

Behavioral Health Population-based Measures

Table of contents

Social Determinants of Behavioral Health	2
Rate of Unstable Housing and Homelessness	2
Homelessness (Unsheltered and Sheltered)	2
Housing Cost Burden	3
“Doubled-Up” Households	3
Residential Instability	4
Eviction Rates	5
Unemployment Rate	6
Access to Healthcare (Mental Health and Substance Use Treatment)	6
Part 1: How to Get the Data from HRSA	7
Code to Get Health Insurance Coverage Data	8
Community Social Cohesion and Support Networks	9
Part 1: Download the Election Data	11
Part 2: Code to Calculate Voter Turnout	11
Part 1: Download the Response Rate Data	12
Part 2: Code to Extract the Response Rate	13
Part 1: How to Get the Non-profit Data	14
Part 2: Code to Calculate Non-profit Density	14
Explanation of the PCA	17
Adverse Childhood Experiences (ACEs) Score	18
Outcome-based Behavioral Health Measures	19
Fatal Overdose Rate (by substance, e.g., fentanyl)	20
Suicide Rates (by age group, gender, and race)	21
Emergency Department Visits for Behavioral Health Crises	22
Part 1: Guidance for Querying ESSENCE	22
Part 2: R Code for Analyzing the Exported Data	23
Mental Health-Related Civil Commitment Rates	24
Criminal Justice System Involvement (mental health-related arrests)	24
Substance Use Disorder Treatment Admission and Completion Rates	25
How to Find the Data	25
Process and Systems-based Measures	26
Wait Times for Behavioral Health Appointments	26
Patient/Consumer Satisfaction with Behavioral Health Services	26
Retention Rates in Behavioral Health Programs	26

Social Determinants of Behavioral Health

These measures are often considered causal or impactful because they reflect underlying social and environmental factors that drive behavioral health outcomes.

Rate of Unstable Housing and Homelessness

This is a crucial measure for Multnomah County, as it's a known driver of both mental health and substance use issues. Homelessness can lead to increased stress, trauma, and a lack of access to care, while substance use and mental illness can contribute to housing instability, creating a vicious cycle.

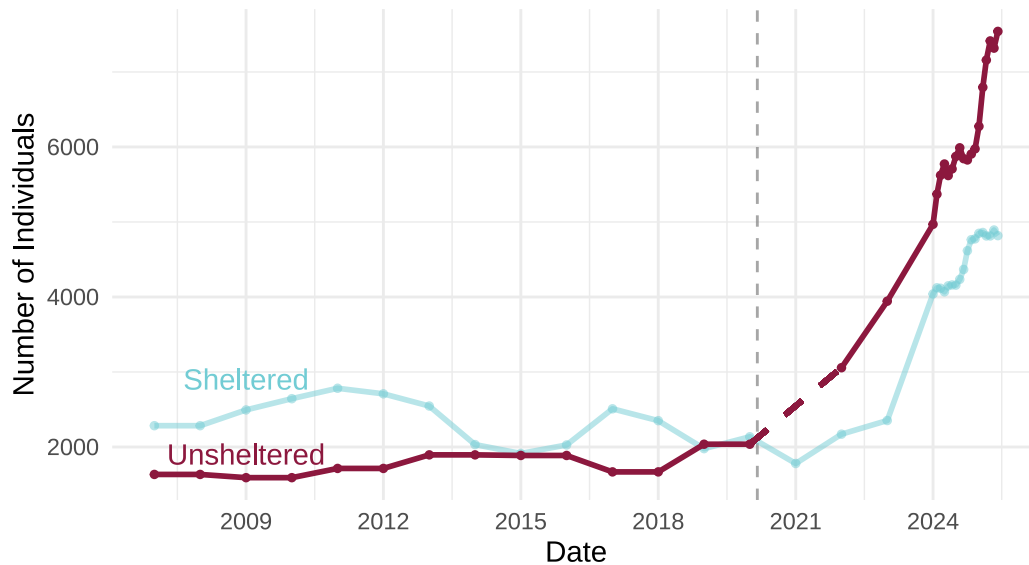
Unstable housing can be measured in several ways, often by combining multiple indicators to get a more complete picture. The key is to move from a single metric to a continuum of housing instability.

