Synthetic Control Method

Professor Song Yao
Olin Business School

Customer Analytics

Diff-in-Diff Revisit

EXAMPLE: Seattle Soda Tax, using Bellevue as the Control Group

- One key assumption: Parallel Trends (another one is SUTVA)
  - The treated and control groups must follow similar trends in the outcome variable before the intervention.
  - Key Implication: Any post-treatment differences can be attributed to the treatment rather than pre-existing trends.
- Testing:
  - Visual inspection: Plot the outcome over time for both groups
  - Statistical tests: Regressions using pre-treatment periods
- However
  - This test may not always pass—invalid control
  - Even if it passes, we cannot be 100% sure (merely a necessary condition for being a good control unit)

Matching Revisit

EXAMPLE: Matching students with and without emails

- Two key requirements/assumptions
  - Match on observables: Observed features can accout for confounders
  - Sufficient overlap/balance in features/propensity scores between the two groups.
- These two requirements may not always satisfy
  - Poor balance/match and inaccurate causal inference results

Synthetic Control Method (SCM)

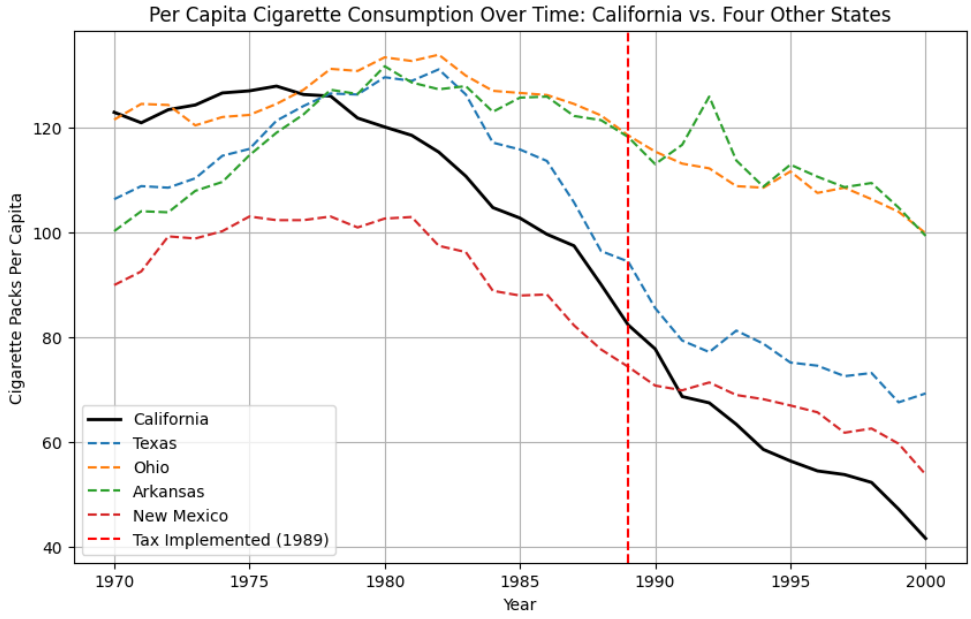
EXAMPLE: California Tabacco Tax 1989

- California Proposition 99 (the only state at the time)
  - Increasing cigarette taxes 1989 to discourage smoking
- Goal: Did *the tax* reduce per capita cigarette consumption?
  - Data: Per capita consumption (and other info) of each state from 1970-2000

	state	year	cigsale	lnincome	beer	age15to24	retprice
741	California	1989	82.400002	10.142313	23.700001	0.153525	126.400000
17	California	1970	123.000000	NaN	NaN	0.178158	38.799999
110	Connecticut	1972	110.800000	9.981137	NaN	0.166379	51.299999
224	Indiana	1975	162.399990	9.743661	NaN	0.183795	36.700001
83	Ohio	1972	124.400000	9.779579	NaN	0.175860	38.400002
330	West Virginia	1978	131.600010	9.687846	NaN	0.176877	53.900002
765	Wisconsin	1989	100.300000	9.965262	30.200001	0.149157	138.300000
631	Rhode Island	1986	136.300000	10.002565	26.000000	0.168093	101.800000
725	Virginia	1988	129.500000	10.078459	23.299999	0.160865	102.100000
963	North Carolina	1994	112.700000	9.966932	20.799999	NaN	135.899990

