Toggle Fullscreen

12. Counterfactual Imputation

Causal Data Science for Business Analytics

Christoph Ihl

Hamburg University of Technology

Monday, 13. July 2026



All of Part V filled one blanked-out block of the panel.

S10 filled it with additive effects. S11 filled it with a donor blend.

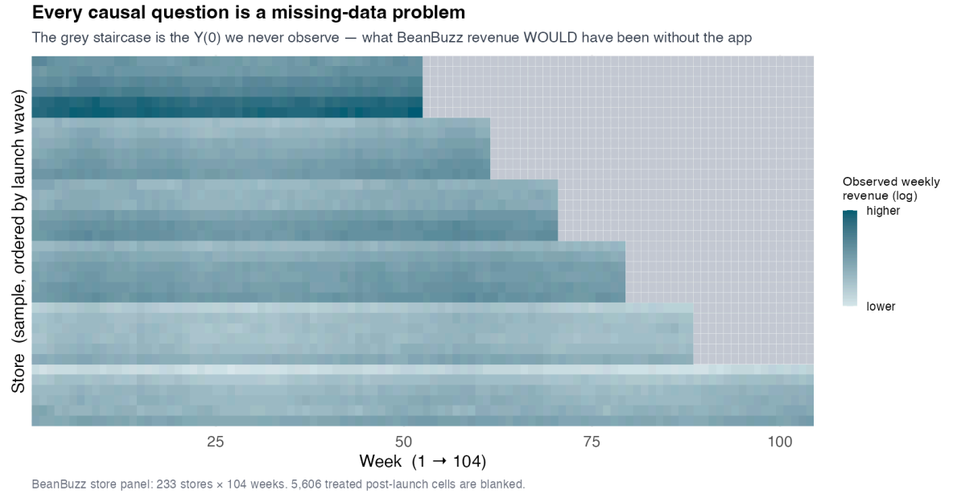
Today: one objective they are both special cases of — and we turn the dial.

Imputation

Every number you reported filled a blank you never saw

The BeanBuzz store panel: **233 stores × 104 weeks**, balanced — **158 treated** (staggered) + **75 never-treated.**

The grey staircase is **5,606 treated cells.** For each we need its **untreated** Y(0) — which by definition we never see.



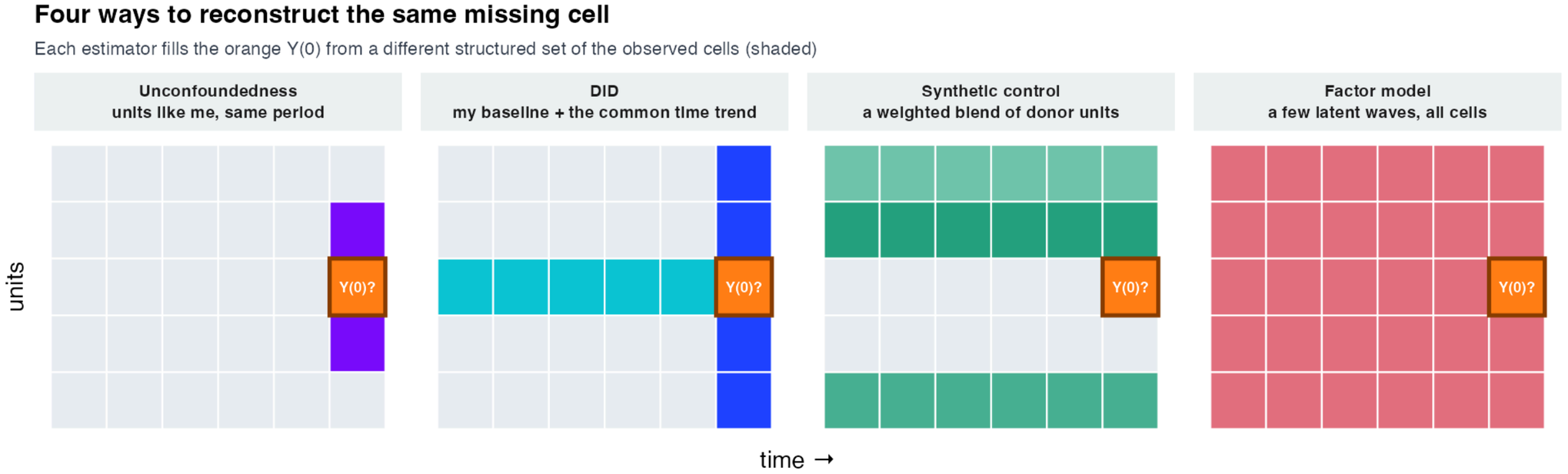
The counterfactual is a missing-data problem

Every ATT is observed – imputed over the grey block – pick the rule that fills grey from white.

Settle the design, and causation becomes prediction

A refrain from week one, half-right: *correlation predicts; it does not identify a cause.* (Athey and Imbens 2017)

But once the **design** is settled — who is treated, when, which cells are clean controls — what remains is **prediction**: forecast the missing Y(0) from the observed cells, then subtract. (Athey et al. 2021)



One objective fills the blank: fit the observed cells, pay for the rest

Write the untreated panel as a matrix L, observed only on the white cells $\mathcal{O}$. Every **imputation** estimator solves the *same* penalized least squares:

$$\min_{L} \underbrace{\sum_{(i,t)\in\mathcal{O}} \left(Y_{it} - L_{it}\right)^2}_{\text{fit the observed (white) cells}} + \lambda \cdot \underbrace{\text{penalty}(L)}_{\text{the structural assumption}}$$

L_{it} is the reconstructed $Y_{it}(0)$: on the **grey** treated cells it *is* the imputed counterfactual, and the ATT averages observed $- L_{it}$ there.

The **penalty is the only thing that changes** (Athey et al. 2021): the *assumption* on the grey cells — **additivity**, a **fixed-rank** factor, or a **nuclear-norm** shrink. One dial, not a menu.

The penalty is a dial, not a menu

One objective, one fect() call — only the **penalty** argument changes (Liu, Wang, and Xu 2024):

```
fect(log_rev ~ app_live, data = panel, index = c("sid","week"), method = ...)
```

Name	Idea — what it does to the latent structure	Implementation
Additive FE *(FEct)*	Keep **no** factor — strict additivity. The dial at r = 0	method = "fe"
Interactive FE *(IFEct)*	Keep the **top-r** latent factors *in full*, drop the rest — **hard-impute**; r chosen by CV	method = "ife"
Matrix Completion *(MC)*	Shrink **all** singular values by λ — **soft-impute** (nuclear norm); no rank to pick	method = "mc"

R and Python

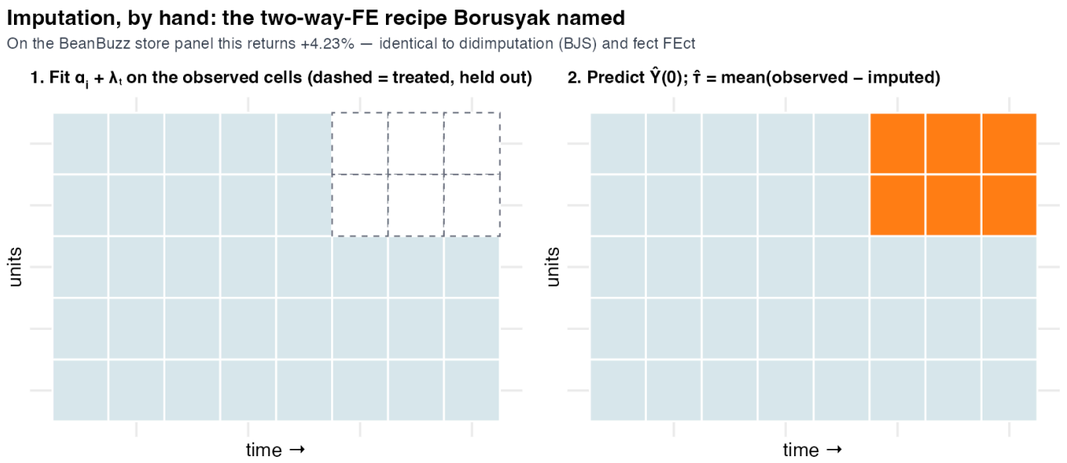
Python has no fect port: the additive rung is *deterministic*, so pyfixest reproduces it to the digit (+4.23%); Interactive FE and MC have no packaged estimator — hand-roll numpy.linalg.svd (hard-/soft-threshold), or read the R output. *(Lab 1 is bilingual; Labs 2+ are R-led.)*

