	1222개	1222개
advertisement_id	30개	28개

변수	범주 개수(전)	범주 개수(후)
device_os	2개	2개
device_os_version	125개	21개
device_model	1664개	80개
device_carrier	536개	40개
device_make	299개	9개
device_connection_type	8개	6개
device_language	33개	21개
device_region	148개	27개
device_city	1326개	138개

아래의 x변수를 제외하고 나머지는 feature selection을 통하여 범주의 개수를 줄였다

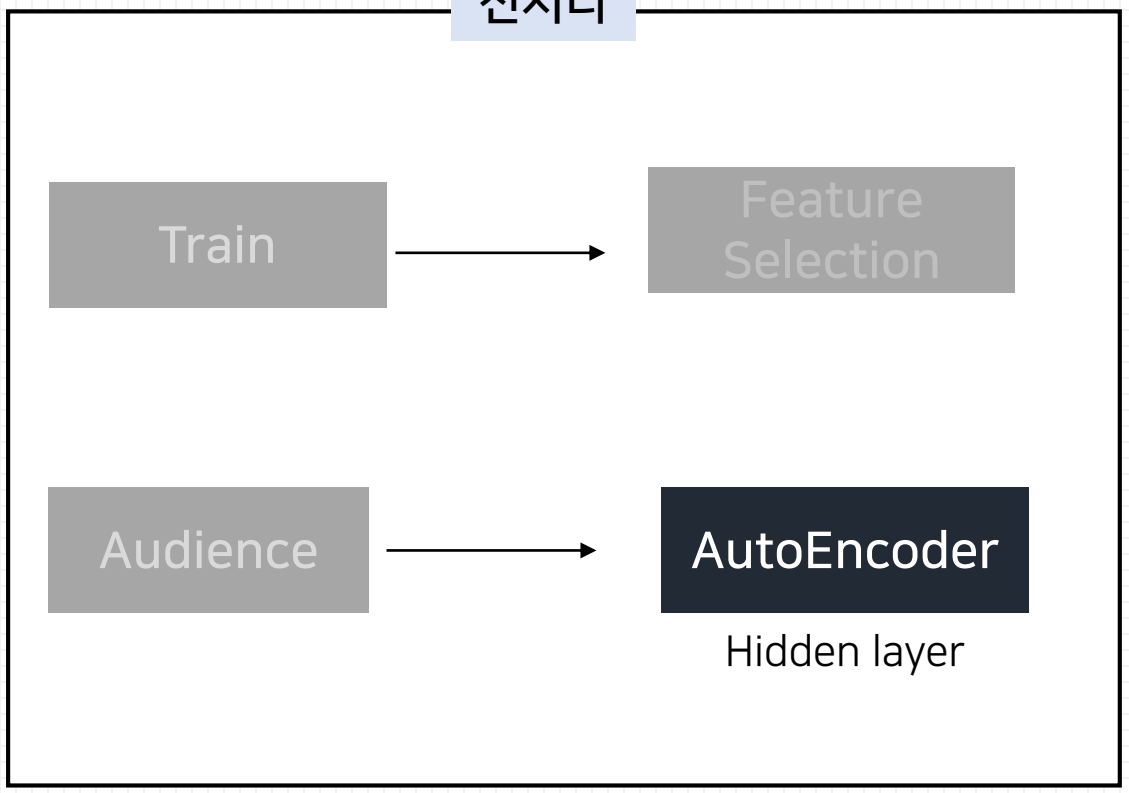
- media_id, publisiher_id, publisher_name: 범주 개수가 너무 많으므로 불가능, 원시데이터 사용
- placement type, device_os: 이미 범주 개수 적으므로 할 필요 없음, 원시 데이터 사용

igaworks

02 전처리 방향

줄이기

전처리



AutoEncoder

- 앱의 종류 약 242개에 대하여 사용자가 평점 부여
- 응답 유무에 대한 변수
  부여한 점수에 대한 변수
- 두 가지 변수에 대하여 각각 AutoEncoder를 사용하여 5차원으로 축소

igaworks

02 전처리

줄이기

합치기

전체 데이터 중 25%를 sampling한 후, train set에 있는 device_ifa중 누락된 것만 추가하여 데이터셋 완성

변수명	INFO
device_ifa	Train/ Test data와 합치는 key
age	모델링에 사용
gender	
marry	
install_pack	유의미하지 않은 변수라고 판단
cate_code	추가 전처리 필요
predicted_ house_price	NA가 너무 많으므로 제거

