Pseudo Lab Causal Inference Lab

인과추론과 실무 : 10. 지역 실험과 스위치백 실험

가짜연구소 인과추론팀

발표자 : 최지환

Causal Inference Lab
Pseudo Lab

0. 들어가며

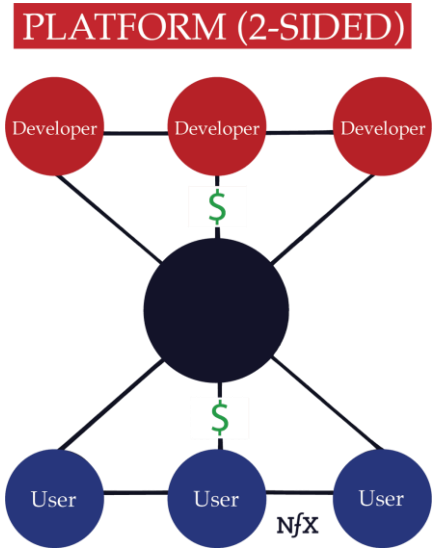
네트워크효과

Two-sided network effect

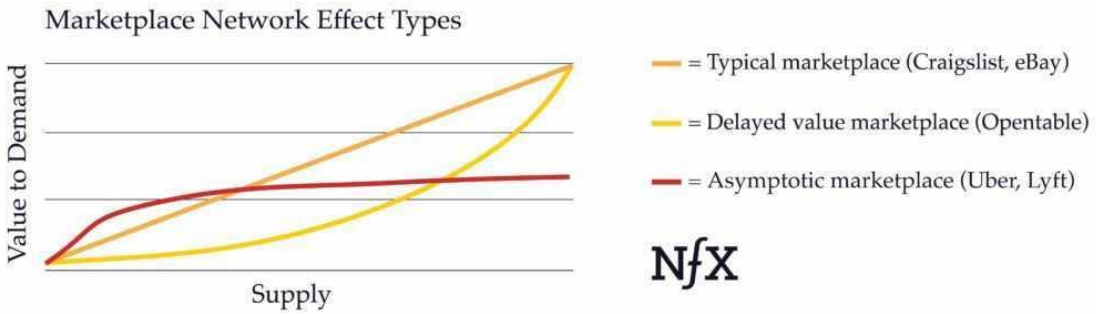
BOB METCALFE : 네트워크의 규모가 커지면 비용은 직선적으로 늘지만, 그 가치는 기하급수적으로 증가한다는 법칙

양면 플랫폼 시장

수요 공급이
하나의 플랫폼 내에
서 조절된다



Uber나 Lyft와 같은 자동차 공유 회사에서 볼 수 있다.
요컨대 공급자(운전자)가 많으면 대기 시간이 단축되어
승객에게 많은 혜택을 준다.



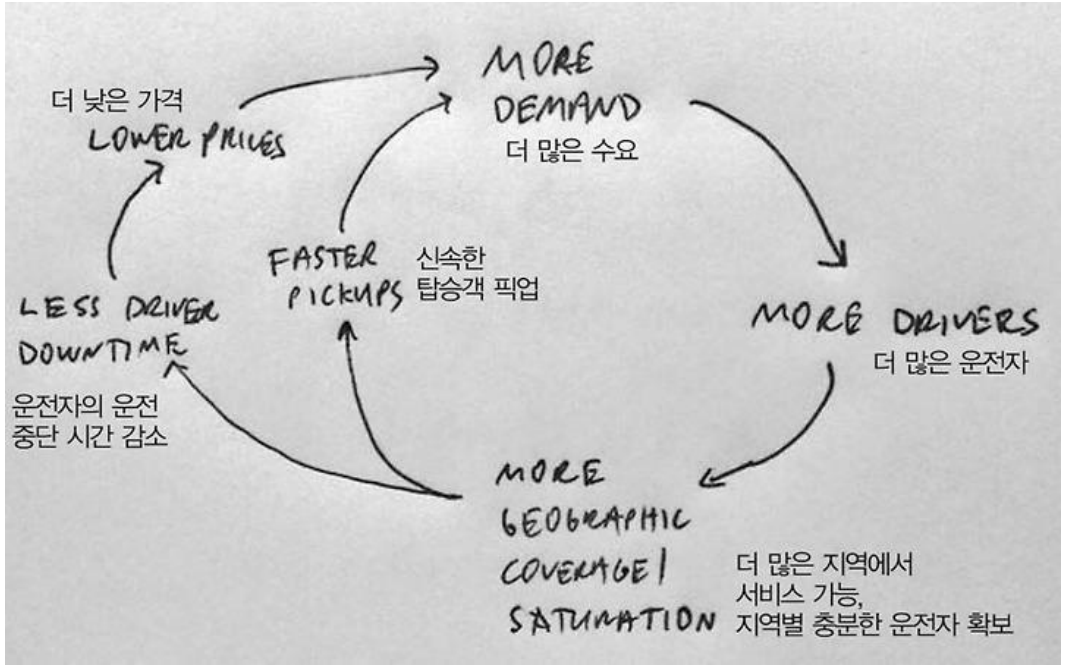
Source : https://www.mimul.com/blog/network-effects/

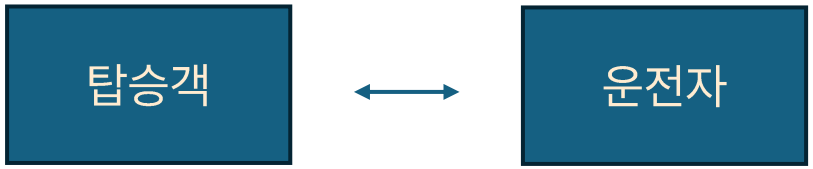
네트워크효과

Two-sided network effect

BOB METCALFE : 네트워크의 규모가 커지면 비용은 직선적으로 늘지만, 그 가치는 기하급수적으로 증가한다는 법칙

우버





탑승객이 운전자를 끌어들이고,
운전자가 탑승객을 끌어들이는 구조

탑승객의 수에 비해 지나치게
운전자가 많아져 중단시간이 늘어날 때,
운전자 수의 비해 지나치케
탑승객이 많아져 대기시간이 길어질 때

Source : 책_플랫폼 레볼루션 P.76

스필오버 효과

Spillover EFFECT

특정한 현상이 다른 형상에도 영향을 미칠 때

한 회사의 직원들에게 생산성 향상 교육을 제공한 후,
이 직원들이 다른 부서 직원들에게도
지식을 공유하여 전체적인 생산성이 향상됨

한 그룹이 경매에서 이기면 다른 그룹은 진다.
이런 이유로, 한 그룹이 경매에서 이기는 데 유리한 이벤
트를 받으면 다른 그룹은 질 가능성이 더 크다.
따라서 치료 효과가 과대평가될 수 있다.

Source : Why Spillover Effects Bias Your AB Testing Results and Ways to Overcome them