Homelessness (Unsheltered and Sheltered)

This is the most visible form of unstable housing.

- **Definitions:**
 - **Unsheltered:** People living on the streets, in vehicles, or in other places not meant for human habitation.
 - **Sheltered:** People in emergency shelters, transitional housing, or domestic violence shelters.
- **Primary Source:** [Multnomah County Homeless Services Department](#). Shows monthly counts of how many people are experiencing homelessness in Multnomah County, broken down by sheltered vs. unsheltered. Date range available: January 2024 to present (monthly).
- **Secondary Source:** [U.S. Department of Housing and Urban Development \(HUD\)](#). They publish the Annual Homeless Assessment Report (AHAR) to Congress, which includes point-in-time counts of homelessness. Date range available: 2007-2024 (yearly).
- **Potential Source:** The ideal source would be the actual 'By Name List of people experiencing homelessness' maintained by the Multnomah County Homeless Services Department. This would provide the most accurate and up-to-date counts, including demographic breakdowns.
- **How to Access:** This data is saved in `data/homelessness_counts_multnomah_2007-present.csv` in this project repository. This file needs to be updated by hand.

Trends in Homelessness Counts Over Time, Multnomah County



Housing Cost Burden

A household is considered “cost-burdened” if it spends more than 30% of its income on housing costs, including rent/mortgage and utilities. This can be further broken down into severely cost-burdened if the amount exceeds 50%. This is a huge indicator of housing insecurity.

- **Primary Source:** U.S. Census Bureau’s American Community Survey (ACS). This is the gold standard for this type of socioeconomic data at small geographic levels.
- **How to Access:** The `tidycensus` package is perfect for this. It provides a straightforward way to query the Census API and get data directly into a tidy `tibble`.
- **Key Variables:** Table B25070 (Gross Rent as a Percentage of Household Income) and B25091 (Mortgage Status by Selected Monthly Owner Costs as a Percentage of Household Income). We would sum the categories for households paying 30% or more.

“Doubled-Up” Households

This refers to people living with friends or family because they have no other place to go. This is a hidden form of homelessness and a major sign of housing instability.

- **Primary Source:** U.S. Census Bureau’s American Community Survey (ACS). This is the gold standard for this type of socioeconomic data at small geographic levels.
- **How to Access in R:** The `tidycensus` package is perfect for this. It provides a straightforward way to query the Census API and get data directly into a tidy `tibble`.
- **Key Variables:** This is harder to measure directly, but a common proxy is to use relationship data from tables like B11016 (Household Type by Household Size) to identify households with multiple families or non-relatives.

Residential Instability

This is measured by the number of times a person or family moves within a specific period (e.g., two or more moves in a year). It indicates a lack of stable residence.