Additive FE

FEct forces the interaction to rank zero

Objective	$L_{it} = \alpha_i + \lambda_t$ (no factor)
Borrows from	my baseline + the common time trend, fit on the **white** cells only
Regularizes by	strict **additivity** (parallel trends)
Dial position	r = 0

```
m <- feols(log_rev ~ 1 | sid + week, data = p[app_live == 0]);
p$y0_hat <- predict(m, p)
```



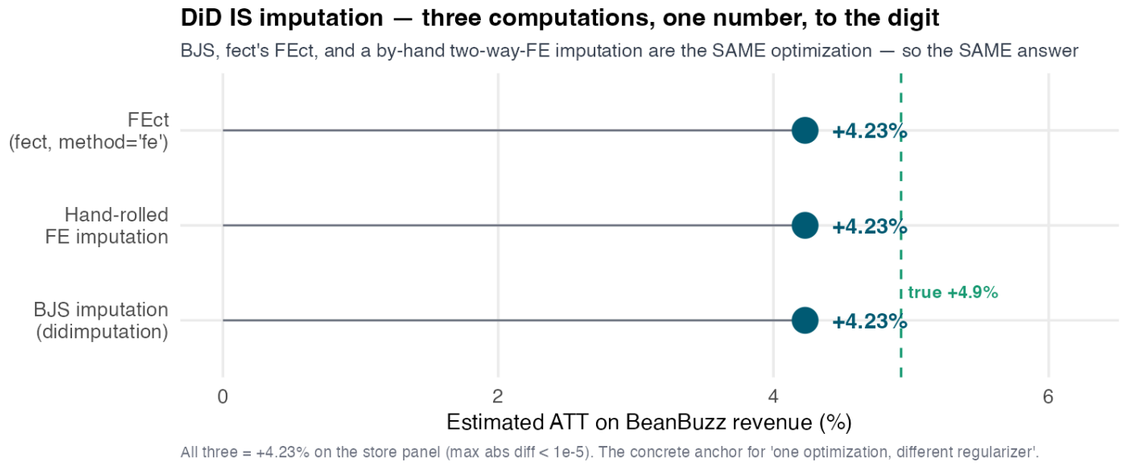
Fit on the untreated cells, predict the rest, average observed − imputed → +4.23%. This rung *is* the Borusyak–Jaravel–Spiess / Gardner imputation. (Borusyak, Jaravel, and Spiess 2024)

Three implementations, one number — to the digit

Three computations that look different:

- fect **FEct** (method="fe")
- **BJS imputation** (didimputation)
- **hand-rolled** feols + predict

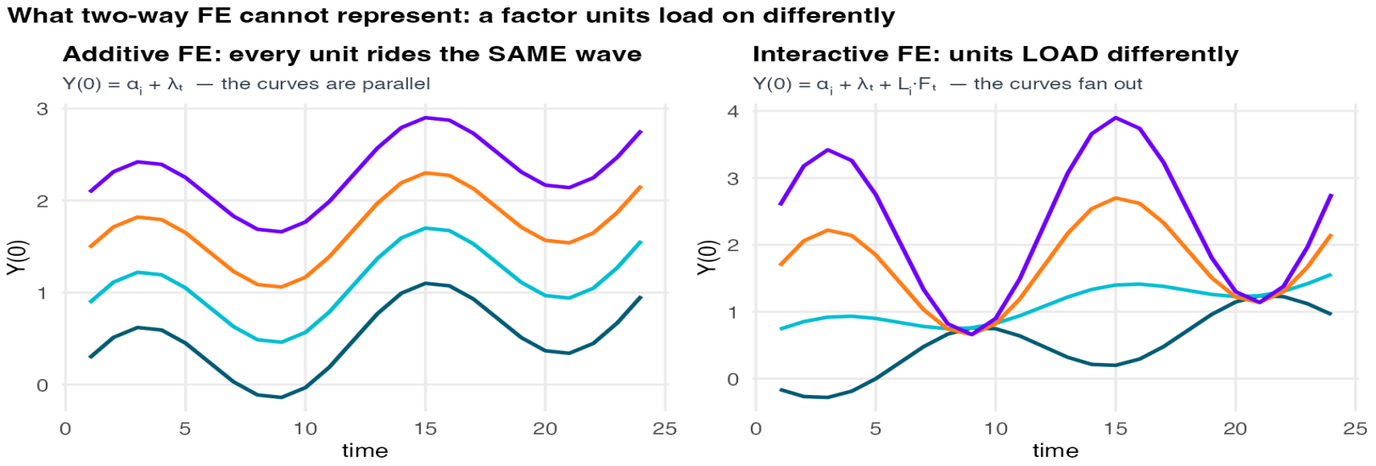
all return +4.23% — max abs diff $< 10^{-5}$.



Not "about the same." **The same estimator, three ways** — restricted two-way-FE imputation is a fixed linear operation, so any correct code *must* return it. (Gardner's did2s lands here too — a coincidence exact only because there are no covariates.)

Blind, not biased: additivity cannot see the factor

The imputation is **honest** — clean-cell fit, no contamination — yet it lands at +4.23%, below the true +4.93%. It isn't biased by selection — **it is blind.**



Two waves run through BeanBuzz. Whole **regions** boom and slump together, and some ride that cycle far harder than others — *the big one*. Fainter: **winter footfall vs year-round routine** makes urban stores ride the seasonal wave harder than roadside ones.

The loading is a β

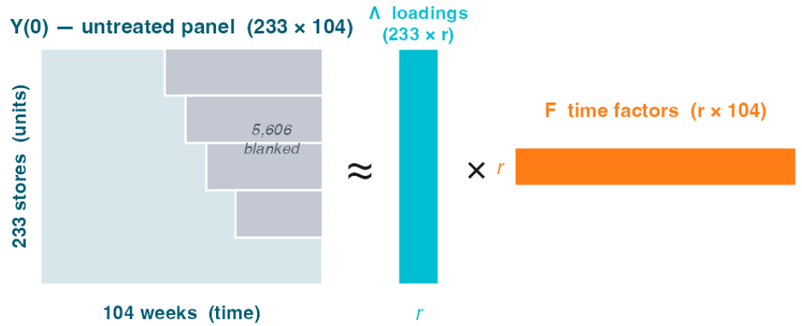
$Y(0) = \alpha_i + \lambda_t$ **is strictly additive — every store rides the *same* time wiggle, so it cannot represent a latent wave F_t that stores ride by *different* amounts. Each store's loading L_i is its exposure to that wave — exactly like a stock's β on the market. Additivity forces the same β on every store, averages the factor away, and undershoots.**