Synthetic Control Method (SCM)

EXAMPLE: California Tabacco Tax
California: Treated/Intervened; Other states: Control (often called donor units)



Test using Regressions with Interactions

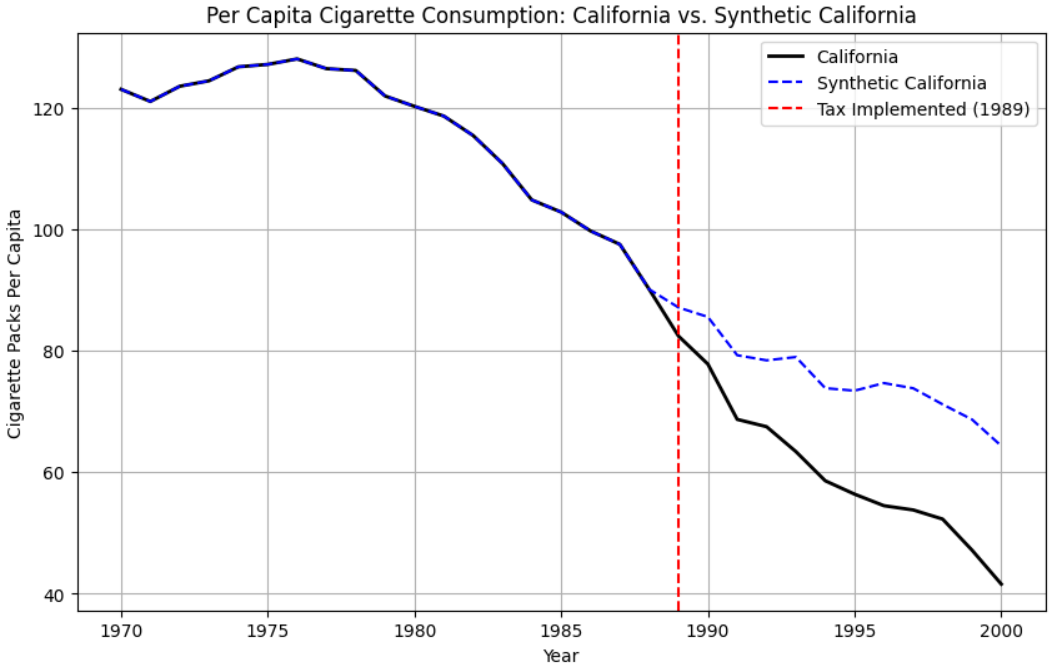
```
for state in donor_states:
    # Create subset of data for California and current control state
    state_data = df[df['state'].isin(['California', state])].copy()

    # Create dummy for California
    state_data['California'] = (state_data['state'] == 'California').astype(int)

    # Create formula for regression
    formula = 'cigsale ~ year*California'

    # Run regression
    model = sm.OLS.from_formula(formula, data=state_data)
    results = model.fit()
```

```
Significant year*California interactions by control state:
Rhode Island: True
Tennessee: True
Indiana: True
Nevada: True
Louisiana: True
.....
```

All interaction terms btw year and California are significant.
What does it imply?

Synthetic Control Method (SCM)

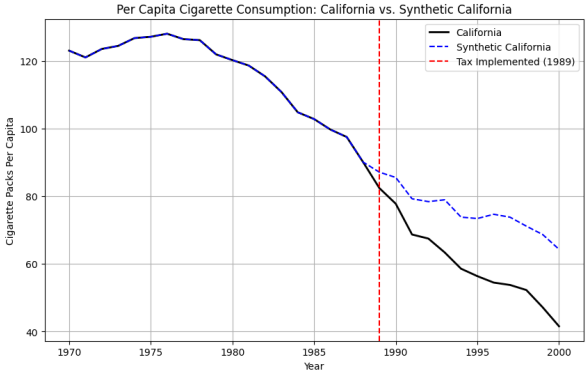
What can we do: Construct a "Synthetic California"