SUTVA

Stable Unit Treatment Value Assumption

Consistency + No interference

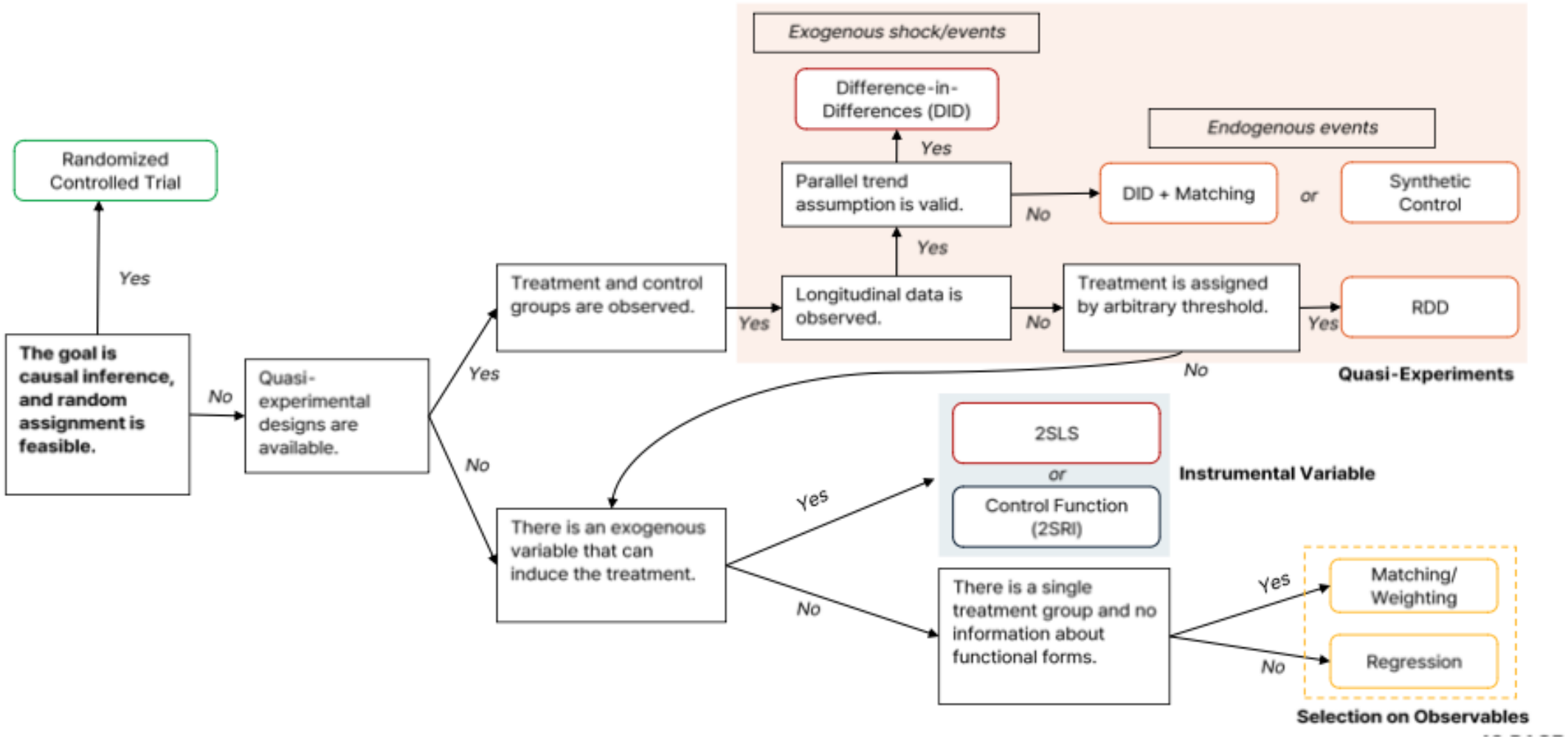
Potential outcome 을 명확하게 정의할 수 있는
Treatment를 디자인 하는 것이 중요

실험군과 대조군 내 사용자들을
고립 시키는 것이 중요

SUTVA 가정을 위반 : '간섭'

Source : 인과추론과 실무 : RCT 실험플랫폼

Causal inference Toolbox

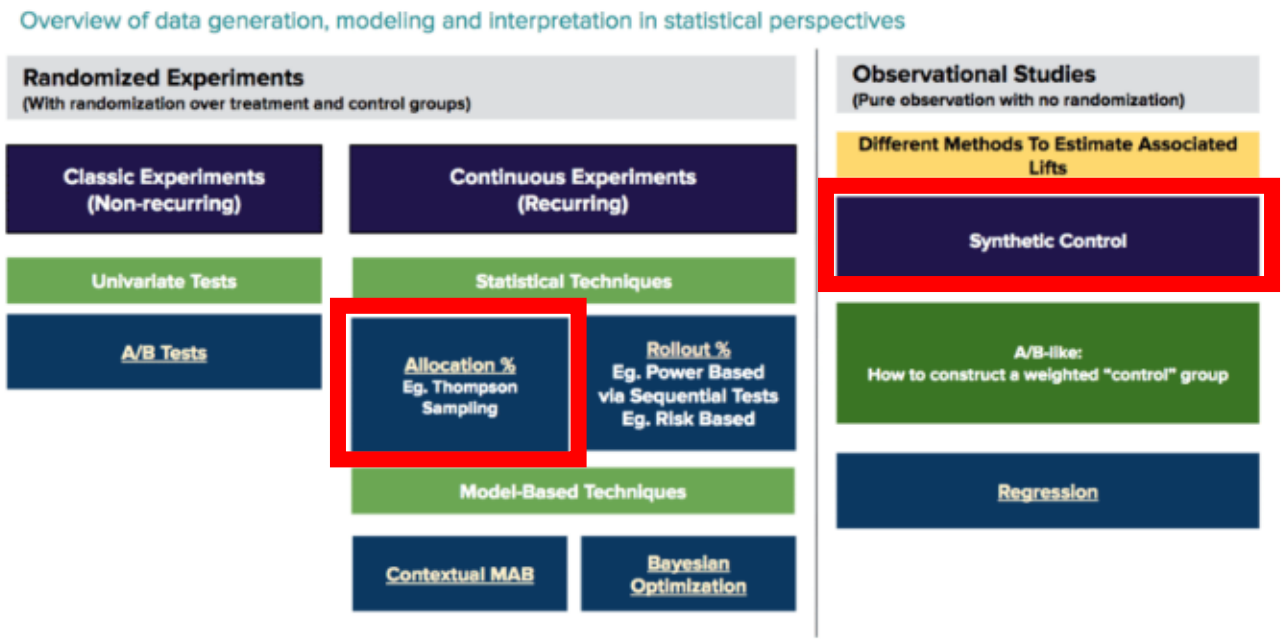


Source : 인과추론과 실무 : RCT 실험플랫폼

대안 실험

무작위 통제 실험이 어려울 때

Bias를 제거하기 위해 처치(Treatment) 가 어떻게 배정되었는 지 중요 -> 독립성 가정



스위치백 = 연속실험

: 시간의 경과에 따라 서로 다른 처치를 적용하여 그 효과를 비교하는 방식

시네틱컨트롤 = 관찰연구

: 유사한 인공적인 통제군을 생성하여, 실험군의 변화가 개입(처치)에 의한 것인지 평가하는 방식

Source : 책_실무로 통하는 인과추론 CHAPTER 10 지역 시험과 스위치백 실험

1. 통제집단 합성법

실험 대상이 적을 때

네트워크 효과

통제집단 합성법 설계

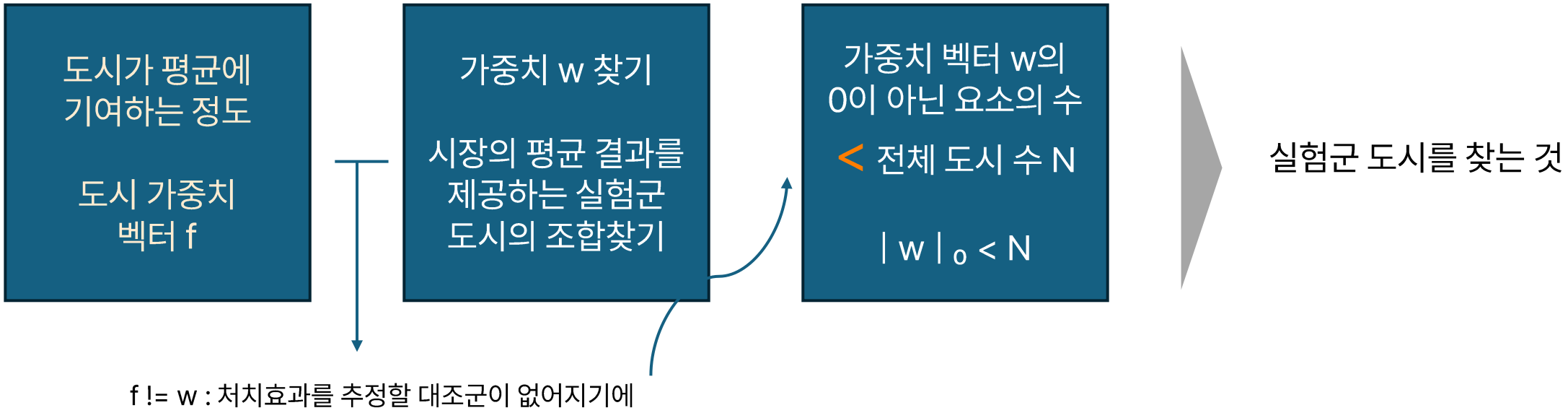
1차 목표

모든 실험 대상의 평균 행동을 근사하는 가상의 실험군 을 찾자

대조군 vs 실험군

실험군 도시를 찾아서

모든 실험 대상의 평균 행동을 근사하는 가상의 실험군 (synthetic treatment unit) 을 찾는다 (!= 대조군찾기)



Source : 책_실무로 통하는 인과추론 CHAPTER 10 지역 시험과 스위치백 실험

대조군 vs 실험군

대조군 도시를 찾아서

모든 실험 대상의 평균 행동을 근사하는 가상의 실험군 (synthetic treatment unit) 을 찾는다

첫 번째와
다른 도시 그룹 찾기

$$Y_{post}\, f = Y_{post}\, v$$

$$s.t\ w_i\, v_i = 0\ \forall i$$

동일한 도시를
실험군 및 대조군 도시로
동시에 사용할 수 없다.