Interactive FE

Low rank: the panel hides in two thin matrices

Factorization: a 233 × 104 matrix hides in one thin column and one thin row

Strip the additive $\alpha_i + \lambda_t$; what is left is low rank — loadings (per store) × factors (per week)

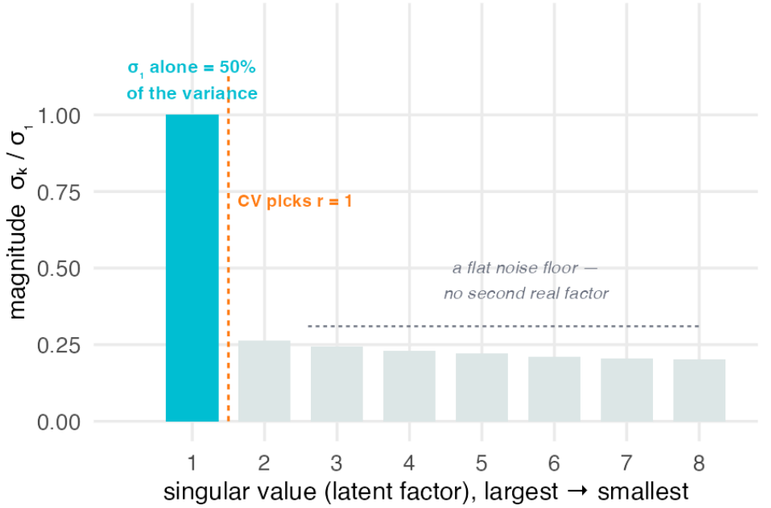


Λ·F is defined on EVERY (store, week) — including the blanked cells.

Multiply the two thin matrices → the blanks are filled. That is completion.

…and the singular values count the factors

SVD of the de-meaned untreated block (75 never-treated stores × 104 weeks)



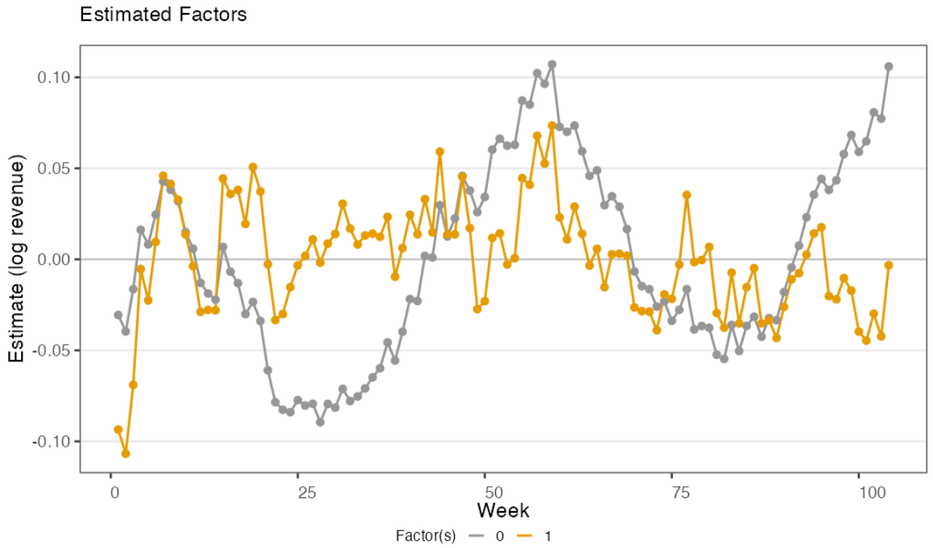
BeanBuzz store panel: two-way de-meaned, then SVD. One factor towers; the rest is noise.

Both panels are **one tool: the singular value decomposition (SVD)**. Split a matrix into $U\Sigma V^\top$, keep the top r, and you *have* the two thin matrices — $\Lambda = U_r\Sigma_r$, $F = V_r^\top$ — while the σ_k on the right are its spectrum. And ΛF is defined on **every** (store, week), including the **5,606 blanked** cells: multiply, and the blanks are filled. That is **completion.**

IFEct hard-imputes a rank-r factor — CV picks r = 1

Objective	$L_{it} = \alpha_i + \lambda_t + \sum_r L_{ir} F_{rt}$
Borrows from	a few shared latent factors everyone loads on
Regularizes by	a **fixed rank** r (hard-impute), CV-chosen
Dial position	r ≥ 1 (FEct is $r = 0$)

So which r? *Cross-validation* — no oracle needed: **hide** a block of *pre-launch* weeks you *do* observe, **re-impute** them at each candidate r, and **keep the simplest** r whose held-out error is within one SE of the best. Here it picks **r = 1**. (Xu 2017; Liu, Wang, and Xu 2024)



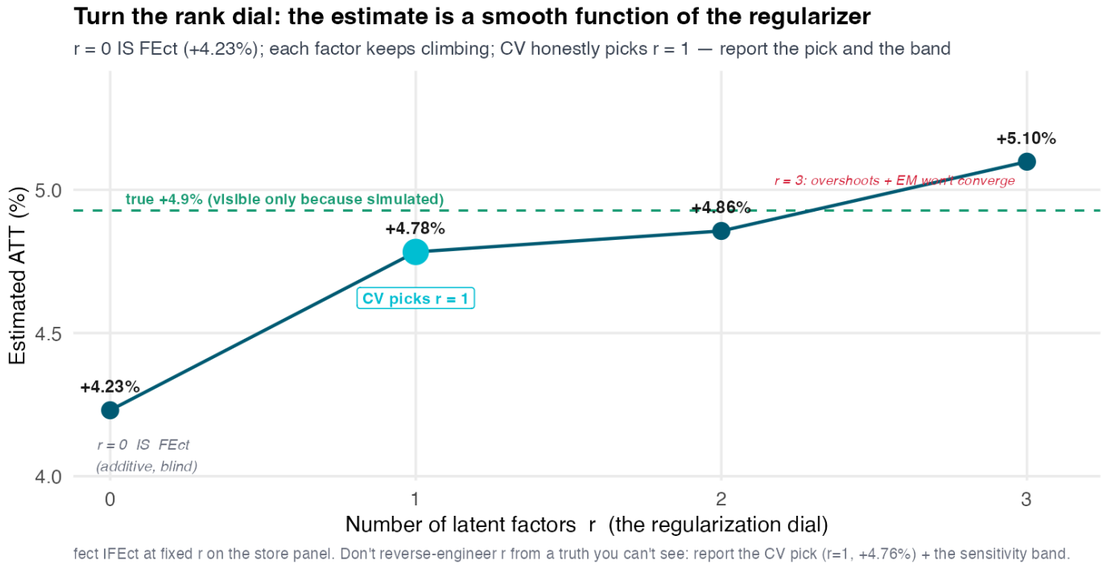
plot(out.ife, type = "factors") — **but read the legend twice.**

0 is not a factor. It is the **time effect** λ_t — loading **1** for every store. A **lift**. The additive model *already had it*.

1 is the one CV bought. Its loadings run -0.069 to $+0.027$ — **18% negative**. Same week, **opposite directions.**

A **fan**, not a lift — and *that* is what additivity cannot see.

