What are the determinants of village-level public good expenditures in an Indian state?

Kumar H, Ayush Lahiri, Brian Holland

PPOL564: Status Update. 11.30.2022

Outline

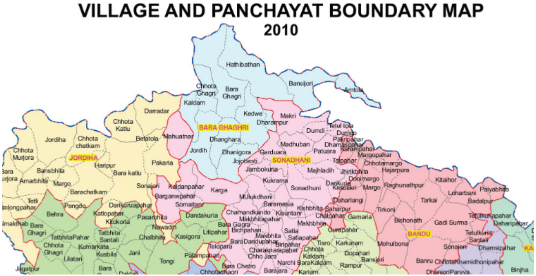
- Motivation
- Terminology
- Research Questions
- Data
- Methods
- Results thus far
- Limitations/Next Steps

Motivation

- Most developing nations have enacted measures to decentralize governance (Crook and Manor 1998)
- India's decentralization reforms in 1993 - Panchayats/ village level. Own spending and implementation capabilities.
- Since 1970s, rapid expansion of federal spending to favor two historically disadvantaged groups - **Scheduled Castes** and the **Scheduled Tribes**. (Banerjee and Somanathan, 2007)
- Social heterogeneity influences availability of public goods. Especially in societies where resource allocation is not automatic or rule bound. (Alesina et al, 1999)
- Some groups have more bargaining power, and ability to get resources from the state.

Motivation

- Administrative divisions in India
- State - District - Block - GP - Village
- Gram Panchayt (GP) - elected village council - can raise funds and implement projects.
- Villages nested within GPs.



Research Questions

- What factors influence the amount of spending per capita a village council gets?
  - How is it associated with initial infrastructure levels?
  - How does a village council's demographic composition affect it?

Data Sources

- **GP-level expenditure** Data from approved spending plans uploaded by GPs in Odisha, from 2015 - 2022.
- Courtesy Bhatia and Leighton (2022)

	District_Panchayat	Block_Panchayat	Village_Panchayat	Plan_Year	activity_name	activity_category	sector	estimated_cost	scheme_name
0	ANUGUL	ATHMALLIK	NAGOAN	2017-2018	Const. of Solar Street Light at Dudhianali Vil...	Gen	Rural electrification	500000.0	Fourteen Finance Commission
1	ANUGUL	ATHMALLIK	NAGOAN	2017-2018	Provision of Tanker water to Dudhianali, Antar...	Gen	Drinking water	200000.0	Fourteen Finance Commission
2	ANUGUL	ATHMALLIK	NAGOAN	2017-2018	Provision of Jalachatra (8 Places)	Gen	Administrative & Technical Support	70000.0	Fourteen Finance Commission

Data Sources

- **GP-level expenditure** Courtesy Bhatia and Leighton (2022)
- **Census of India, 2011** Village demographic and infrastructure data
- The Census provides data at the village level, nested within GPs.

	State Name	District Name	CD Block Name	Gram Panchayat Name	Village Name	Total Population of Village	Total Scheduled Castes Population of Village	Total Scheduled Tribes Population of Village	Hospital Allopathic (Numbers)	Gravel (kuchha) Roads (Status A(1)/NA(2))	Power Supply For All Users Summer (April-Sept.) per day (in Hours)
2	ODISHA	Bargarh	Paikamal	Chhetgaon	Chhatagaon	875	71.0	311.0	0.0	1.0	19.0
3	ODISHA	Bargarh	Paikamal	Chhetgaon	Ghuchapali	814	136.0	36.0	0.0	1.0	18.0
4	ODISHA	Bargarh	Paikamal	Chhetgaon	Kuturamal	965	183.0	41.0	0.0	2.0	18.0

- With no primary key(s) for either table, we cannot perform a JOIN.