개입 전 기간을
살펴보고

각각의 가중평균이
시장 평균에 근접하
도록 한다

전체 시장 전에 진행
하기에 통제 집단 합
성법을 사용하면

외삽이 허용되지 않
으므로 큰 도시 선택

기존 데이터 범위 내에서 예측 > 신뢰성 확보

Source : 책_실무로 통하는 인과추론 CHAPTER 10 지역 시험과 스위치백 실험

무작위로 실험군 선택하기

With SyntheticControl

충분히 적절한 도시 그룹 (실험군/대조군) 을 찾으면 된다

```
def get_sc(geos, df_sc, y_mean_pre):

    model = SyntheticControl(fit_intercept=True)
    model.fit(df_sc[geos], y_mean_pre)

    selected_geos = geos[np.abs(model.w_) > 1e-5]

    return {"geos": selected_geos, "loss": model.loss_ }


get_sc(rand_geos, df_piv, y_avg)
```

```
{'geos': array(['salvador', 'aracaju'], dtype='<U23'),
 'loss': 1598616.8087526595}
```

절편 이동 허용 : SyntheticControl 클래스 사용

2가지 도시를 실험군으로 선택

손실 함수의 값도 정확히 기록해야 함

손실 함수를 최소화

Source : 책_실무로 통하는 인과추론 CHAPTER 10 지역 시험과 스위치백 실험

무작위로 실험군 선택하기

With SyntheticControl

충분히 적절한 도시 그룹 (실험군/대조군) 을 찾으면 된다

```
def get_sc_st_combination(treatment_geos, df_sc, y_mean_pre):

    treatment_result = get_sc(treatment_geos, df_sc, y_mean_pre)

    remaining_geos = df_sc.drop(
        columns=treatment_result["geos"]
    ).columns

    control_result = get_sc(remaining_geos, df_sc, y_mean_pre)

    return {"st_geos": treatment_result["geos"],
            "sc_geos": control_result["geos"],
            "loss": treatment_result["loss"] + control_result["loss"]}


resulting_geos = get_sc_st_combination(rand_geos, df_piv, y_avg)
```

```
[25] resulting_geos.get("st_geos")
```

```
array(['salvador', 'aracaju'], dtype='<U23')
```

```
len(resulting_geos.get("st_geos")) + len(resulting_geos.get("sc_geos"))
```

```
50
```

m이 작을 경우 get_sc 호출을 통해
이전과 동일한 실험군을 얻을 수 있다.

m > 0 이지만 매우 작은 경우,
모든 도시들을 포함하고 가중치를 약간 조정

Source : 책_실무로 통하는 인과추론 CHAPTER 10 지역 시험과 스위치백 실험

무작위로 실험군 선택하기

With SyntheticControl

AVERAGE : 평균
SC : 가상의 대조군
ST : 가상의 실험군

대부분의 손실은 가상의 실험군에서 발생
가상의 대조군은 시장평균과 거의 비슷

(∵ 실험군을 설정 후, 대조군을 호출)

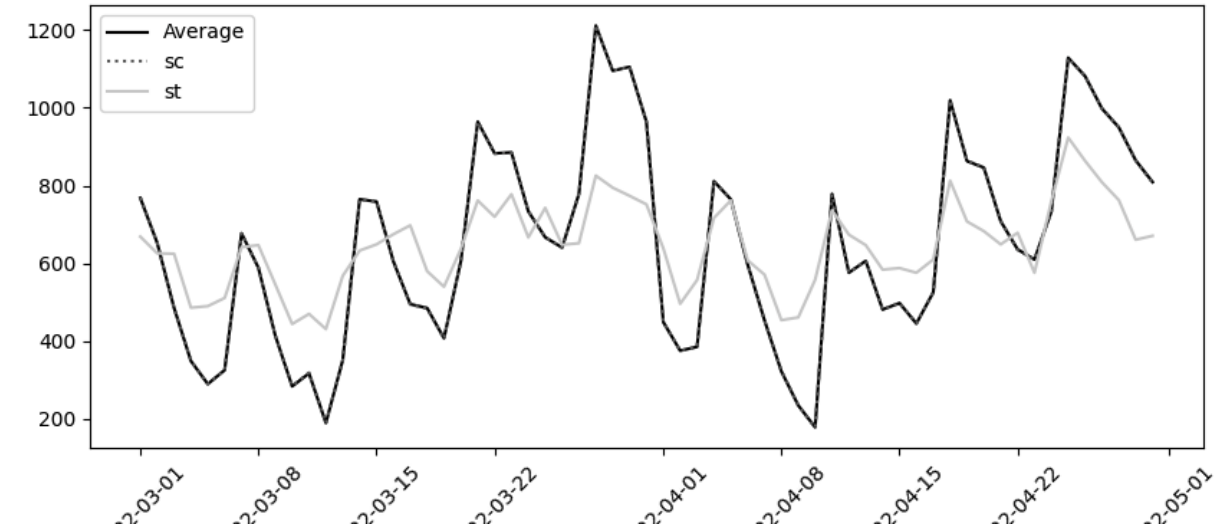
```
synthetic_tr = SyntheticControl(fit_intercept=True)
synthetic_co = SyntheticControl(fit_intercept=True)

synthetic_tr.fit(df_piv[resulting_geos.get("st_geos")], y_avg)
synthetic_co.fit(df_piv[resulting_geos.get("sc_geos")], y_avg)

plt.figure(figsize=(10,4))
plt.plot(y_avg, label="Average")
plt.plot(y_avg.index, synthetic_co.predict(df_piv[resulting_geos.get("sc_geos")]), label="sc", ls=":")
plt.plot(y_avg.index, synthetic_tr.predict(df_piv[resulting_geos.get("st_geos")]), label="st")


plt.xticks(rotation=45)
plt.legend()
```

```
<matplotlib.legend.Legend at 0x7f3e681da5f0>
```



Source : 책_실무로 통하는 인과추론 CHAPTER 10 지역 시험과 스위치백 실험

Causal Inference Lab
Pseudo Lab

2. 스위치백 실험

실험 대상이 1개

네트워크 효과

스위치백 실험 설계

1차 목표

동일한 처치배정에 처치 배정/미배정을 반복하며, 전체 처치효과를 분석한다

이월 효과의 차수가 작아서 관측처치효과가 금방 사라지는 경우에 가능

실험 대상이 1개

네트워크 효과

수요와 공급 ISSUE + 서로가 서로에게 영향을 미치는 네트워크, 스필오버 효과 포함

KEY POINT

1) 가격이 원래 수준으로 돌아갔을 때,
2) 가격 인상의 효과가 금방 사라진다면
3) 여러번 가격 인상을 반복하며 전후 비교

이월 효과 구하기

Ex) 가격이 원래 수준으로 돌아올 때
과잉 공급이 얼마만큼 만에 사라지는 지

실험 대상이 1개

네트워크 효과

수요와 공급 ISSUE + 서로가 서로에게 영향을 미치는 네트워크, 스필오버 효과 포함

```
df = pd.read_csv("/content/drive/MyDrive/sb_exp_every.csv")
df.head()
```

	d	delivery_time	...	delivery_time_0	tau
0	1	2.84	...	5.84	-3.0
1	0	4.49	...	6.49	-5.0
2	0	7.27	...	8.27	-6.0
3	1	5.27	...	8.27	-6.0
4	1	5.59	...	10.59	-6.0