Turn the dial: CV picks the rung, not the oracle



You pick the rung **without** the truth line:

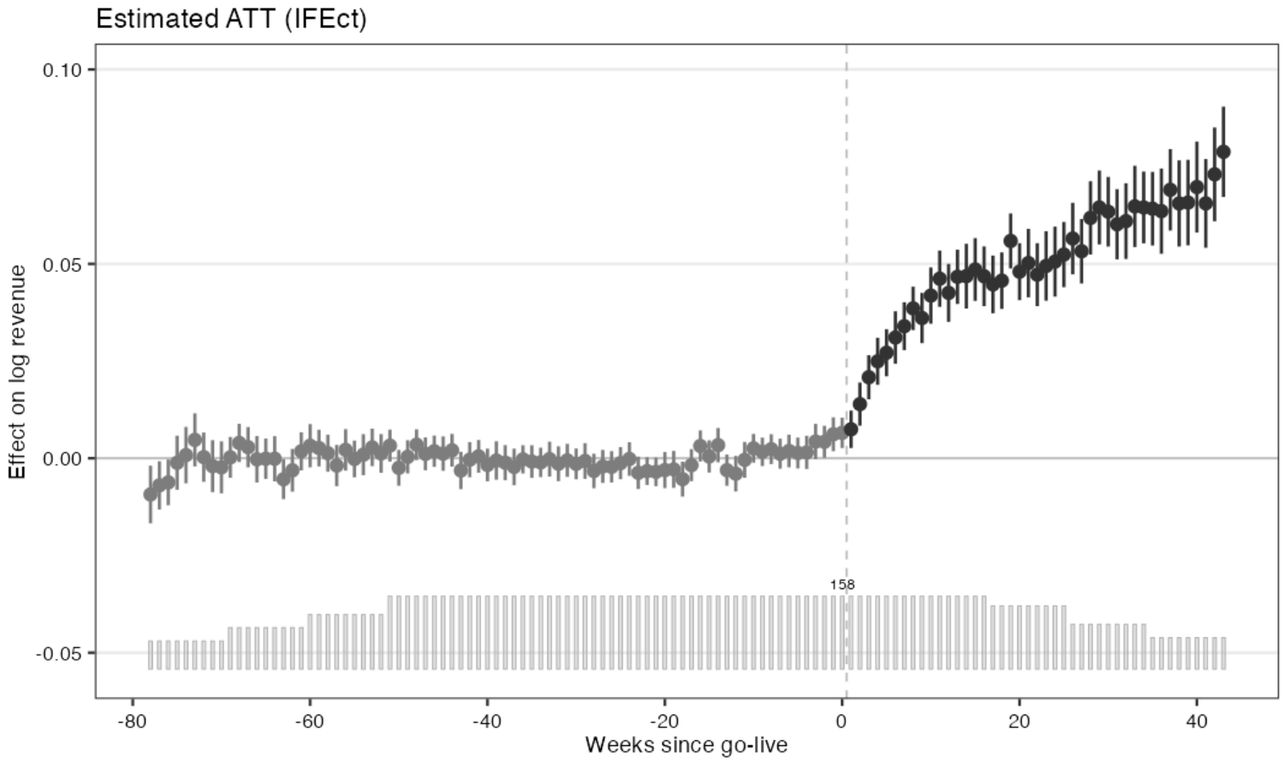
1. **Hide** a 3-week block of *observed, pre-launch* weeks in 10% of stores — and delete everything *after* it from training.
2. **Re-impute** the block at each candidate r; score the held-out error.
3. **Keep** the *simplest* r within one SE of the best — here **r = 1**.

Refit at *fixed* r and the estimate is a smooth function of the regularizer: **r = 0 IS FEct** (+4.23%) → r = 1 **+4.78%** → r = 2 +4.86% → r = 3 +5.10% (overshoots; the EM stops converging).

Report what CV gave you, not what the oracle knows

CV picks r = 1 → +4.76%. You *know* — only because this is simulated — that r = 2 lands nearer +4.9%. A practitioner has no truth line. Report the CV pick; never reverse-engineer the rank from an answer you cannot see.

+4.76% is the *average of a ramp* — not a level



plot(out.ife, main = "Estimated ATT (IFEct)")

Before: flat on zero. *That flatness is a **claim** — the Sensitivity act audits it.*

After: a **ramp**, not a jump — +0.7% wk 1 → +**4.3**% wk 12 → +**6.6**% wk 52.

Adoption is a **diffusion**, not a switch.

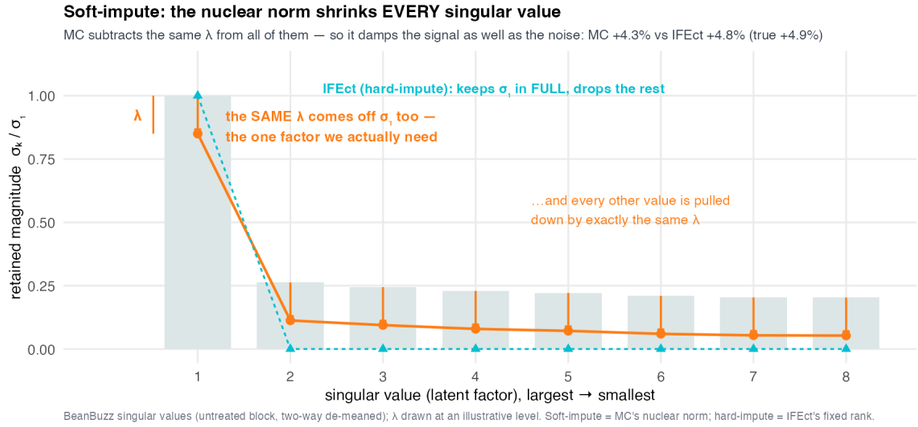
So +**4.76**% is not "the effect." It is the **mean over the post-window you chose** — and the window is a *choice*.

Matrix Completion

MC soft-imputes by shrinking the singular values

The **nuclear norm** = the sum of a matrix's singular values; penalizing it shrinks them toward zero. That is what "encourages low rank" — and it is the *same objective*, a softer knob. (Athey et al. 2021)

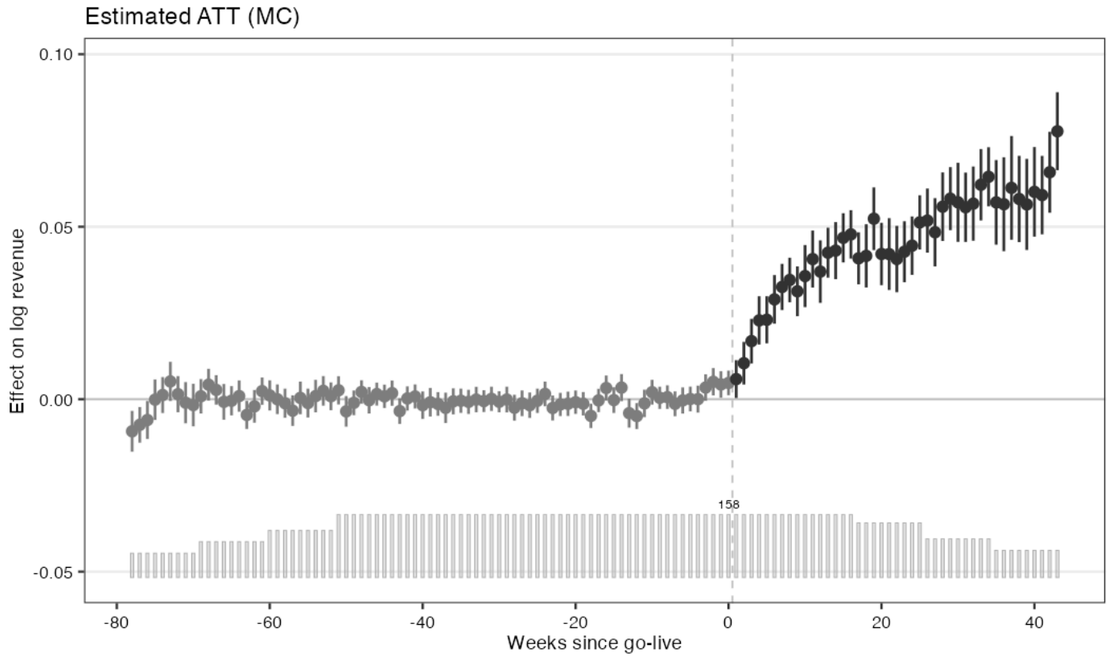
Objective	$\min_L \sum_{\mathcal{O}} (Y - L)^2 + \lambda \|L\|_*$
Borrows from	a low-rank reconstruction — *all* factors, shrunk
Regularizes by	the **nuclear norm** $\|L\|_*$ (soft-impute), CV-λ
Dial position	soft — and **MC → FEct** once λ exceeds $\sigma_{\max}$



Nested, to the boundary

Push λ past the largest singular value and *every* factor is shrunk to zero — MC collapses back to FEct. Hard-impute keeps a few factors whole; soft-impute keeps *all of them, smaller*.