Data Cleaning: Census

Variables can be combined into broad groups:Health, Education, Power/Elect

Group	Examples	No. of columns
Health	Community Health Centre (Numbers), Community Health Centre Doctors Total Strength (Numbers)	64
Education	'Govt Middle School (Numbers), Govt Arts and Science Degree College (Numbers)	28
Roads	National Highway (Status A(1)/NA(2)), Black Topped (pucca) Road (Status A(1)/NA(2)),	7
Finances	ATM (Status A(1)/NA(2)),Commercial Bank (Status A(1)/NA(2)), Cooperative Bank (Status A(1)/NA(2))	5
Power	Power Supply For All Users Summer (April-Sept.) per day (in Hours), Power Supply For All Users Winter (Oct.-March) per day (in Hours)	2

Data Processing: Census

- Variables can be combined into broad groups: Health, Education, Power/Electricity coverage, Roads and Financial Assets
- Creating unique IDs at the GP level

```
df_cen['id_cen'] = df_cen['District Name'] + "_" +
    df_cen['CD Block Name']+ "_" + df_cen['GP_Name']
```

Data Processing: Census

- Variables can be combined into broad groups: Health, Education, Power/Electricity coverage, Roads and Financial Assets
- Creating unique IDs at the GP level
- Aggregating village level data to the GP level

	State Name	District Name	CD Block Name	Village Name	Gram Panchayat Name	id_cen	educ_sum	fin_sum	roads_sum	power_hrs_sum
25437	ODISHA	Anugul	Anugul	Lokeipasi	Basala	anugul_anugul_basala	1.0	1.0	5.0	0.0
25438	ODISHA	Anugul	Anugul	Basala	Basala	anugul_anugul_basala	4.0	1.0	5.0	0.0
25439	ODISHA	Anugul	Anugul	Bherubania	Basala	anugul_anugul_basala	4.0	0.0	5.0	0.0

	id_cen	n_vill	educ_sum	fin_sum	roads_sum	power_hrs_sum
9	anugul_anugul_basala	3	9.0	2.0	15.0	0.0

Data Processing: Expenditure

- Creating unique IDs at the GP level
- We group at the unique ID, sector and year

	Village_Panchayat	Block_Panchayat	District_Panchayat	id_exp	sector	estimated_cost	sc_fund	st_fund	own_fund_exp
1225620	ANGARBANDHA	ANUGUL	ANUGUL	anugul_anugul_angarbandha	Drinking water	50000.0	0	0	0.0
1225621	ANGARBANDHA	ANUGUL	ANUGUL	anugul_anugul_angarbandha	Drinking water	30000.0	0	0	0.0
1225622	ANGARBANDHA	ANUGUL	ANUGUL	anugul_anugul_angarbandha	Drinking water	40000.0	0	0	0.0
1225623	ANGARBANDHA	ANUGUL	ANUGUL	anugul_anugul_angarbandha	Drinking water	120000.0	0	0	0.0
1225624	ANGARBANDHA	ANUGUL	ANUGUL	anugul_anugul_angarbandha	Drinking water	338542.0	0	0	0.0
...	...	...	...	...	...	...	...	...	...

	id_exp	sector	Plan_Year	tot_spend	n_act	sc_fund	st_fund	own_spend
36	anugul_anugul_angarbandha	Roads	2022-2023	251388.0	2	0	0	0.0
21	anugul_anugul_angarbandha	Education	2022-2023	1084032.0	9	0	0	0.0
12	anugul_anugul_angarbandha	Drinking water	2020-2021	1183391.0	16	0	0	0.0

Data Processing - Fuzzy Merging

- Fuzzy matching: Why ?

	District_Panchayat	Block_Panchayat	Village_Panchayat	id_exp
55617	BARGARH	GAISILET	FUIRINGIMAL	bargarh_gaisilet_fuiringimal

	District Name	CD Block Name	Gram Panchayat Name	id_cen
394	Bargarh	Gaisilet	Phiringimal	bargarh_gaisilet_phiringimal

Data Processing - Fuzzy Merging

- The 'recordlinkage' package could not be used, since there were no common variables between the two datsets except the place names.
- **Fuzzy Matching**: How?
- Calculate score for how similar two strings are - based on edit distance. Higher Score means more similarity.
- Options: **Fuzz ratio**, partial ratio, token sort ratio, token set ratio.