5 rows × 5 columns

다음 단계: df변수로 코드 생성 추천 차트 보기

예시 데이터

D = 처치 여부
delivery_time : 평소
delivery_time_1 : 가격 인상이 적용되었을 때
Delivery_time_0 : 가격 인상이 적용되지 않았을 때

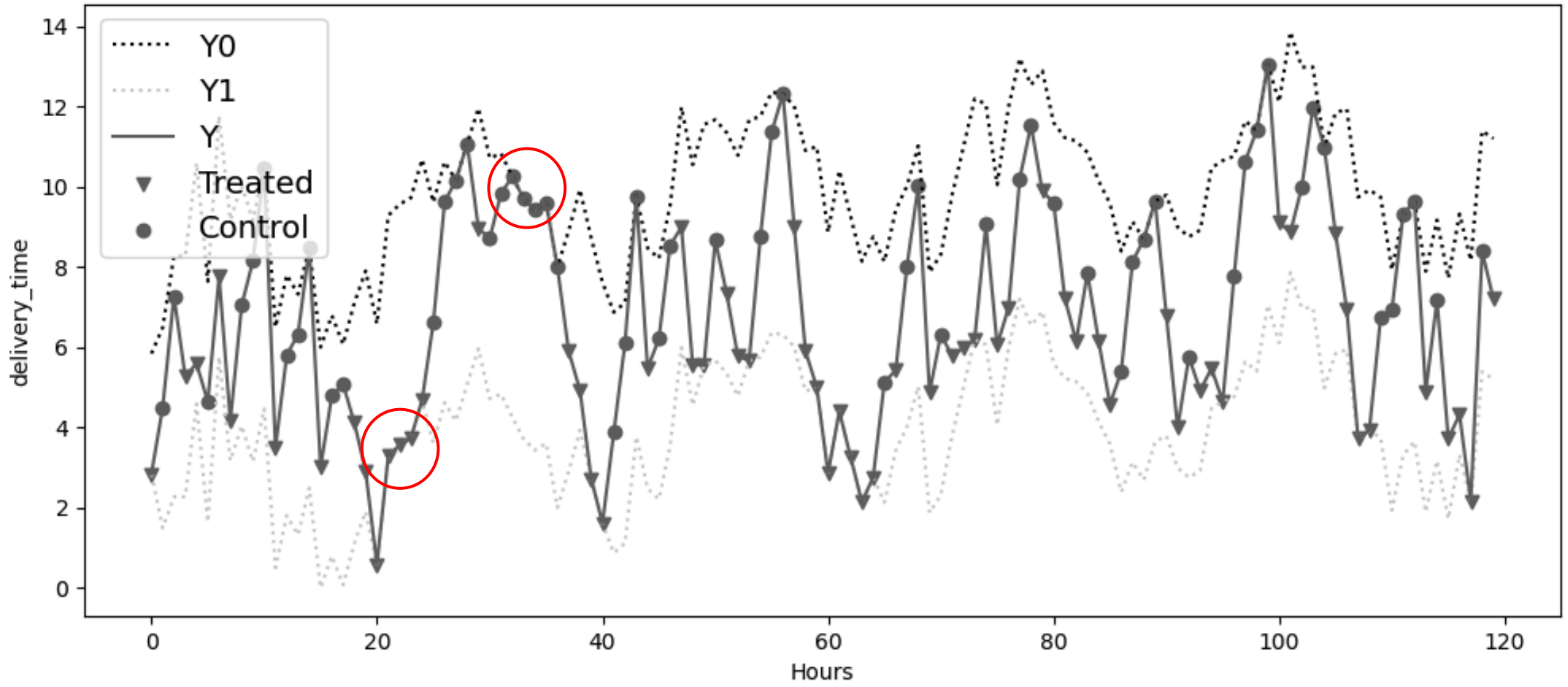
Tau (전체 처치효과)
: delivery time_1 - delivery time_0

Source : 책_실무로 통하는 인과추론 CHAPTER 10 지역 시험과 스위치백 실험

실험 대상이 1개

delivery_time_1 : 가격 인상이 적용되었을 때
delivery_time_0 : 가격 인상이 적용되지 않았을 때

```
<matplotlib.legend.Legend at 0x7f3e681bc400>
```



Source : 책_실무로 통하는 인과추론 CHAPTER 10 지역 시험과 스위치백 실험

이월 효과의 차수 측정

최적 설계는 차수에 영향을 미친다

잠재적 결과의 모델 설정이 정확하다는 전제 (도메인 지식을 활용해야 한다)

```
import statsmodels.formula.api as smf

model = smf.ols("delivery_time ~" + "+".join([f"d_l{l}"
                                              for l in range(7)]),
                data=df_lags).fit()

model.summary().tables[1]
```

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	9.3270	0.461	20.246	0.000	8.414	10.240
d_l0	-2.9645	0.335	-8.843	0.000	-3.629	-2.300
d_l1	-1.8861	0.339	-5.560	0.000	-2.559	-1.213
d_l2	-1.0013	0.340	-2.943	0.004	-1.676	-0.327
d_l3	0.2594	0.341	0.762	0.448	-0.416	0.935
d_l4	0.1431	0.340	0.421	0.675	-0.531	0.817
d_l5	0.1388	0.340	0.408	0.684	-0.536	0.813
d_l6	0.5588	0.336	1.662	0.099	-0.108	1.225

이월 효과 = 2
두 번째 시차까지 유의하다

시차를 최소한으로 하여,
분산을 크게 줄이는데 포커스를 맞춘다

랜덤화 지점 + 시차를 활용하여,
최적 설계를 만든다.

Source : 책_실무로 통하는 인과추론 CHAPTER 10 지역 시험과 스위치백 실험

디자인 기반의 추정

통계적 모델 설정이 불안할 때

이월 효과의 차수를 어떻게 하면 더 쉽게 찾을 수 있을까?

Y_d 잠재적 결과 모델 설정이 정확할 때

IPW **1) 관측된 결과를** /처치 확률의 역수인 $E[Y_d] = N^{-1} \sum(Y_d 1(D = d)/P(D = d))$ 에 **2) 곱해** 잠재적 결과를 재구성

---

1. 랜덤화 지점으로부터 랜덤화 창(randomization window)을 식별
2. 이월 창(carryover window)을 계산
3. 고유한 요소를 센다
4. 각 열을 합산한 후 1을 더하면 원래 배열의 각 지점에 해당하는 랜덤화 창의 개수가 반환

> 랜덤화 빈도가 다를 때 어떻게 작동하는 지 확인할 수 있다

Source : 책_실무로 통하는 인과추론 CHAPTER 10 지역 시험과 스위치백 실험

디자인 기반의 추정

단순화 과정은 m의 값을 알아야 하기에 도메인 지식을 활용하여 구해야함

$$\hat{r} = \frac{1}{T-m}\sum_{t=m+1}^{T}\left\{Y_t\left(\frac{1(D_{t-m:t}=1)}{P(D_{t-m:t}=1)} - \frac{1(D_{t-m:t}=0)}{P(D_{t-m:t}=0)}\right)\right\}$$

IPW 추정량

Source : 책_실무로 통하는 인과추론 CHAPTER 10 지역 시험과 스위치백 실험

최적의 스위치백 설계

분산을 최소화 하는 설계

가능한 최소한의 가정을 하여, 직관적인 수준에서 실험을 설계할 때

1차 대안 이월 효과의 차수가 2(m) 인 경우, 3(m+1) 기간 마다 랜덤화 하는 것

2차 대안 T = 최적의 랜덤화 지점, m은 이월 효과의 차수, n은 T/m = n 을 만족하는 4이상의 정수

T= 12, m = 2

t=1 에서 랜덤화, 크기가 2인 간격을 남겨둔 후,
t=3, 5, 7, 9 에서 랜덤화 하고
t = 11, 12 에서 최종적으로 크기가 2인 간격을 남긴다