2 Audience data

cate_code전처리

- A : 1 이라면 카테고리 A에 대하여 1점을 부여한 것으로 카테고리와 점수를 split
- - 응답 유무를 cate_answer 컬럼으로 0, 1 처리
  - 부여한 점수에 따라 cate_like 컬럼으로 무응답, 1,2,3점은 0, 4,5점은 1로 처리
- cate_answer, cate_like 컬럼 각각 오토인코더를 사용하여 5개의 컬럼으로 합쳐준다 (92%~93% 정도의 accuracy)

igaworks

02 전처리

최종데이터	
Audience data	**Train/Test data**
age gender marry cate_answer1~5 cate_like1~5	column별 feature selection hour(시간) day(요일)

NA 처리

- Audience data와 Train/ Test data는 'device_ifa' 변수를 사용하여 각각 merge함
- Audience data에는 있지만 Train / Test data에는 없는 cate_answer와 cate_like1~5의 NA는 원시데이터에서 category에 대하여 모두 0을 찍은 device_ifa의 오토인코더 결과 값으로 채워넣었다
- gender, marry, age의 경우 각각 'U', 'C', 'old'로 새로운 범주를 만들어 NA값을 채워주었다

igaworks

contents

01 분석 방향

02 전처리

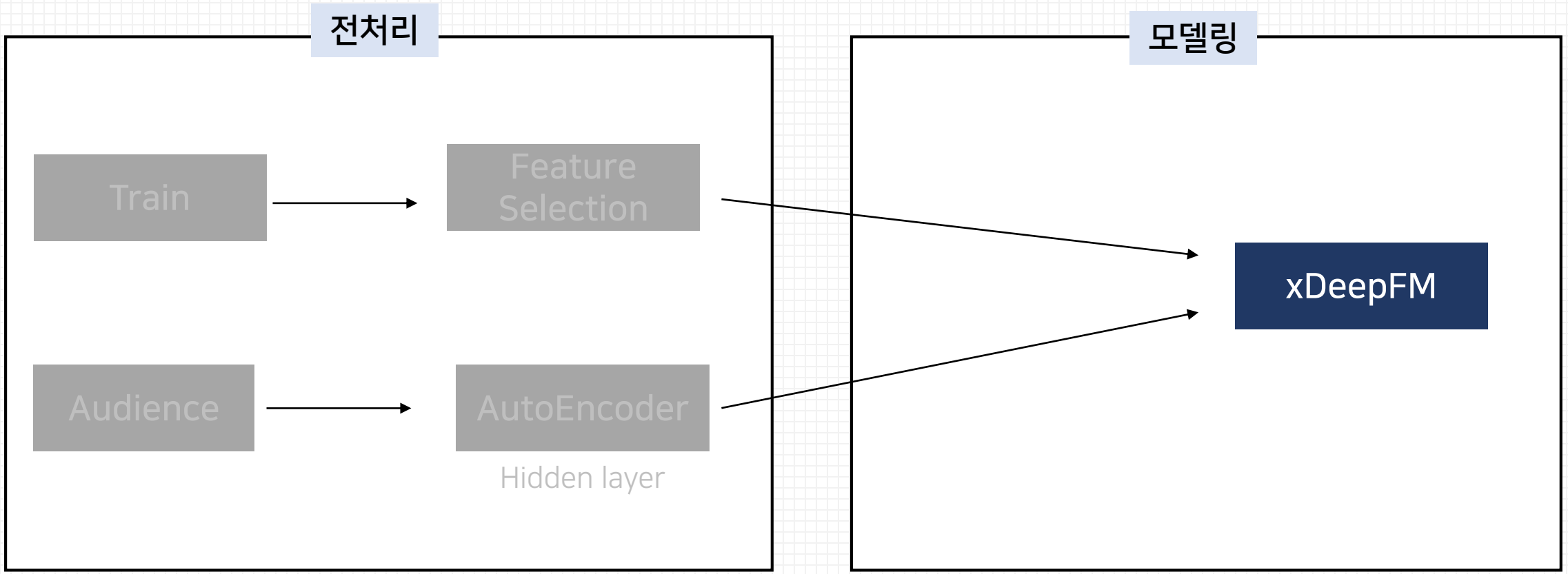
03 모델링

- xdeepFM

04 총정리

02 전처리 방향

줄이기



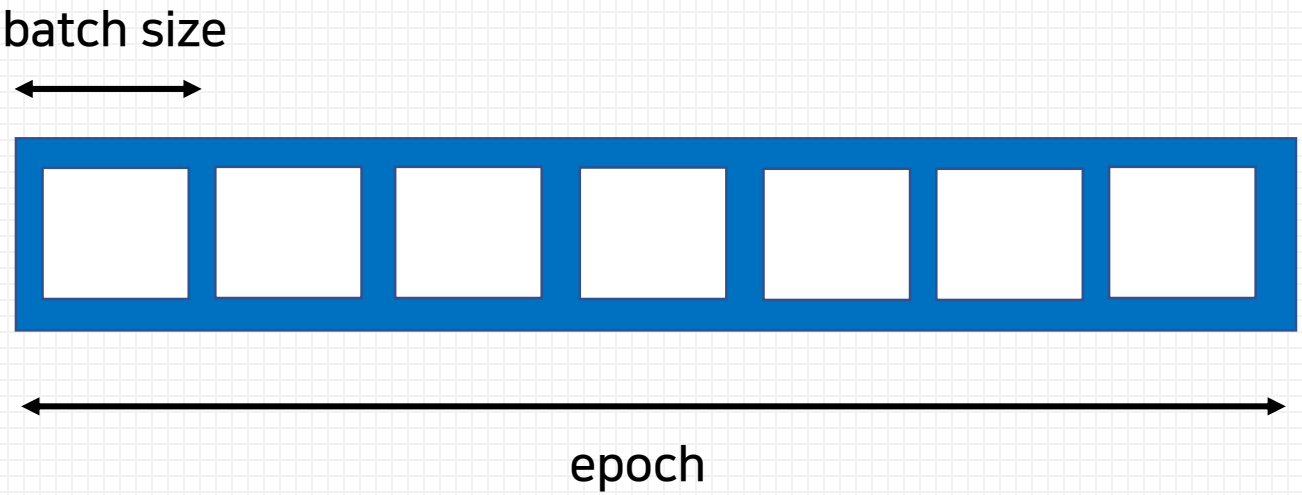
igaworks

03 모델링

고차원의 spare 데이터에 강한 딥러닝 기반의 xDeepFM을 최종 모델로 선택

xDeepFM