Data Processing - Fuzzy Merging

- Fuzzy matching : Fuzz ratio, partial ratio, token sort ratio, token set ratio
  - Fuzz ratio:

```
fuzz.ratio("NEW YORK METS", "NEW YORK MEATS") = 96
```

Data Processing - Fuzzy Merging

- Fuzzy matching : Fuzz ratio, partial ratio, token sort ratio, token set ratio
  - Partial ratio:

```
fuzz.ratio("NEW YORK METS", "NEW YORK YANKEES") = 75
fuzz.partial_ratio("NEW YORK METS", "NEW YORK YANKEES
    ") = 69
```

Data Processing - Fuzzy Merging

- Fuzzy matching : Fuzz ratio, partial ratio, token sort ratio, token set ratio
  - Token sort ratio:

```
fuzz.ratio("New York Mets vs Atlanta Braves", "Atlanta Braves vs New York Mets") = 45

fuzz.partial_ratio("New York Mets vs Atlanta Braves", "Atlanta Braves vs New York Mets") = 45

"new york mets vs atlanta braves" —> "atlanta braves mets new vs york"

fuzz.token_sort_ratio("New York Mets vs Atlanta Braves", "Atlanta Braves vs New York Mets") = 100
```

Data Processing - Fuzzy Merging

- Fuzzy matching : Fuzz ratio, partial ratio, token sort ratio, token set ratio
  - Token set ratio:

```
s1 = "angels mariners vs"
s2 = "anaheim angeles angels los mariners of seattle vs"

[SORTED_INTERSECTION] = ["angels mariners"]
t0 = [SORTED_INTERSECTION]
t1 = [SORTED_INTERSECTION] + [SORTED_REST_OF_STRING1]
t2 = [SORTED_INTERSECTION] + [SORTED_REST_OF_STRING2]

fuzz.ratio(t0, t1) = 90
fuzz.ratio(t0, t2) = 46
fuzz.ratio(t1, t2) = 50
fuzz.token_set_ratio(t1,t2) = 90
```

Data Processing - Fuzzy Merging

- Removing perfect matches between the two datasets:
- Number of unique IDs in expenditure data: 6903
- Number of unique IDs in census data: 6241
- Number of unique IDs that perfectly match already between two datasets: 2997
- In the time gap between the census and the expenditure records, 192 new GPs were created, so we just simply drop those since they don't exist in both.

Data Processing - Fuzzy Merging

- Removing perfect matches between the two datasets
- Fuzzy matching, by setting threshold score at 80

```
metric = fuzz.ratio
thresh = 80
ca = np.array(imperfect_exp) #array of unique IDs in expenditure data that did not have a perfect match as found above
cb = np.array(imperfect_cen) #array of unique IDs in census data that did not have a perfect match as found above


def parallel_fuzzy_match(idxa,idxb):
    return [ca[idxa],cb[idxb],metric(ca[idxa],cb[idxb])]
results = Parallel(n_jobs=-1,verbose=1)(delayed(parallel_fuzzy_match)(idx1, idx2) for idx1 in range(len(ca)) for idx2 in range(len(cb)) \
                    if(metric(ca[idx1],cb[idx2]) > thresh))
scores_ratio = pd.DataFrame(results,columns = ["id_exp","id_cen","Score"])
```

Data Processing - Fuzzy Merging

- Removing perfect matches between the two datasets
- Using the Fuzzywuzzy Python package, we find match scores between unique IDs in the two datasets.
  - Set threshold score = 80 i.e only keep match pairs with high scores
- For each GP ID, keep only matching ID with the highest score

	id_exp	id_cen	Score
0	kendujhar champua kutariposi	kendujhar champua kasipal	83
1	kendujhar champua kutariposi	kendujhar champua jajapasi	81
2	kendujhar champua kutariposi	kendujhar champua kutaripasi	96