비용과 리소스가 적을 때
실행해봐도 괜찮음
(이월 효과가 있을 때)

Source : 책_실무로 통하는 인과추론 CHAPTER 10 지역 시험과 스위치백 실험

2. 스위치백 실무사례

실무 사례 : STATSIG

a/b test 실험 플랫폼

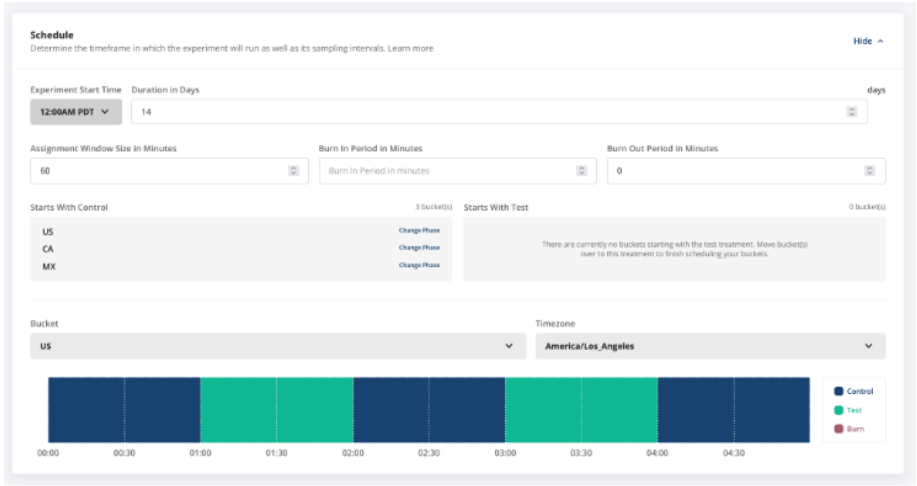
최근 많은 플랫폼에서 활용하고 있는 a/b test 플랫폼 (유사 = abtesty, vwo, 핵클 등)

TEST SCENARIO

1. 스위치백 버킷에 귀속시킨다
2. 귀속 수준을 바탕으로 지표를 계산한다
3. 실험군과 대조군과의 평균차이를 계산하고,
   부트스트랩 기법을 사용하여 신뢰구간을 얻는다

<-8장 이중차분법

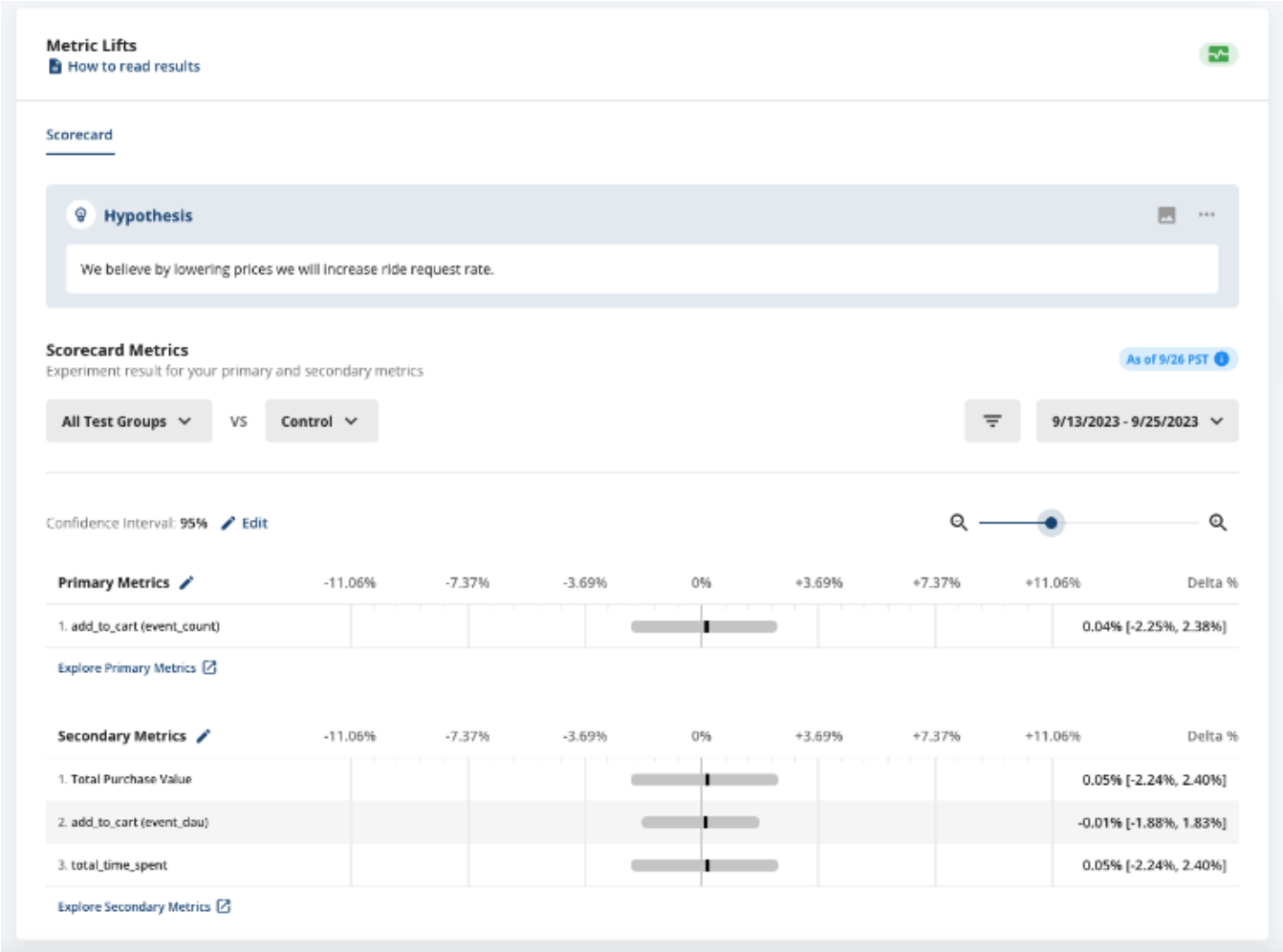
- 시작 시간
- 기간(일)
- 할당 창 크기(분)
- 번인/번아웃 기간(분)
- *(사전 정의된 버켓팅만 해당)* 각 버킷에 대한 시작 단계(치료 그룹)



< Statsig 스위치백 >

Source : STATSIG 엔지니어링 블로그 : https://docs.statsig.com/experiments-plus/switchback-tests

실무 사례 : STATSIG



가격을 낮추면 요청 비율이 증가할 것

<- KPI

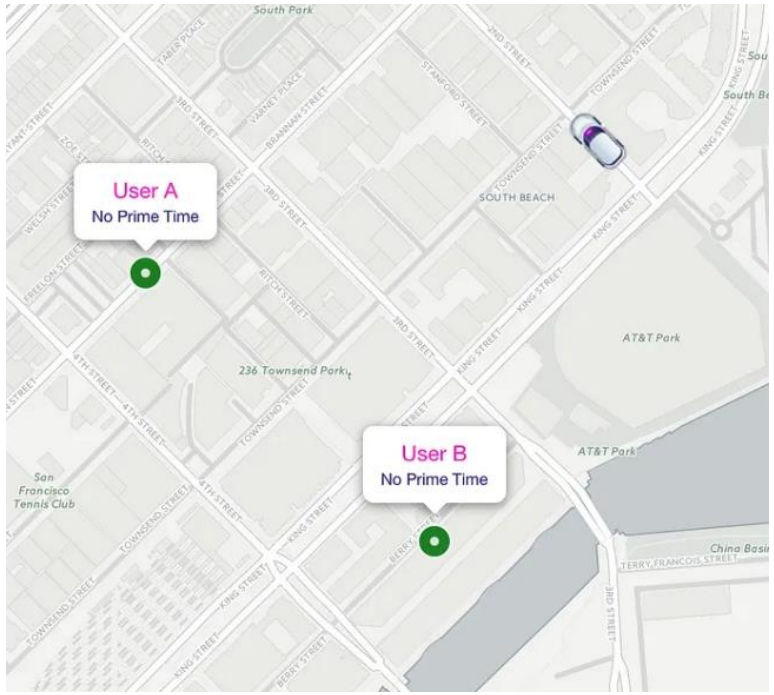
<- 2차 지표

Source : STATSIG 엔지니어링 블로그 : https://docs.statsig.com/experiments-plus/switchback-tests