Same panel. Same picture. A different number.



plot(out.mc, main = "Estimated ATT (MC)")

Flat before. Ramps after. **The same shape** — the two post-launch paths correlate 0.99.

ATT	95% CI
IFEct +**4.76**%	[+4.23, +5.28]
MC +**4.31**%	[+3.73, +4.89]

The eye cannot choose. Neither can the interval.

Each CI contains the other's estimate. The whole disagreement — 0.45 pp — is *smaller than a single CI half-width* (0.53 pp). Two models, one panel, one picture, two answers — and nothing you can *see* separates them.

CV also picks the *family* — no truth line needed

One call — fect(method = "both") — builds the folds **once**, then races *both* ladders over them.

IFEct — turn r

r	**held-out MSPE**	
0	0.00171	= FEct
1	0.00105	kept
2	0.00116	
3	0.00118	
4 · 5	0.00154 · 0.00199	

MC — turn λ

$\lambda/\sigma_{\max}$	**held-out MSPE**	
1.00	0.00171	= FEct
0.42	0.00149	kept
0.18	0.00137	lowest
0.075	0.00149	
0.032 · 0.013	0.00160 · 0.00166	

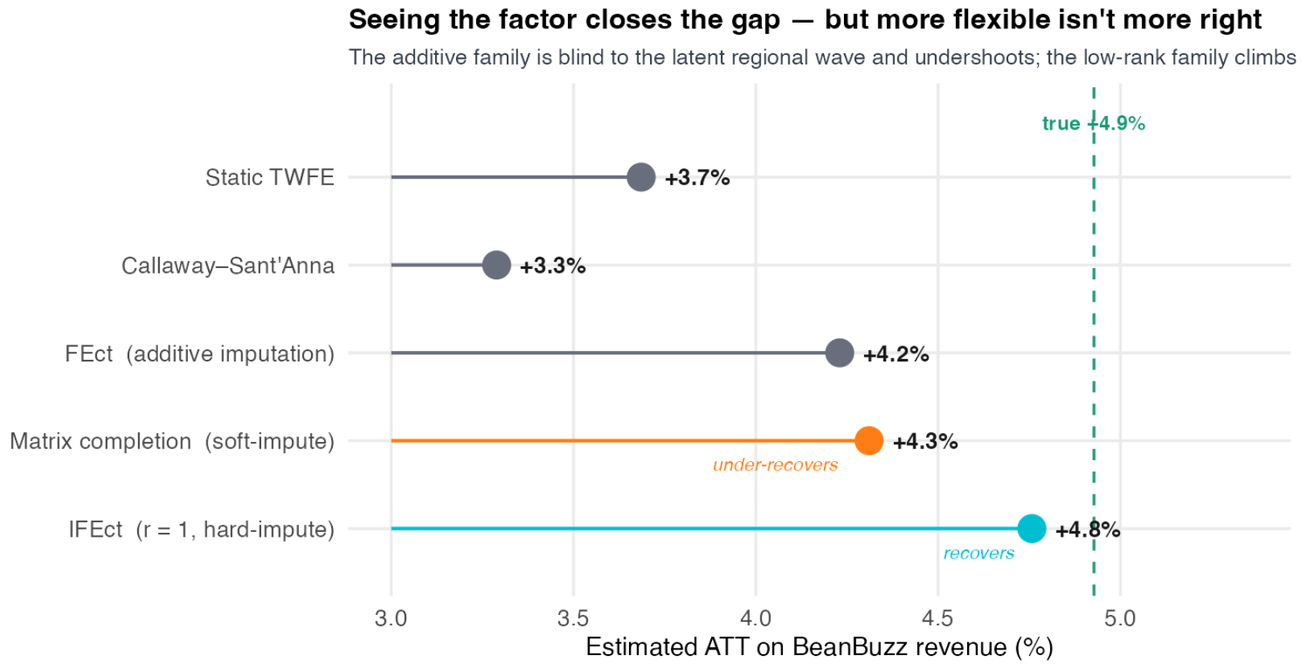
Both ladders open on the **same rung and the same score** — $r = 0$ *is* $\lambda = \sigma_{\max}$ *is* **FEct**, 0.00171 on each. That identity is the proof they share one ruler. And MC keeps **0.42, not its lowest 0.18**: fect's rule is *one-SE*, not *argmin*.

fect's own verdict: **"Recommended method through cross-validation: ife"** → r = 1, +4.76% — the headline, chosen with **no oracle.**

CV can RANK — nothing can ACQUIT

The race is run on 345 pre-launch weeks — not one treated post-launch cell is ever scored. Held-out fit ranks the candidates. It never certifies the winner.

Seeing the factor closes the gap — but more flexible isn't more right



Additive family (blind to the latent wave) undershoots; **IFEct +4.76% recovers**; but **matrix completion +4.31% under-recovers** — its nuclear norm shrank the very factor we needed. *More flexible ≠ more right.*

Sensitivity Analysis

Test 1 · Placebo — it fires, and it is small

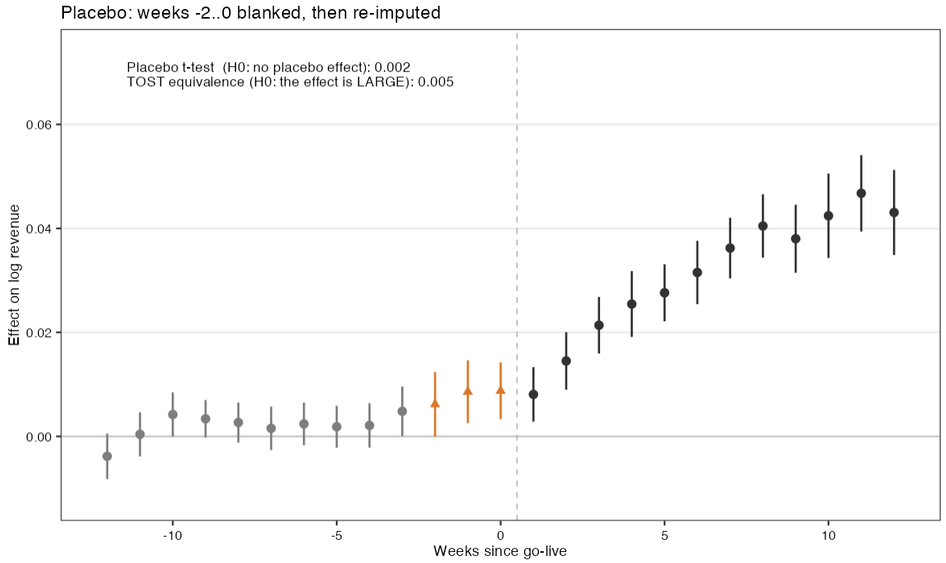
Why. The pre-period fit is *in-sample* — the model was **fitted on** those cells. So **blank three you *do* observe** and make it re-impute them **out of sample.**

H_0 · **t-test:** $\mathrm{ATT} = 0$ on the blanked window $\{-2, -1, 0\}$.