- **Primary Source:** U.S. Census Bureau's American Community Survey (ACS). This is the gold standard for this type of socioeconomic data at small geographic levels.
- **How to Access in R:** The `tidycensus` package is perfect for this. It provides a straightforward way to query the Census API and get data directly into a tidy tibble.
- **Key Variables:** Table B07003 (Geographical Mobility in the Past Year) is a direct measure of how many people have moved within the last year.

```
# -----  
# 1. Define the variables we need from the ACS  
# -----  
# We're using table B07003: "GEOGRAPHICAL MOBILITY IN THE PAST YEAR"  
# Denominator: Total population 1 year and over  
# Numerator: Total population minus those who lived in the same house  
acs_instability_vars <- c(  
  total_pop_over1 = "B07003_001E",  
  same_house      = "B07003_004E"  
)  
  
# -----  
# 2. Get data for Multnomah County from the Census API  
# -----  
multnomah_instability_raw <- get_acs(  
  geography = "county",  
  state     = "OR",  
  county    = "Multnomah",  
  variables = acs_instability_vars,  
  year      = 2023,  
  survey    = "acs5",  
  output    = "wide"  
)  
  
# -----  
# 3. Calculate the final residential instability measure  
# -----  
dat_residential_instability <- multnomah_instability_raw %>%  
  # Calculate the number of people who moved in the last year  
  mutate(  
    n_moved = total_pop_over1 - same_house,  
  
    # Calculate the percentage of the population that moved  
    pct_moved = (n_moved / total_pop_over1) * 100  
  ) %>%  
  # Select and rename for a clean final table
```

```

select(
  year = NAME,
  total_pop = total_pop_over1,
  n_moved,
  pct_moved
) %>%
# Clean up the year column
mutate(
  year = str_extract(year, "\\d{4}-\\d{4}")
)

# -----
# 4. Display the final data table
# -----
knitr::kable(
  dat_residential_instability,
  digits = 1,
  caption = "Residential Instability in Multnomah County (2019-2023 ACS)",
  col.names = c("Data Period", "Total Population (1+)", "Number Who Moved", "Percent Who Moved")
)

```

Eviction Rates

The number of formal evictions filed in the court system is a direct measure of housing loss and a strong predictor of future homelessness.

- Go to the Eviction Lab's data download page: <https://evictionlab.org/data/>
- Find the download link for their **county-level data**. It will be a single CSV file containing data for all counties in the U.S.
- Download the file and save it into your R project directory. For organizational purposes, I recommend creating a subfolder named `data/` and placing the file there (e.g., `your-project-folder/data/counties.csv`).

```

# -----
# 1. Load the downloaded data into R
# -----
# Make sure the file path is correct for where you saved the CSV file.
# The `read_csv` function is part of the tidyverse.
eviction_raw_data <- read_csv("data/counties.csv")

# -----
# 2. Filter for Multnomah County and select key variables
# -----
# The GEOID for Multnomah County is 41051. Filtering by GEOID is the most
# reliable way to select the correct county.
dat_eviction_rates <- eviction_raw_data %>%
  filter(GEOID == 41051) %>%

```

```

select(
  year,
  name,
  renter_occupied_households = `renter-occupied-households`,
  eviction_filings = `eviction-filings`,
  eviction_filing_rate = `eviction-filing-rate`
)

# -----
# 3. Display the final data table
# -----
# This will show a time series of eviction filing rates for the county.
knitr::kable(
  dat_eviction_rates,
  digits = 2,
  caption = "Eviction Filing Rates in Multnomah County (The Eviction Lab)",
  col.names = c("Year", "County", "Renter Households", "Eviction Filings", "Eviction Filing Rate")
)

```

Unemployment Rate

High unemployment is directly correlated with increased rates of depression, anxiety, and substance use disorders. It's an economic indicator with significant behavioral health consequences.

- **Primary Source:** **U.S. Bureau of Labor Statistics (BLS)**. They manage the Local Area Unemployment Statistics (LAUS) program.
- **How to Access in R:** There are a few packages, like `blscrapeR`, or we can build a query to hit the BLS API directly. The API is free to use.
- **Key Variables:** The API allows you to query for the unemployment rate (seasonally and not seasonally adjusted) for specific areas, including Multnomah County.

Access to Healthcare (Mental Health and Substance Use Treatment)

This isn't just about the number of clinics but also about patient-to-provider ratios, geographic distribution of services, and the number of people with health insurance. Audits have shown that Multnomah County's ability to serve those with serious mental illness is limited, making this an especially impactful measure.

- **Primary Source:** **Health Resources and Services Administration (HRSA)** and the **National Plan and Provider Enumeration System (NPPES)**.
- **How to Access in R:** HRSA has a data portal with downloadable datasets and some API endpoints. The NPPES NPI registry can be downloaded as a large CSV file. We could write code to download and process these files.
- **Key Variables:** HRSA data includes information on Health Professional Shortage Areas (HPSAs). We can use this to calculate provider-to-population ratios for mental health professionals.