실무 사례 : LYFT

분산 – 편향 트레이드 오프

공급부족 시나리오를 해결하기 위한 가격 지원 정책 (승객 대상)



가격 지원을 하게 되면, A 유저가 기사를 선택했을 때,
B 유저가 해당 기사를 볼 확률이 낮아진다 > 동일하게 테스트 필요

Randomization unit	Bias axis	Variance axis
User sessions	↑	↓
Users		
Fine spatial units (geohash)		
Time interval (hour)		
Coarse spatial units (city)		

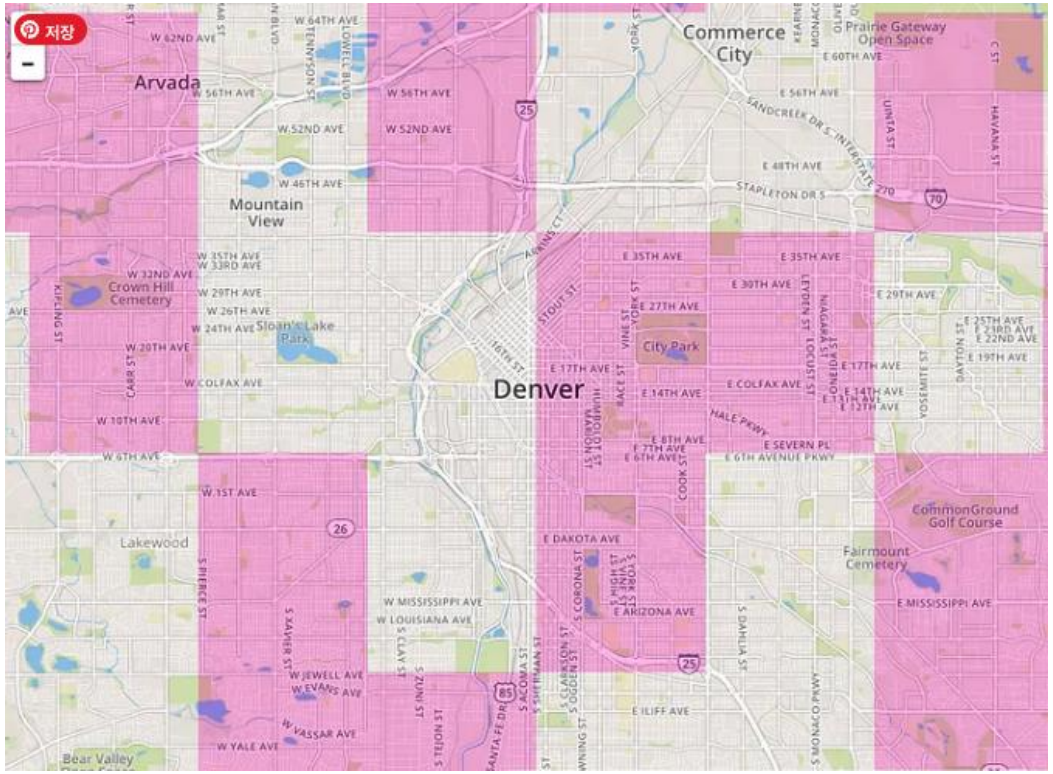
표 1. 실험 단위의 다양한 선택은 편향-분산 트레이드오프 스펙트럼의 다양한 지점에 해당합니다. 네트워크 실험의 맥락에서 편향은 간섭 효과에서 비롯되고 분산은 단위 집합의 기수 감소와 단위 간 이질성에서 비롯됩니다.

분산 – 편향 트레이드오프

Source : LYFT 엔지니어링 블로그 : https://eng.lyft.com/experimentation-in-a-ridesharing-marketplace-f75a9c4fcf01

실무 사례 : LYFT

주요 KEY POINT



KEY POINT

1) 1시간
2) 공간 단위
3) 무작위 처치 여부

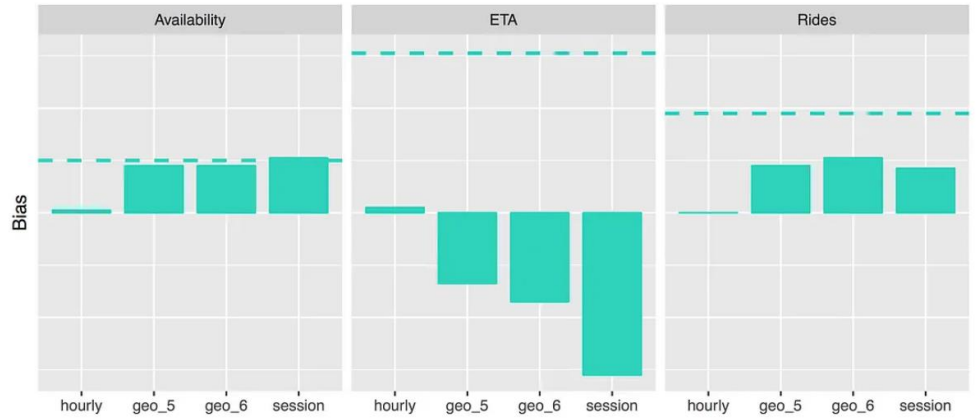
KPI

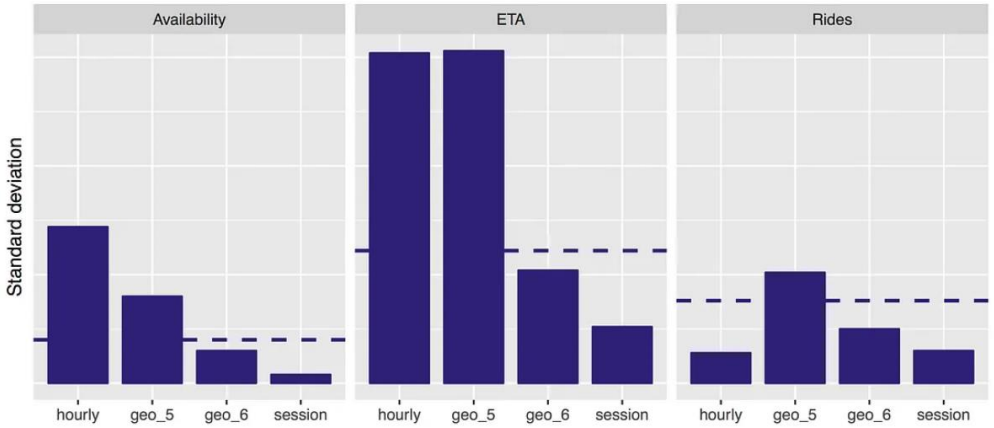
1) 가용성
2) ETA (예상도착시간)
3) LYFT 승차 횟수

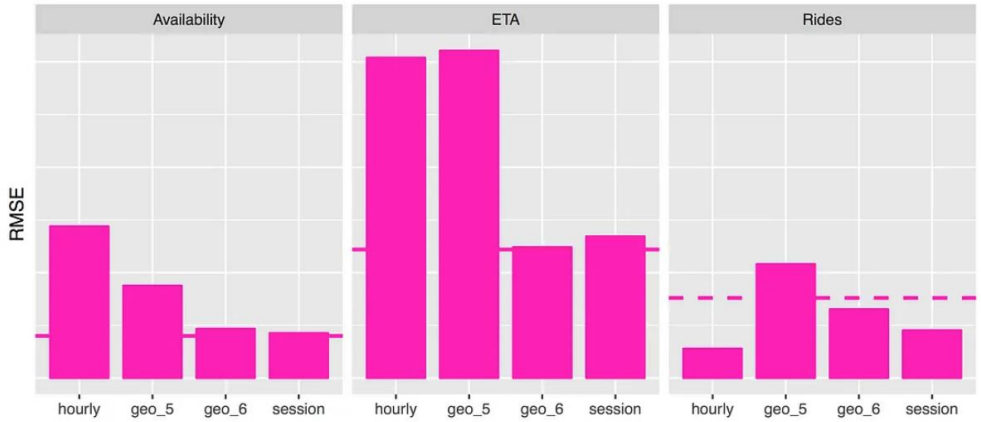
Source : LYFT 엔지니어링 블로그 : https://eng.lyft.com/experimentation-in-a-ridesharing-marketplace-f75a9c4fcf01