H_0 · **TOST** (*two one-sided tests*): $|\mathrm{ATT}| \geq \theta = 1.44$ pp.

```
out.p <- fect(..., method = "ife", r = 1,
              placeboTest    = TRUE,
              placebo.period = c(-2, 0))

plot(out.p, stats = c("placebo.p", "equiv.p"))
```



Both are true. Significance is not size.

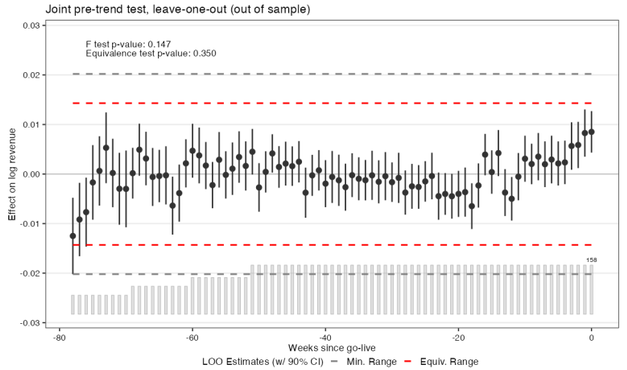
t-test $p = 0.002$ → reject "no placebo effect": the violation is REAL. · TOST $p = 0.005$ → reject "the effect is LARGE": the violation is SMALL.
A ruler, not a verdict: path $+0.62 \to +0.86 \to +0.88$ pp ⇒ week-to-week steps $0.240, 0.018$; the worst, 0.240 pp/week, denominates Test 3.

Test 2 · LOO pre-trend — nothing certifies

Why. Every pre-period so far was **in-sample** — the model was **fitted on it**. LOO re-imputes **every** pre-period t from a model that **never saw** t: **88 refits**, 79 tested.

```
out.loo <- fect(..., method = "ife", r = 1,
                loo = TRUE)

plot(out.loo, type = "equiv", loo = TRUE)
```



test	H_0	in-sample	LOO	what the number means
F	no pre-trend	0.126	0.147	*cannot reject* → **absence of evidence.** Never certifies.
F-equiv.	the pre-trend is LARGE	0.019	0.015	*rejects* → it **certifies** — but it **averages**, and averages **forgive.**
TOST	**SOME** pre-period is LARGE	**0.082**	0.350	*cannot reject* → **no certificate.** Intersection–union ⇒ the **worst of 79**, so it alone **quadruples.**

TOST certifies only if the grey band fits inside the red one. It does not.

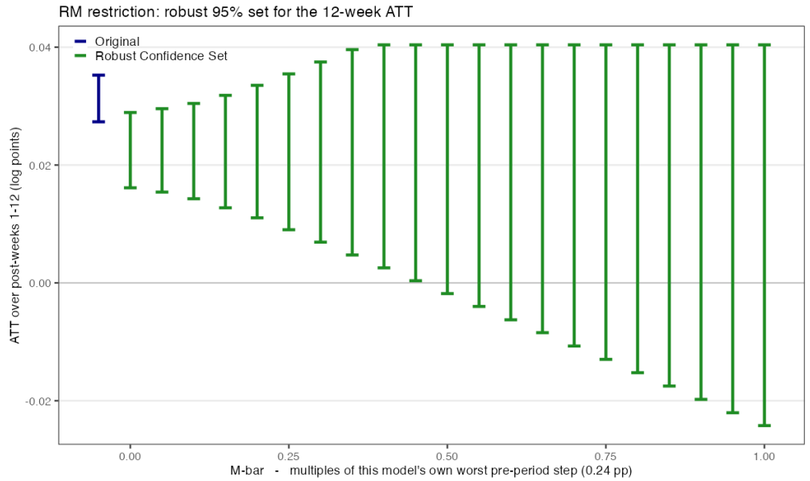
Red – the *bar*: Equiv. Range **1.43 pp** $= 0.36\,\sigma_{\mathrm{FEct}}$ **– what counts as negligible.** ***(This fit's own bar. Test 1's was 1.44 – a placebo fit has a different baseline MSE, so it cuts a different bar.)*** · **Grey – the *worst of the 79*:** Min. Range **2.02 pp** $= \max_t \left(|\hat{\delta}_t| + 1.645\,\mathrm{SE}_t\right)$ **– the largest violation you cannot rule out in the weakest pre-period. ⇒ Grey sits *outside* red: no certificate. The two bands are the TOST – *they are the 0.350, drawn.***

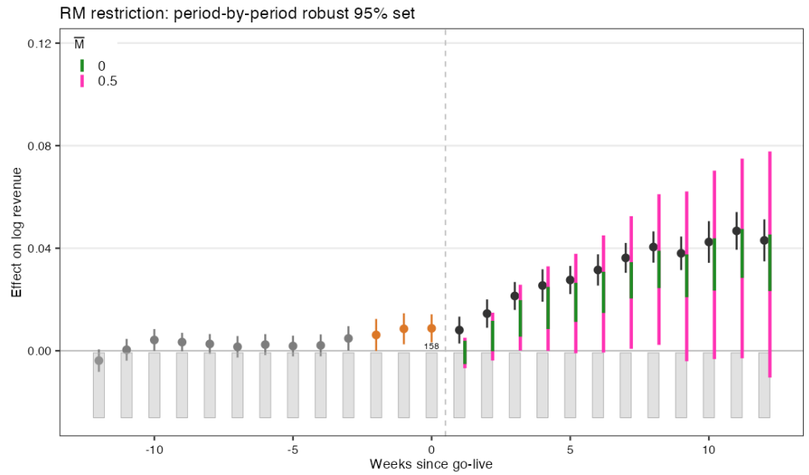
Test 3 · RM — the answer survives to $\bar{M}^* = 0.46$

Why. Don't *assume* parallel trends — **bound** the violation by the drift the placebo measured: $\Delta^{RM}(\bar{M}) = \{\delta : |\delta_{t+1} - \delta_t| \leq \bar{M}R\}$, with R = Test 1's ruler = **0.240 pp.**

```
s <- fect_sens(out.p,
       post.periods  = 1:12,
       Mbarvec       = seq(0, 1, 0.05),
       periodMbarvec = c(0, 0.5))

plot(s, type = "sens",    restrict = "rm")
plot(s, type = "sens_es", restrict = "rm")
```





The strictest setting is not the null — and a breakdown value is only half an answer

① **The freeze. Original +3.13% → at $\bar{M}$=0: +2.25%. And $3.13 - 0.88 = 2.25$ — it subtracted δ_0. *A null cannot move an estimate* ⇒ $\bar{M}$=0 freezes the violation; it does not zero it.**

② **The pair. $\bar{M}^*$=0.46 — of *what*? $\bar{M}$ is a multiple of R: 0.46×0.240 pp = 0.11 pp/week. Never quote one without the other.**