Part 1: How to Get the Data from HRSA

1. Navigate to the HRSA Data Warehouse

- Go to the main portal: <https://data.hrsa.gov/>

2. Find the HPSA Data

- Use the search bar and look for “Health Professional Shortage Areas” or “HPSA”. You should find datasets related to designated HPSAs. Look for a nationwide file that includes mental health designations.

3. Download the Data File

- Find the option to download the data, usually as a **CSV file**.
- Save this file into your **data/** subfolder. For this example, let’s assume the file is named **mental_health_hpsa.csv**.

Part 2: Code to Process the HPSA Data

```
# -----  
# 1. Load the downloaded HRSA data  
# -----  
# Make sure the file path is correct for where you saved the CSV.  
hpsa_raw_data <- read_csv("data/mental_health_hpsa.csv")  
  
# -----  
# 2. Filter for Mental Health Shortage Areas in Multnomah County  
# -----  
# We'll filter the data to our specific county and focus only on areas that  
# are currently "Designated" as having a shortage.  
dat_multnomah_hpsa <- hpsa_raw_data %>%  
  # Column names may vary slightly depending on the exact file you download.  
  # Use janitor::clean_names() to standardize them first.  
  janitor::clean_names() %>%  
  filter(  
    hpsa_discipline_class == "Mental Health",  
    common_county_name == "Multnomah",  
    hpsa_status == "Designated"  
  ) %>%  
  # Select the most relevant columns for our report  
  select(  
    hpsa_name,  
    designation_type = hpsa_designation_type,  
    hpsa_score,  
    population = hpsa_population,  
    percent_below_poverty = hpsa_percent_of_population_with_income_at_or_below_100_percent_fpl  
  )  
  
# -----  
# 3. Display the final data table
```

```
# -----
knitr::kable(
  dat_multnomah_hpsa,
  digits = 1,
  caption = "Designated Mental Health Professional Shortage Areas (MHPSAs) in Multnomah County",
  col.names = c("Shortage Area Name", "Designation Type", "HPSA Score", "Population in Area",
)
)
```

Code to Get Health Insurance Coverage Data

```
# -----
# 1. Define the variables we need from the ACS
# -----
# We're using Subject Table S2701: "SELECTED CHARACTERISTICS OF HEALTH INSURANCE"
# This table conveniently provides a pre-calculated percentage for us.
acs_insurance_vars <- c(
  total_population = "S2701_C01_001E",
  percent_uninsured = "S2701_C05_001E" # Note: This variable is for Percent Uninsured now
)

# -----
# 2. Get data for Multnomah County from the Census API
# -----
multnomah_insurance_raw <- get_acs(
  geography = "county",
  state = "OR",
  county = "Multnomah",
  variables = acs_insurance_vars,
  year = 2023,
  survey = "acs5",
  output = "wide"
)

# -----
# 3. Clean up the final table
# -----
dat_insurance_coverage <- multnomah_insurance_raw %>%
  # Select and rename the columns for a clean output
  select(
    year = NAME,
    total_population,
    percent_uninsured
  ) %>%
  # Clean up the year column
  mutate(
    year = str_extract(year, "\\d{4}-\\d{4}")
  )
)
```



```
# -----
# 4. Display the final data table
# -----
knitr::kable(
  dat_insurance_coverage,
  digits = 1,
  caption = "Health Insurance Coverage in Multnomah County (2019-2023 ACS)",
  col.names = c("Data Period", "Total Population", "Percent Uninsured")
)
```

Community Social Cohesion and Support Networks

This can be measured through surveys or community engagement data. A strong sense of community and social support can act as a protective factor against behavioral health crises.

- **Primary Source:** This is difficult to measure with administrative data. The best sources are often surveys. For a good proxy, we can use data from **County Health Rankings & Roadmaps**.
- **How to Access in R:** They provide downloadable data files. We could also check their source data (e.g., the U.S. Religion Census) to see if there are APIs available.
- **Key Variables:** They have a measure called “Social Associations,” which is the number of membership associations per 10,000 people.