실무 사례 : LYFT

타협점을 찾기란 어렵다







무작위 세션은 편향 + 표준편차 + RMSE(평균 제곱근오차) 모두 안좋은 수치
시간당 실험은 좋은 성과 > ETA 에서 실패를 보이고 있음 (평균편차, RMSE)
공간 설계는 부정적 영향을 과소평가할 가능성이 있음 (Rides 수치)

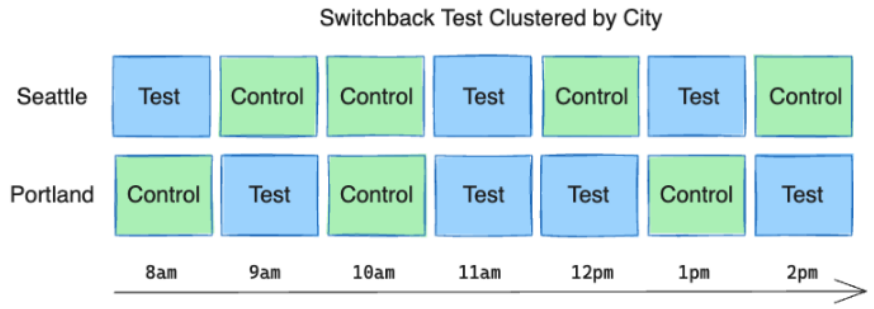
향후 직접적, 간접적인 효과 측정을 통해
연결된 가설을 세우고, 개선을 기대할 수 있음

Source : LYFT 엔지니어링 블로그 : https://eng.lyft.com/experimentation-in-a-ridesharing-marketplace-f75a9c4fcf01

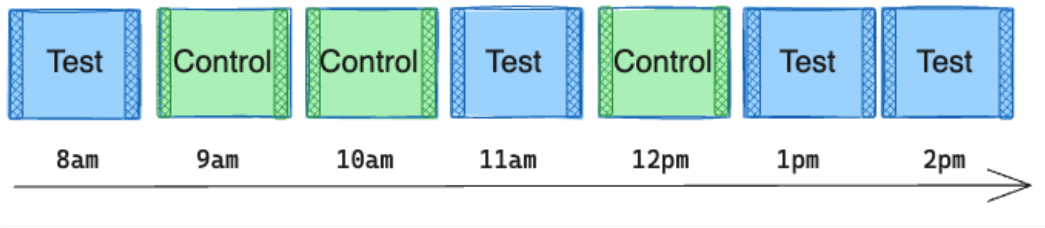
실무 사례 : LYFT

주요 TIP

공급부족 시나리오를 해결하기 위한 가격 지원 정책 (승객 대상)



도시 클러스터링



교차 오염 방지

Source : STATSIG 블로그 : https://www.statsig.com/blog/switchback-experiments

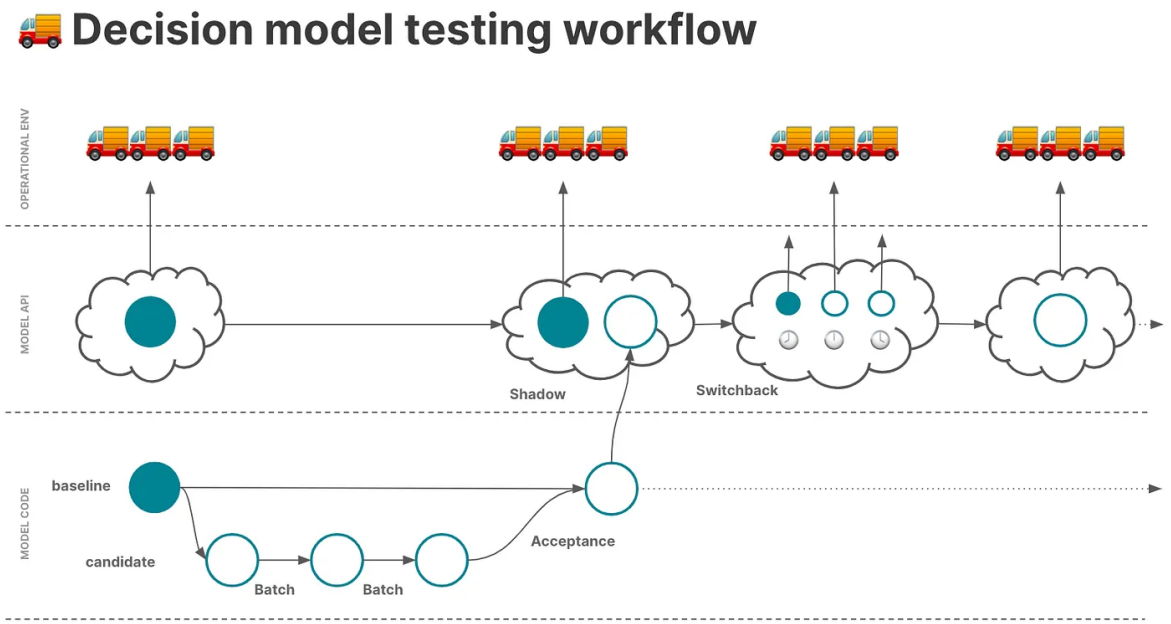
사례 예시 : 콜드체인 식자재 배달

의사결정 과정

최적의 경로 분석 : 시간, 선택, 도로주행, 배정 정거장 등을 고려한 모델 결정

▼ Plan Summary

UNIT INDEX	UNIT INSTANCE	UNIT DURATION (MINUTES)	UNIT START TIME	UNIT STOP TIME
0	prod	60	2024-01-13 · 6:00:00 pm	2024-01-13 · 7:00:00 pm
1	prod	60	2024-01-13 · 7:00:00 pm	2024-01-13 · 8:00:00 pm
2	prod	60	2024-01-13 · 8:00:00 pm	2024-01-13 · 9:00:00 pm
3	staging	60	2024-01-13 · 9:00:00 pm	2024-01-13 · 10:00:00 pm
4	prod	60	2024-01-13 · 10:00:00 pm	2024-01-13 · 11:00:00 pm
5	staging	60	2024-01-13 · 11:00:00 pm	2024-01-14 · 12:00:00 am
6	staging	60	2024-01-14 · 12:00:00 am	2024-01-14 · 1:00:00 am
7	prod	60	2024-01-14 · 1:00:00 am	2024-01-14 · 2:00:00 am
8	prod	60	2024-01-14 · 2:00:00 am	2024-01-14 · 3:00:00 am
9	prod	60	2024-01-14 · 3:00:00 am	2024-01-14 · 4:00:00 am
10	staging	60	2024-01-14 · 4:00:00 am	2024-01-14 · 5:00:00 am



Source : https://www.nextmv.io/blog/what-is-switchback-testing-for-optimization-models-and-decision-algorithms
: https://towardsdatascience.com/what-is-switchback-testing-for-decision-models-e26d2007325a