Data Processing - Fuzzy Merging

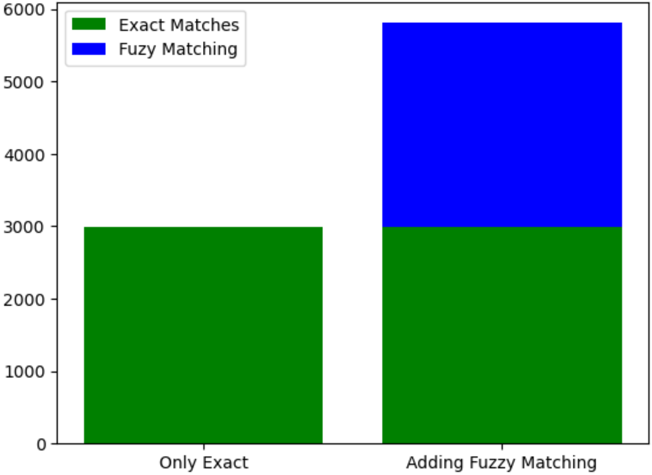
- Removing perfect matches between the two datasets
- Using Fuzzywuzzy, we find match scores between unique IDs in the two datasets.
  - Set threshold score $= 80$ i.e only keep match pairs with high scores
- For each GP ID, keep only matching ID with the highest score
- Still, false matches persist.

	id_exp	id_cen	Score
3472	ganjam bhanjanagar mudulipalli	ganjam bhanjanagar baiballi	84.0
3590	koraput jeypore gadapadara	koraput jeypur godopodara	82.0
3677	khordha bhubaneswar ranasinghpur	khordha bhubaneswar paikerapur	81.0

- After manual evaluation, we only keep match pairs with score > 93

Data Processing: Fuzzy Merging Results

- Merging with approximate string matching almost doubles the number of rows in merged Dataframe (149k to 293k, of total 332k)



Data Processing

- Aggregating GP level public goods for: Health, Education, Power/Electricity coverage, Roads and Financial Assets
- Creating GP level IDs and grouping by IDs to convert data from village level to GP level
- Fuzzy matching
- Quantifying variables as percentages of population and per-capita spending

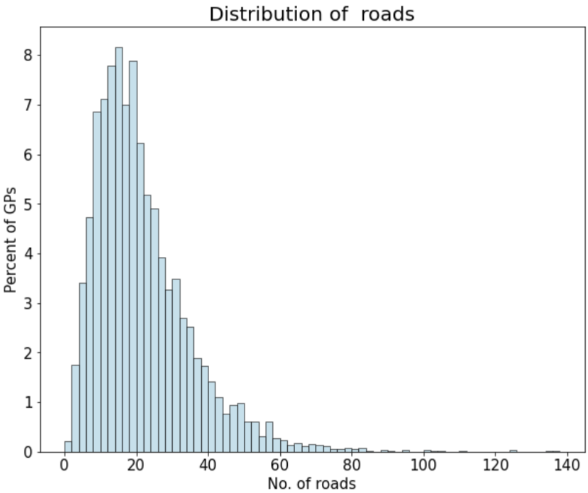
Data Processing

- Variables can be combined into broad groups: Health, Education, Power/Electricity coverage, Roads and Financial Assets
- Aggregating village level data to the GP level for the Census
- Aggregate activity level data to sectors for Expenditure data.
- Fuzzy matching and merging
- Creating final variables for analysis

```
#GP level demographic percentages
df['st_perc'] = df['Total Scheduled Tribes Population of
    Village']/(df['Total Population of Village'])

#GP per capita public goods
df['educ_per_capita'] = df['educ_sum']/df['Total Population of
    Village']

# Ranking GPs
df["educ_pc_rank"] = df["educ_pcap"].rank(ascending=False)

#Per capita expenditure
df['spend_per_capita] = df['Total Spending']/df['Total
    Population of Village']