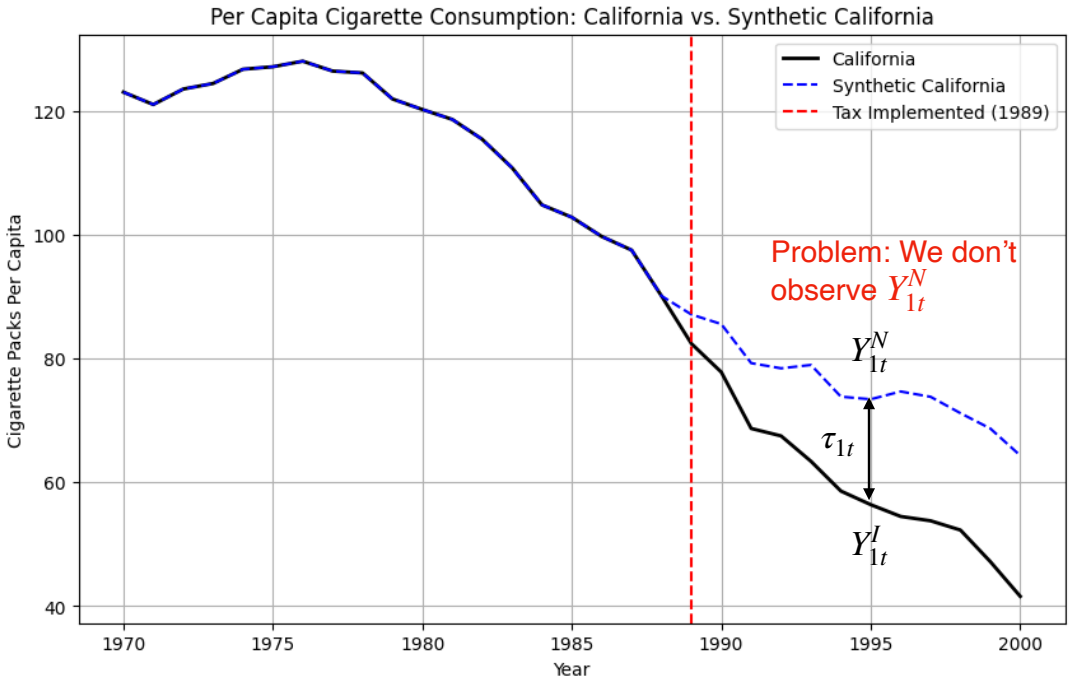
Core Idea of SCM

Consider the California Tabacco Tax example

- Lack proper control group
- Use other states to create a synthetic control (SC)
  - Weights determined by pre-treatment data
  - Minimize pre-treatment differences btw California and synthetic control
  - *Still need SUTVA to hold*
- Effect of the tax: Compare California's post-treatment consumptions with the Synthetic Control's consumptions



Formalize the Implementation of SCM



Approximate Y_{1t}^N as it is not observed

Use the donor pool to approximate

- For the weights $W = \{w_2, w_3, \ldots, w_{J+1}\}$
- The estimate of Y_{1t}^N is

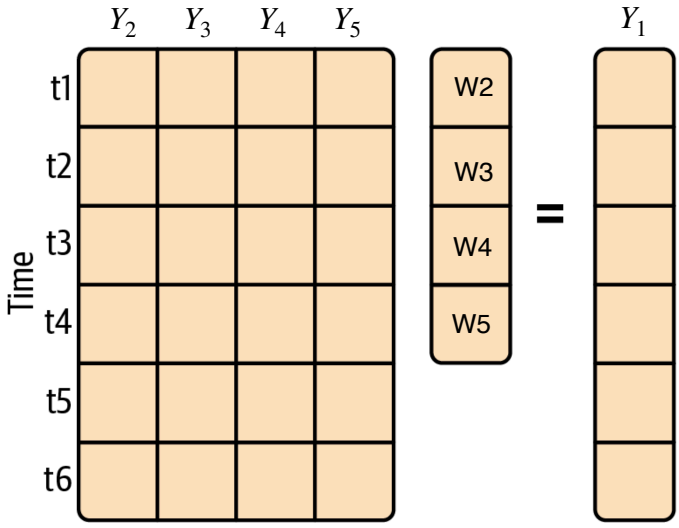
$$\hat{Y}_{1t}^N = \sum_{j=2}^{J+1} w_j Y_{jt},$$

  - Donors are not treated
  - Y_{jt}, observed and untreated outcomes of donors
  - Use the weighted sum of Y_{jt} to approximate Y_{1t}^N
- But, how to determine $W = \{w_2, w_3, \ldots, w_{J+1}\}$
  - If a donor has more similar outcomes to California, more weight
  - If not, less weight