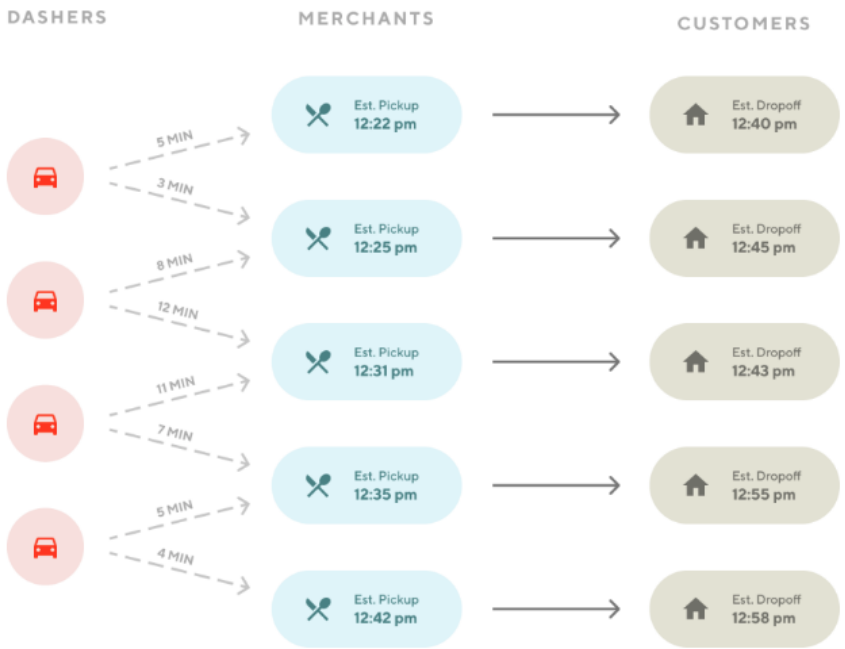
사례 예시 : 도어대시

배송을 어떻게 더 잘할 수 있을까?

DASHER / 음식점 / 도어대시 3면 시장에서의 강화

KEY POINT

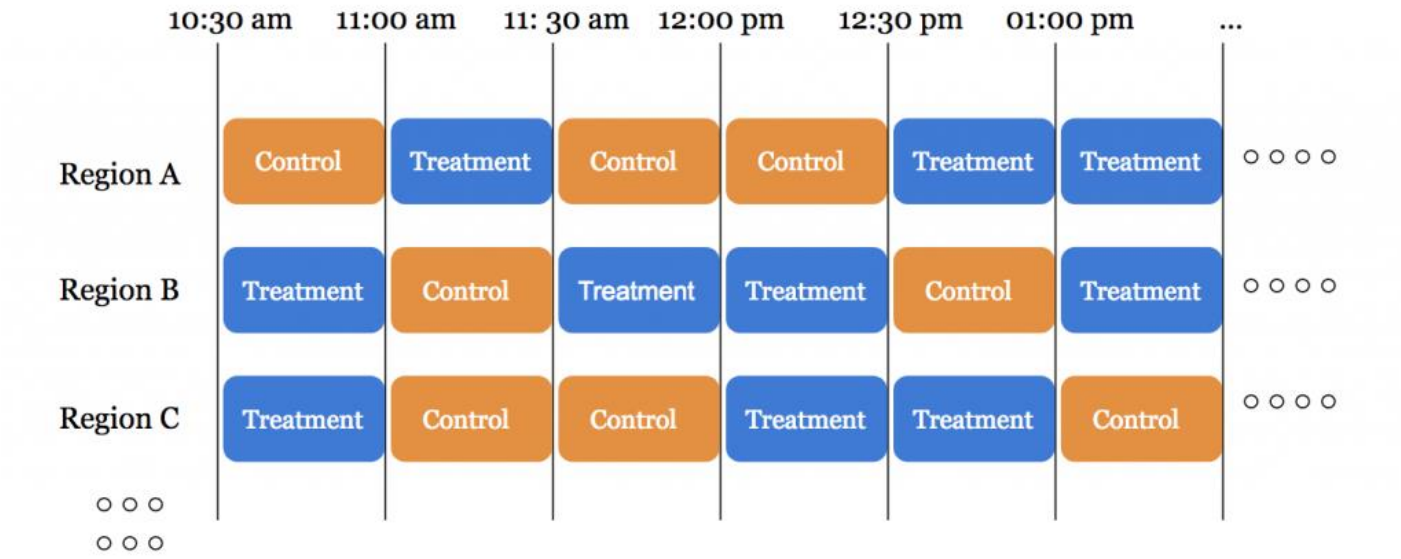
1) 배송 이행에 적합한 대셔 결정
2) 배송 과정 중 적절한 시점 예측
3) 유사한 배달 건 그룹화
4) 공급과 수요 균형을 맞추기 위한 가격조정



Source : https://doordash.engineering/2018/02/13/switchback-tests-and-randomized-experimentation-under-network-effects-at-doordash/

사례 예시 : 도어대시

가격 실험 : SOS 가격 정책 > 대셔의 수요가 적을 때, 가격지원 정책을 쓰는 방향



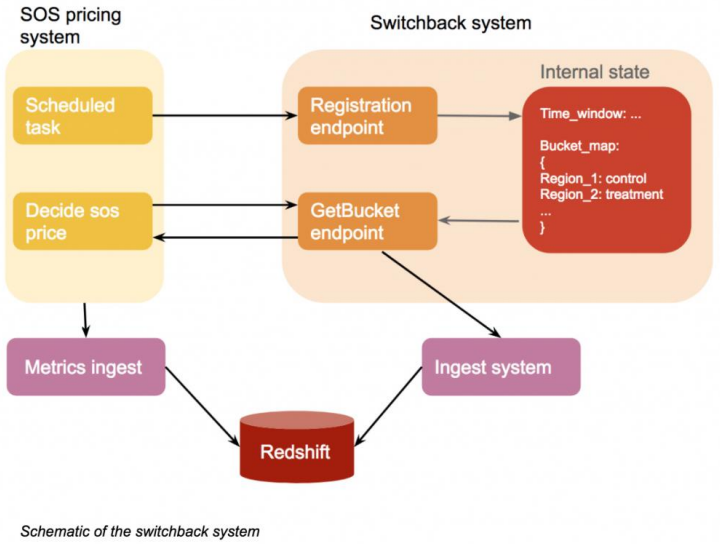
지역 / 시간(30분)

Source : https://doordash.engineering/2018/02/13/switchback-tests-and-randomized-experimentation-under-network-effects-at-doordash/

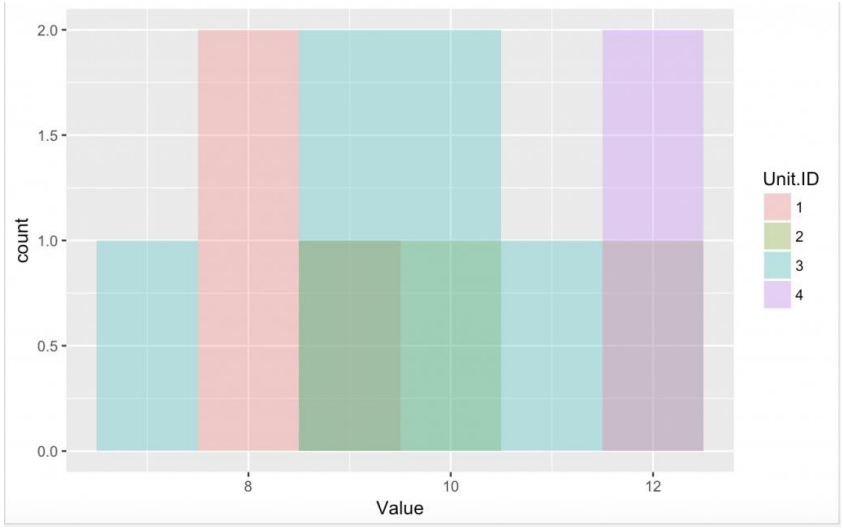
사례 예시 : 도어대시

적절한 시간과 지역 단위에 대해 찾아야 한다

가격 실험 : SOS 가격 정책 > 대셔의 수요가 적을 때, 가격지원 정책을 쓰는 방향



지표 수집 시스템



실험 결과

Time window	Sample stddev	% of time units (indexed to 60min window)	Total margin of error (indexed to 60min window)
20 Minutes	7.94	300%	0.68
30 Minutes	7.47	200%	0.79
60 Minutes	6.72	100%	1
1 day	3.56	4.17%	2.59
1 week	3.12	0.60%	6.02

A/A TEST

Source : https://doordash.engineering/2018/02/13/switchback-tests-and-randomized-experimentation-under-network-effects-at-doordash/

감사합니다
Q&A