```

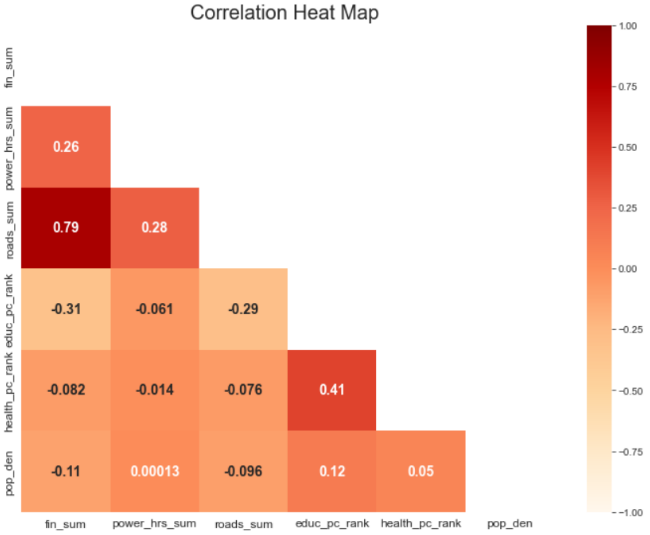
Analysis: Visualization

- Large variation in levels of public goods across villages



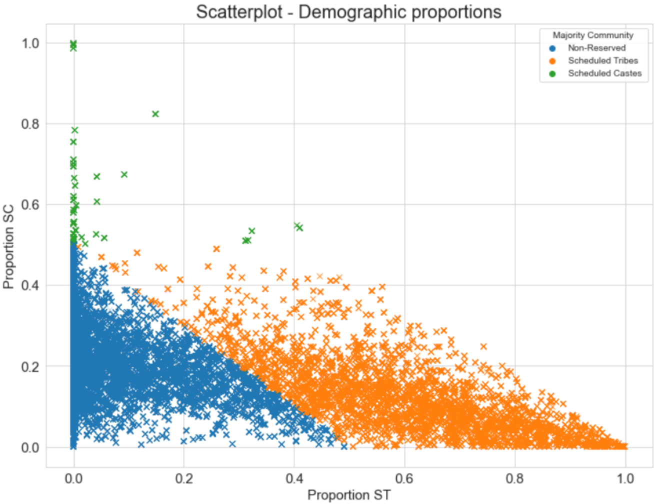
Analysis: Visualization

- Amenities are correlated with each other



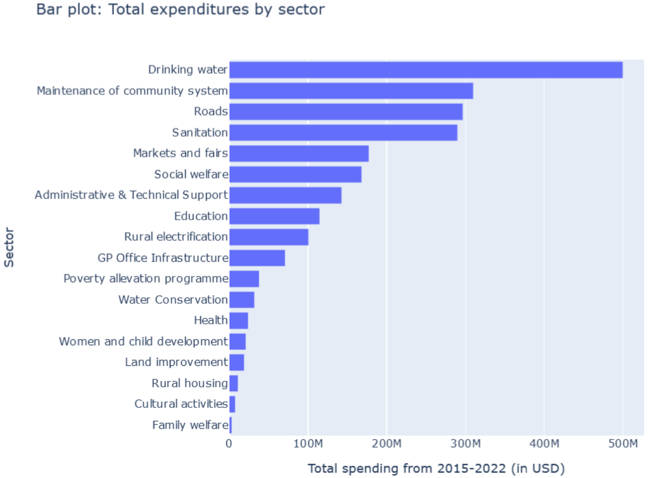
Analysis: Visualization

- Settlement patterns - STs concentrated in much more homogeneous villages.