- extreme DeepFM (DeepFM에서 기원함)
- 명시적인 방법, 벡터 단위로 피처 상호작용 산출
- 고차원 수준의 sparse feature를 학습 시킬 수 있다
- 범주형, 연속형 자료를 모두 다룰 수 있다



DeepFM

- Linear part, FM layer, Hidden layer(DNN)으로 구성
- 암시적인 방법, 비트 단위로 피처 상호작용 산출

- xDeepFM은 DNN 기반의 모델
- batch size와 epoch를 튜닝하면서 최적의 파라미터 조합 찾음

igaworks

03 모델링

batch_size	epoch	train_log_loss	validation_log_loss	test_log_loss
128	4	0.2445	0.2452	0.237
128	10	0.2424	0.2435	0.2361
128	13	0.2451	0.2494	0.2364
128	15	0.2421	0.2446	0.2367
256	10	0.2430	0.2499	0.2354
256	15	0.2380	0.2408	0.2347

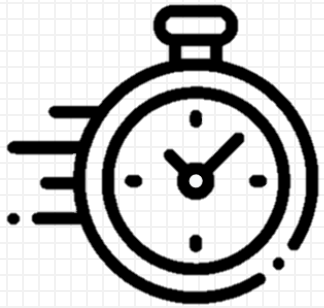
Grid Search를 통하여 최적의 파라미터를 찾음
test_log_loss만 비교하면 batch size 256, epoch 10이 가장 좋으나, 오버피팅 되는 경향이 있어,
최종 모델은 batch size 128 epoch 10으로 결정

	최종 모델
batch size	128
epoch	10
리더보드	0.24834

igaworks

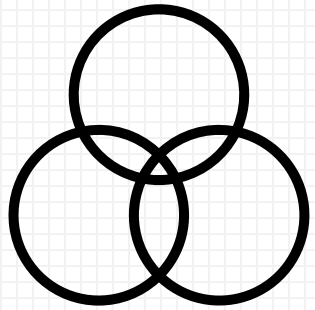
contents

04 총정리



주어진 데이터의 특징에 최적화된 모델을 사용하여
예측 시간 단축

고차원의 범주형 데이터의 차원축소와
정보손실 최소화를 동시에



다양한 경우의 수를 조합하여
보다 정교한 모델링 진행

igaworks

Thank you :)