Simple Solution: Obtain Weights using Regression on the *Pre-treatment* Data

A visual illustration

Imagine there are 6 *pre-treatment* periods and 4 donors ($j = 2,3,4,5$)



Hence, $W^* = \arg\min_W \|Y_1 - Y_{donor}W\|^2$

Go back to the California Tabacco Tax Example

state	year	cigsale	lnincome	beer	age15to24	retprice
Rhode Island	1970	123.9			0.18315789	39.299999
Tennessee	1970	99.800003			0.17804381	39.900002
Indiana	1970	134.60001			0.17651588	30.6
Nevada	1970	189.5			0.16155419	38.900002
Louisiana	1970	115.9			0.18518518	34.299999
⋮						
Colorado	1970	124.8			0.19095023	29.4
Minnesota	1970	104.3			0.17472412	39.099998
California	1970	123			0.17815833	38.799999
Texas	1970	106.4			0.18314135	40.400002
Kentucky	1970	155.8			0.18131015	28.299999

Go back to the California Tabacco Tax Example

Flip the data

```
# Create wide-format data for regression (rows: years, columns: states)
wide_data = df.pivot(index="year", columns="state", values="cigsale")
wide_data.head(10)

# Define California as the target and donors as predictors
y = wide_data.loc[wide_data.index < 1989, "California"].values
X = wide_data.loc[wide_data.index < 1989, donor_states].values
```

State	Alabama	Arkansas	California	Colorado	Connecticut	Delaware	Georgia
Year							
1970	89.800003	100.300003	123.000000	124.800003	120.000000	155.000000	109.900002
1971	95.400002	104.099998	121.000000	125.500000	117.599998	161.100006	115.699997
1972	101.099998	103.900002	123.500000	134.300003	110.800003	156.300003	117.000000
1973	102.900002	108.000000	124.400002	137.899994	109.300003	154.699997	119.800003
1974	108.199997	109.699997	126.699997	132.800003	112.400002	151.300003	123.699997
1975	111.699997	114.800003	127.099998	131.000000	110.199997	147.600006	122.900002
1976	116.199997	119.099998	128.000000	134.199997	113.400002	153.000000	125.900002
1977	117.099998	122.599998	126.400002	132.000000	117.300003	153.300003	127.900002
1978	123.000000	127.300003	126.099998	129.199997	117.500000	155.500000	130.600006
1979	121.400002	126.500000	121.900002	131.500000	117.400002	150.199997	131.000000

...

Weights obtained from Linear Regression

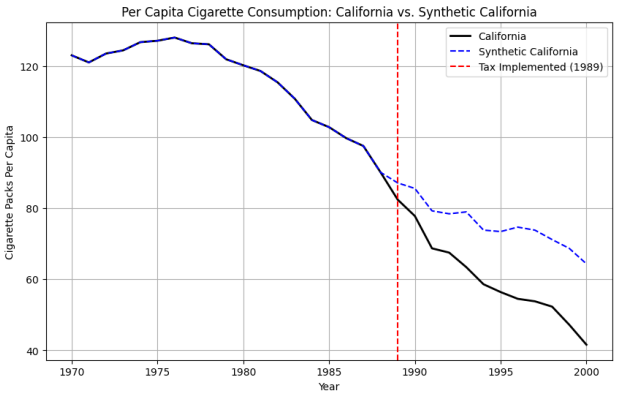
```
# Fit regression model to estimate weights for synthetic control
reg = LinearRegression(fit_intercept=False)
reg.fit(X, y)
weights = reg.coef_
print(weights)
```

```
[-0.0021349   0.03645321  0.10433373  0.15493477 -0.02684126  0.00902384
  0.05063973  0.10651942 -0.07509482  0.01084678  0.10736711  0.06824316
  0.0558359   0.01604695  0.06016016 -0.1328401  -0.03033175  0.14161729
  0.11170235  0.14081065  0.00952088  0.06679391  0.07497294 -0.06623474
 -0.00923212 -0.04691422 -0.02469848 -0.02558485 -0.07344015 -0.01096634
 -0.1117164   0.0834385   0.10906902 -0.09629999 -0.02751082  0.14305213
  0.08117342 -0.02890123]
```