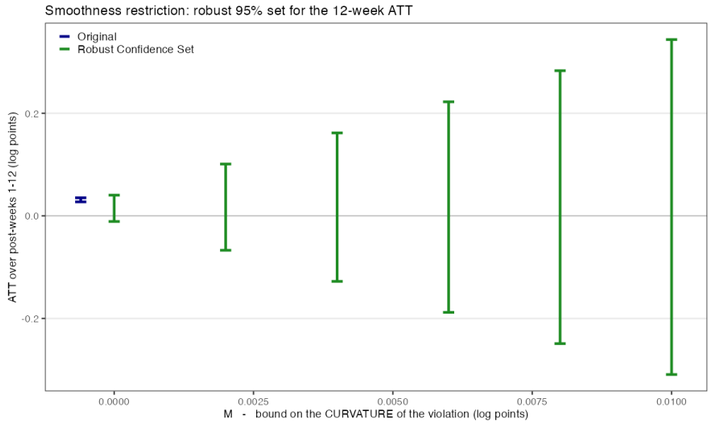
③ **Per week *(right)*. 2 of 12 weeks contain zero at $\bar{M}$=0; 9 of 12 at $\bar{M}$=0.5. A step allowance accumulates ⇒ the sets widen with the horizon.**

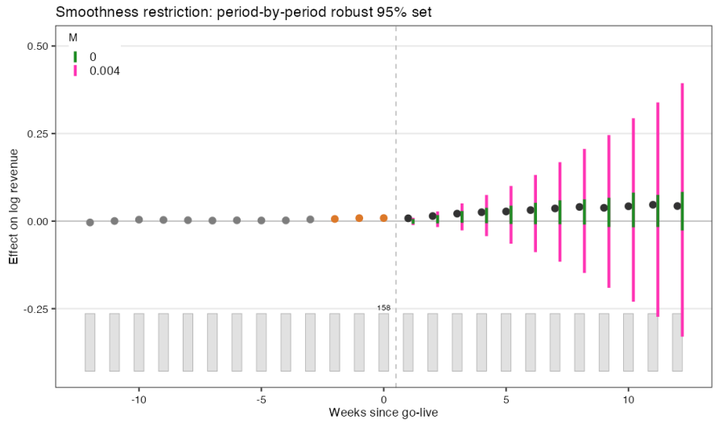
Test 4 · Smoothness — it dies at $M^* = 0$

Why. Bound the **curvature**, not the step:
$\Delta^{SD}(M) = \{\delta : |(\delta_{t+1} - \delta_t) - (\delta_t - \delta_{t-1})| \leq M\} \Rightarrow M{=}0$ makes the violation **exactly linear** — *not* zero.

```
s <- fect_sens(out.p,
       post.periods = 1:12,
       Mvec       = seq(0, 0.010, 0.002),
       periodMvec = c(0, 0.004))

plot(s, type = "sens",    restrict = "sm")
plot(s, type = "sens_es", restrict = "sm")
```





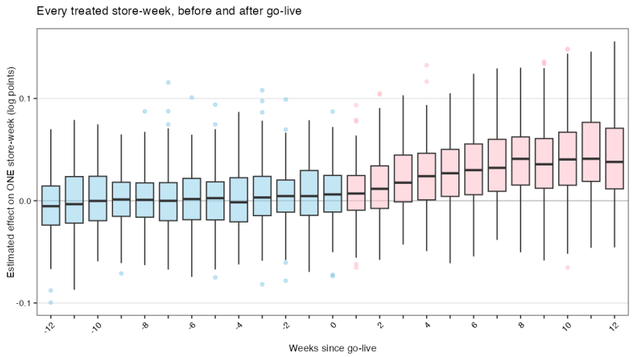
RM survives to 0.46. Smoothness dies at 0. Same panel, same fit, same estimand.

① $M^*{=}0$ – the ***strictest*** setting certifies ***nothing***. The set is already **[–1.10%, +4.04%]** at $M{=}0$: it contains zero, as do 12 of 12 weeks. The Original (navy, ***left***) does not. $\Delta^{SD}(0)$ says the violation is a straight line – ***not*** that it is absent – and a line fitted to a noisy pre-window and run out 12 weeks eats the whole effect.
② A step accumulates ***linearly***; a bend accumulates ***quadratically***. That is the only difference between the two restrictions – and it is worth 0.46 vs 0. Week 12 at $M{=}0.004$: [–33%, +39%] for a +4% effect.
⇒ THE RESTRICTION IS THE ASSUMPTION.

Effect Heterogeneity

Same fit, two axes, two illusions

Why. Don't aggregate at all — read the **cells**. *Is the app hurting some stores?* plot(f, type = "box")

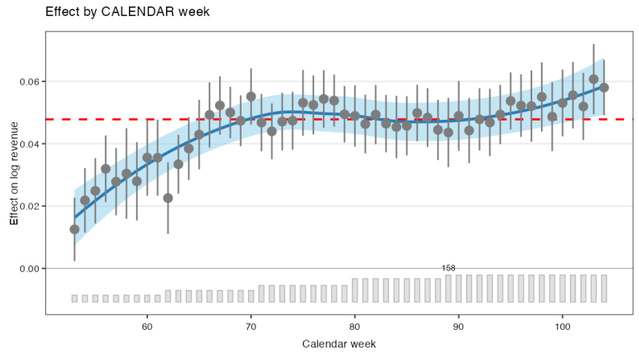


Looks like. **21.4**% of the 1,896 post cells read negative (week 1: **38.6**%) — *"switch the app off in the stores it's hurting."*

But. 48.0% of the **blue** cells do too — **before go-live, no app**. The unit FE is fitted on **exactly** those cells, so **half must land below. The null is 50, not 0** — the losers are **noise**.

Read it as ⇒ the LOCATION, never a box. Pre: median ≈ **0**. Post: it lifts and keeps lifting — +**0.7**% at launch → +**4.1**% by week 8. ***That climb is the effect.*** The *width* is **error, not heterogeneity**: it grows with the horizon (IQR **3.4** → **5.9 pp**).

Why. Fine — **aggregate**, and ask ***when***. *Does the app work better in some weeks?* plot(f, type = "calendar")



Looks like. +**1.25**% (wk 53) → +**5.80**% (wk 104) — a **4.8 pp** swing, as large as the ATT itself. *"It compounds: the app works better later in the year."*

But. No calendar is in it. Calendar time and **tenure** move together here: wk 53 is **40** stores at tenure **1**, wk 104 is **158** at mean tenure **35**. Rebuild the curve with **no calendar term**: inside the CI in **48 of 52** weeks. **It tracks the grey bars — the rollout.**

Read it as ⇒ DELIVERY, not modification. *What the live fleet actually earned each week* — **true, and the right plot for that.** High late because the fleet is **mature**. For *"when does it work better?"* you must **hold tenure fixed**; this plot **cannot**.

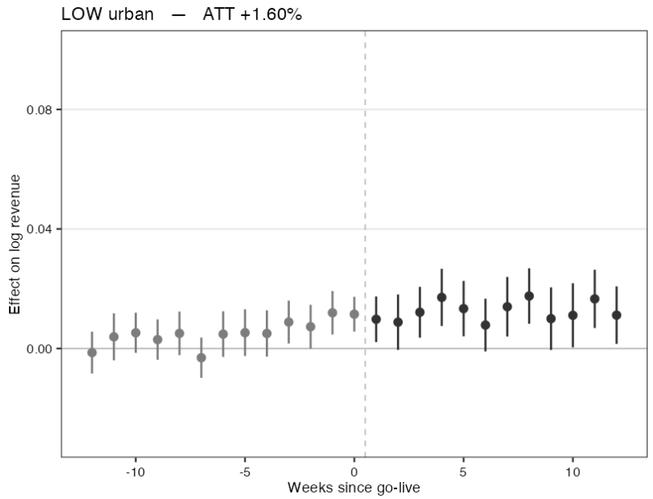
Route 1 — group=: split the panel, and let fect refit