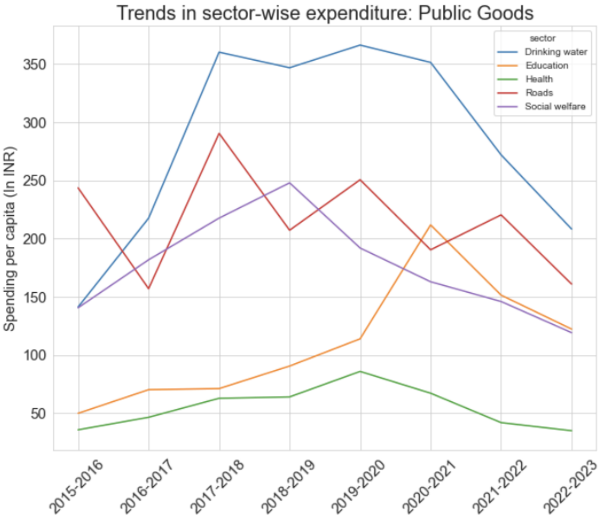
Analysis: Visualization

- Activities in some categories received higher amounts of total spending



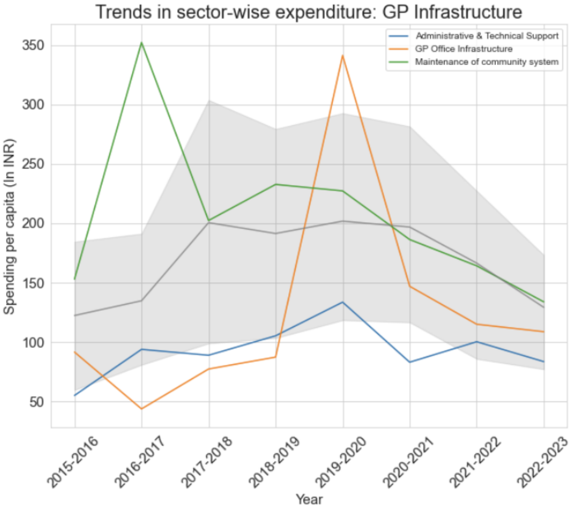
Analysis: Visualization

- Per capita spending for public goods fluctuates, partly due to federally mandated schemes/expenditure.



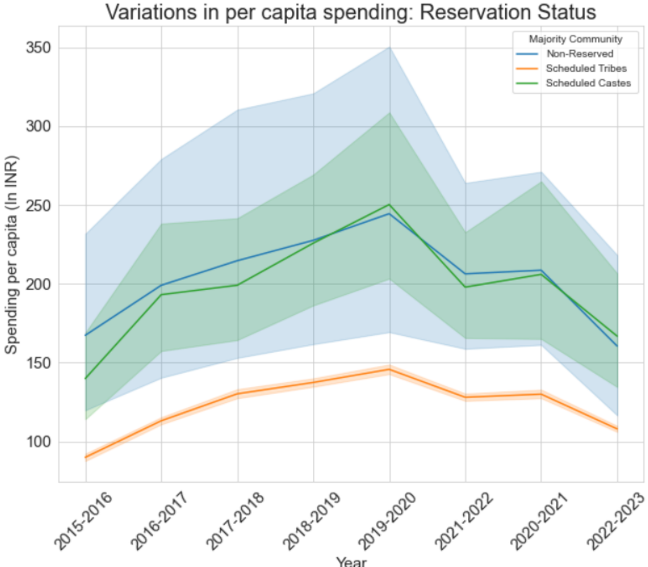
Analysis: Visualization

- Per capita spending to enhance state capability is greater.



Analysis: Visualization

- Per capita spending is dramatically lower for ST villages.



Regression of spending

- OLS Regression specification is given below:

$$y_i = \beta_0 + X_i\beta_1 + \epsilon_i \tag{1}$$

y_i = spending per capita in village i
X_i = [Roads, Power, financial assets, education, health]

- Spending per capita negatively associated with more heterogeneous population.

Limitations and next steps

- OLS regression has difficulty identifying true effects - very low R-squared.
- Distinguish federal (mandated) spending from own spending.
- Fuzzy merging still requires a manual check.