Approximate Post-1989 California without the Tax Hike

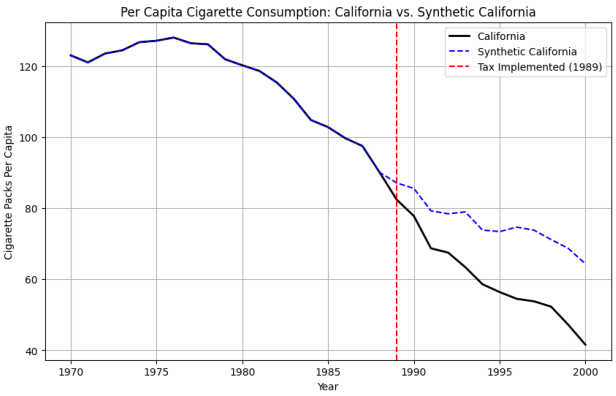
```
# Construct synthetic California using weighted donor states
synthetic_california_full = np.dot(wide_data[donor_states].values, weights)
```

```
Before 1989:
      California  Synthetic California
Year
1984  104.800003            104.800003
1985  102.800003            102.800003
1986   99.699997             99.699997
1987   97.500000             97.500000
1988   90.099998             90.099998

After 1989:
      California  Synthetic California
Year
1989   82.400002             87.104609
1990   77.800003             85.557417
1991   68.699997             79.252194
1992   67.500000             78.415649
1993   63.400002             78.956972
```



One problem though—Overfitting

```
Before 1989:
      California  Synthetic California
Year
1984  104.800003            104.800003
1985  102.800003            102.800003
1986   99.699997             99.699997
1987   97.500000             97.500000
1988   90.099998             90.099998
```



Two key reasons:

1. Too many covariates and not enough observations:

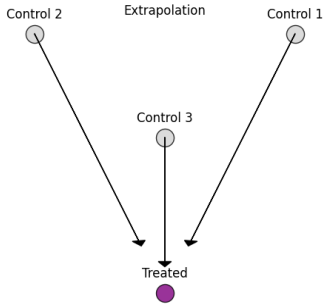
$$A + B = 5$$

$$A - B = 3$$

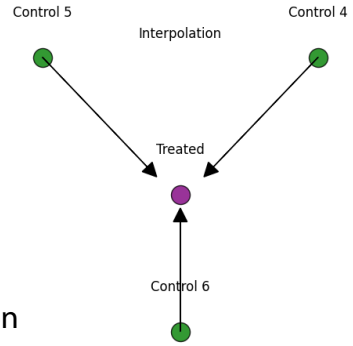
2. Extrapolation too much

Extrapolation from Donor Too Much

Extrapolation from training data often causes problem



"convex hull" or "convex combination"



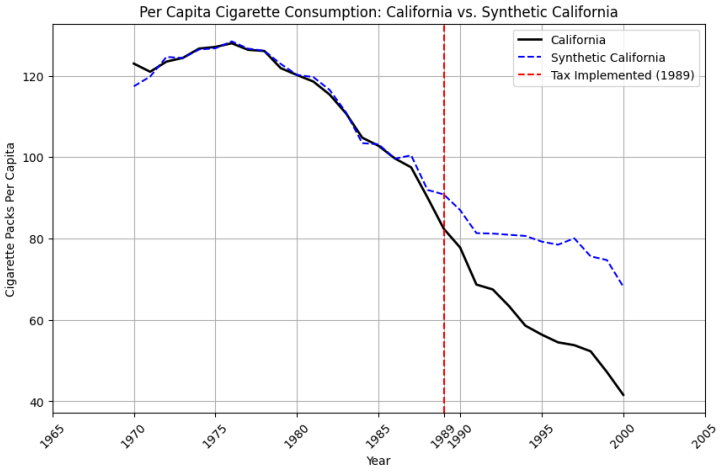
- Add two constraints to limit extrapolation
  - All weights fall between 0 and 1
  - All weights sum up to 1