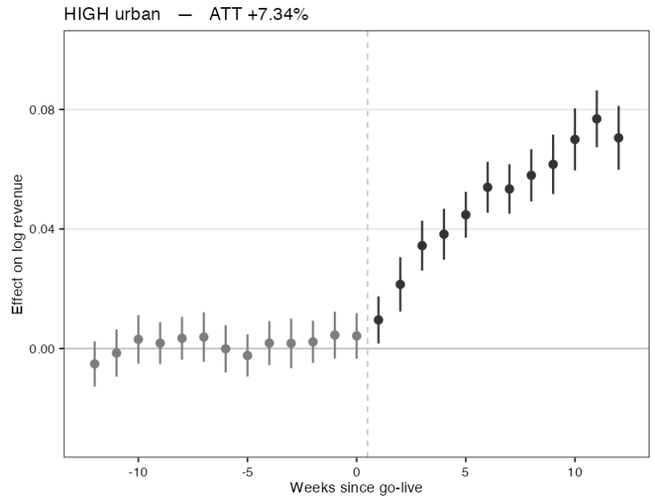
A discrete split needs no covariate in the model at all — there is no slope for the unit FE to absorb. fect **refits each group and bootstraps it.**

```
df$urban_tertile <- cut(u,
  quantile(u, c(0, 1/3, 2/3, 1)),
  include.lowest = TRUE,
  labels = c("Low urban",
             "Mid urban",
             "High urban"))

fg <- fect(log_rev ~ app_live,
  data = df,
  index = c("sid","week"),
  method = "ife", r = 1,
  force = "two-way", se = TRUE,
  group = "urban_tertile")

fg$est.group.att
```





group	ATT	S.E.
Low	+1.60%	0.21 pp
Mid	+4.61%	0.21 pp
High	+7.34%	0.21 pp

Read it as ⇒ same axis, two panels — the heterogeneity is the *gap between them*. Low is **flat at ~+1%**; High **ramps to ~+7%**. A **5.75-point spread**, SE **0.21 pp**, intervals nowhere near touching. **This is the aggregation that buys back what the cell forbade** — and its band is a **real bootstrap.**

Route 2 — type="hte": *build* the covariate

type="hte" smooths the **cell effect** against a covariate's **raw level** — so the covariate must be **in the model**, and it must clear **four hurdles:**

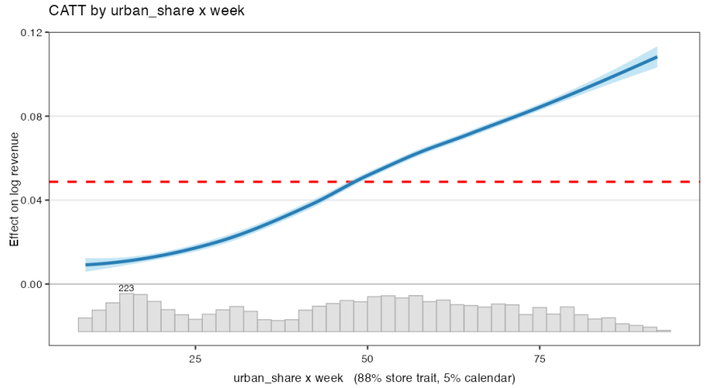
it must…	or else — and what it kills
1. vary within store	the unit FE eats it → *"unit-invariant"*. **Kills all 6 store attributes** — incl. urban_share
2. be nonzero on the *untreated* cells	**fect fits Y(0) *only* there.** urban_share × app_live is **identically 0** → collinear. *The moderator is only "on" where fect refuses to look.*
3. not be moved by the treatment	else it is a **bad control**. Kills mktg_spend: **+€251, t = 130**
4. not be a *clock*	else you are plotting the **tenure ramp**. Kills temp_c (95% week) and footfall_index (88%)

No raw covariate in the panel clears all four. So **build one:**

```
df$urban_x_week <- df$urban_share * df$week

fuw <- fect(log_rev ~ app_live + urban_x_week,
  data = df, index = c("sid","week"), method = "ife",
  r = 1, force = "two-way", se = TRUE)

plot(fuw, type = "hte", covariate = "urban_x_week")
```



It clears all four. Within-store sd **15.8** ①; sd **17.5** on the *untreated* cells ②; **deterministic in (store, week)** — the treatment **cannot** move it ③; and on the treated cells it is **88% store trait, 5% calendar** ④ *(the treated calendar spans 2×; urban_share spans 9× — **the product is urbanity, not time**).*

Read it as ⇒ a continuous gradient (+0.9% → +10.8%) that **agrees with** group= — at **two prices:** the covariate is *in the model*, so it **moved the estimator** (+4.78% → +4.87%); and the **ribbon is a loess SE over 5,606 cells treated as independent** — *it never touches the bootstrap*, even here with se = TRUE.

⇒ group= GIVES A BAND. type="hte" GIVES A SHAPE.

The completed assumption ledger

Every estimator in Part V assumed treatment was **not** set by past outcomes — *strict exogeneity*. Unpacked, it forbids **feedback**, **lagged-outcome dependence**, and **carryover** (Liu, Wang, and Xu 2024; Imai and Kim 2019).

Row	Assumption	Buys	Costs
0	**Strict exogeneity**	licenses *all* of Part V	no feedback / LDV / carryover
1 · S10	Parallel trends	cheap, additive, whole cohort	blind to a unit×time factor
2 · S11	Hull + loadings	a counterfactual at N=1	hull, non-overfit fit, weak inference
3 · S12	Low rank	recovers the latent factor	rank tuning; **restriction choice**; bands are frontier

If treatment **was** set by outcomes, none of this holds — you leave Part V for **g-methods** (MSM, PanelMatch). (Imai and Kim 2019)

The penalty, then the restriction

Every rung fills the same blank under a different assumption, and they diverge only on the grey cells you can't check. Estimation: the *penalty* is the assumption. Inference: the *restriction* is the assumption. *A point estimate without a defensible assumption and a band is not a finding.*

References

Athey, Susan, Mohsen Bayati, Nikolay Doudchenko, Guido Imbens, and Khashayar Khosravi. 2021. "Matrix Completion Methods for Causal Panel Data Models." *Journal of the American Statistical Association* 116 (536): 1716–30. https://doi.org/10.1080/01621459.2021.1891924.

Athey, Susan, and Guido W. Imbens. 2017. "The State of Applied Econometrics: Causality and Policy Evaluation." *Journal of Economic Perspectives* 31 (2): 3–32. https://doi.org/10.1257/jep.31.2.3.

Borusyak, Kirill, Xavier Jaravel, and Jann Spiess. 2024. "Revisiting Event-Study Designs: Robust and Efficient Estimation." *Review of Economic Studies* 91 (6): 3253–85. https://doi.org/10.1093/restud/rdae007.

Imai, Kosuke, and In Song Kim. 2019. "When Should We Use Unit Fixed Effects Regression Models for Causal Inference with Longitudinal Data?" *American Journal of Political Science* 63 (2): 467–90. https://doi.org/10.1111/ajps.12417.

Liu, Licheng, Ye Wang, and Yiqing Xu. 2024. "A Practical Guide to Counterfactual Estimators for Causal Inference with Time-Series Cross-Sectional Data." *American Journal of Political Science* 68 (1): 160–76. https://doi.org/10.1111/ajps.12723.

Xu, Yiqing. 2017. "Generalized Synthetic Control Method: Causal Inference with Interactive Fixed Effects Models." *Political Analysis* 25 (1): 57–76. https://doi.org/10.1017/pan.2016.2.

Thank you for your attention!

TUHH
Institute of
Entrepreneurship
STARTUPENGINEER.io



www.startupengineer.io/team/ihl/

christoph-ihl

christophihl