Instead of Regression, Try Constrained Function

```python
def loss_w(W, X, y) -> float:
    ### This loss function is the sum of the squared errors between
    ### the treated unit and the synthetic treated unit
    ## W is the weights,
    ## X is the matrix of donor units,
    ## y is the treated unit
    return np.sqrt(np.mean((y - X.dot(W))**2))

def get_w(X, y):
    ### This function is used to get the weights
    ### It uses the fmin_slsqp function from the scipy.optimize module
    ### to find the weights that minimize the loss function
    ### The constraints are that the sum of the weights is 1
    ### and the weights are between 0 and 1

    w_start = [1/X.shape[1]]*X.shape[1] # initialize the weights to be equal

    weights = fmin_slsqp(
        # given X and y, calculate the loss function
        partial(loss_w, X=X, y=y),
        # initial weights set above
        np.array(w_start),
        # sum of weights is 1 where f_eqcons is an equality constraint
        # function that np.sum(x) - 1 to be as close as possible to 0
        f_eqcons=lambda x: np.sum(x) - 1,
        # bounds are between 0 and 1
        bounds=[(0.0, 1.0)]*len(w_start),
        # disp=False is used to suppress the output of the optimization
        disp=False)
    return weights
```

Results of the New Algorithm

```
Sum: 1.0000000000003681

Before 1989:
      California  Synthetic California
year
1984  104.800000            103.451323
1985  102.800000            103.185343
1986   99.699997             99.627299
1987   97.500000            100.465106
1988   90.099998             91.966274

After 1989:
      California  Synthetic California
year
1989   82.400002             90.840839
1990   77.800003             87.007573
1991   68.699997             81.334677
1992   67.500000             81.228925
1993   63.400002             80.933639
```



Treatment Effect (ATT, to be exact)

Difference between Observed and Synthetic Outcomes

```python
# Calculate treatment effect (difference between actual and synthetic)
diff_cigsale = wide_data['California'] - synthetic_california_new

# Calculate average treatment effect after implementation (post-1989)
ate_post = diff_cigsale[wide_data.index >= 1989].mean()
print("\nATT after 1989:", round(ate_post, 2), "packs per capita")

# Calculate percentage change
baseline = synthetic_california_new[wide_data.index >= 1989].mean()
percent_change = (ate_post / baseline) * 100
print(f"Percent Change: {round(percent_change, 1)}%")
```

```
ATT after 1989: -19.51 packs per capita
Percent Change: -24.4%
```

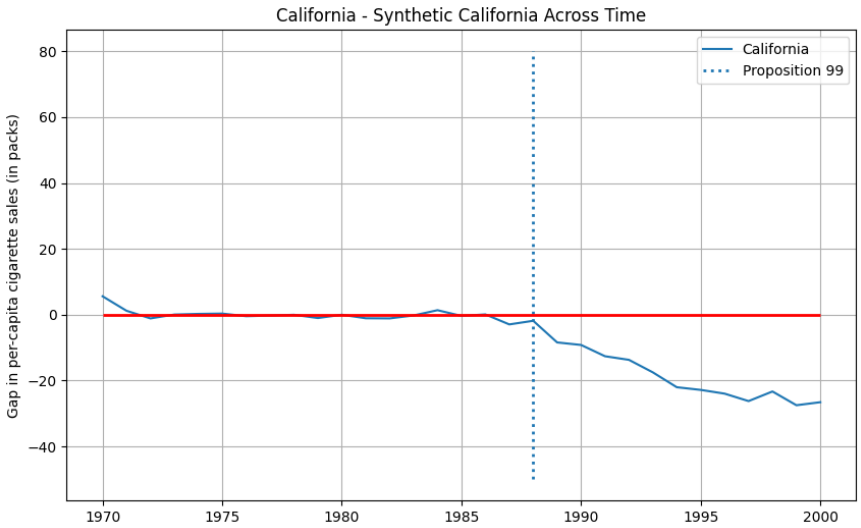
Distribution the Treatment Effect

Unconventional SE Estimation: "Placebo", inspired by Fisher's Exact Test

- For each state other than California
  - Pretend it was treated and other states are donors
  - Create its synthetic control
  - See what the "placebo" treatment effect is
- Collectively, these placebo treatment effects give *the distribution of sales if there was no treatment*
  - Evaluate California's effect against this distribution

The Probability of California's Effect Without the Tax

That is, if there was no tax, how likely to observe the effect of such a size?



The Probability of California's Effect Without the Tax

That is, if there was no tax, how likely to observe the effect of such a size?