Re-creating the Penn State Northeast Regional Center for Rural Development Social Capital Index.

1. Association Density (Establishments per 1,000 people)

The original study used the Census Bureau’s **County Business Patterns (CBP)** dataset to count the number of establishments for specific North American Industry Classification System (NAICS) codes. We can do the same thing using the **tidycensus** package, which has a function specifically for the CBP. The latest available CBP data is for **2022**.

Here’s the code to get the counts for all the association types, sum them up, and calculate the rate per 1,000 people.

```
# -----
# 1. Define the NAICS codes for all association types
# -----
association_naics_codes <- c(
  "813110", # Religious organizations
  "813410", # Civic and social associations
  "813910", # Business associations
  "813940", # Political organizations
  "813920", # Professional organizations
  "813930", # Labor organizations
  "713950", # Bowling centers
  "713940", # Fitness and Recreational Sports Centers
  "713910", # Golf Courses and Country Clubs
)
```

```

"711211" # Sports Teams and Clubs
)

# -----
# 2. Get establishment counts from County Business Patterns (CBP)
# -----
multnomah_associations <- get_cbp(
  geography = "county",
  state = "OR",
  county = "Multnomah",
  year = 2022,
  naics = association_naics_codes
) %>%
  # Sum the establishments from all the different NAICS codes
  summarize(total_associations = sum(ESTAB, na.rm = TRUE))

# -----
# 3. Get the 2022 population for the denominator
# -----
multnomah_population <- get_acs(
  geography = "county",
  state = "OR",
  county = "Multnomah",
  variables = "B01003_001", # Total Population
  year = 2022,
  survey = "acs1" # 1-year ACS is best for a single year's population
)

# -----
# 4. Combine and calculate the association density
# -----
dat_association_density <- multnomah_associations %>%
  mutate(
    population = multnomah_population$estimate,
    association_rate_per_1000 = (total_associations / (population / 1000))
  )

# Display the result
knitr::kable(
  dat_association_density,
  digits = 2,
  caption = "Association Density in Multnomah County (2022)",
  col.names = c("Total Associations", "Population", "Associations per 1,000 People")
)

```

2. Voter Turnout

- **Data Source:** The **MIT Election Data and Science Lab**. They compile the most comprehensive county-level election results.
- **Most Recent Data:** The 2020 Presidential Election.
- **Process:** We'll download their data file, load it into R, and filter for Multnomah County. We'll then divide the total votes cast by the voting-age population for that year (which we can get from `tidycensus`).

Part 1: Download the Election Data

1. Navigate to the MIT Election Lab Dataverse

- Go to their data repository: <https://electionlab.mit.edu/data>

2. Find and Download the File

- Locate the dataset for “U.S. President 1976-2020”.
- Download the county-level data file. It is typically named `1976-2020-president.csv`.
- Save this file into your `data/` subfolder.

Part 2: Code to Calculate Voter Turnout

This R code will load the election file, calculate the total votes cast in Multnomah County in 2020, fetch the voting-age population from the 2020 Census, and compute the final turnout rate.

```
# -----
# 1. Load the downloaded MIT election data
# -----
# Make sure the file path is correct for where you saved the CSV.
election_raw_data <- read_csv("data/1976-2020-president.csv")

# -----
# 2. Calculate Total Votes for Multnomah County in 2020
# -----
# The FIPS code for Multnomah County is 41051.
multnomah_votes_2020 <- election_raw_data %>%
  filter(year == 2020, county_fips == 41051) %>%
  # Sum the votes for all candidates to get the total votes cast
  summarize(total_votes = sum(candidatevotes, na.rm = TRUE)) %>%
  pull(total_votes) # pull() extracts the single value from the data frame

# -----
# 3. Get the 2020 Voting-Age Population (VAP)
# -----
# We'll use get_decennial() for the official 2020 Census count.
# The variable P4_001N is the total population 18 years and over.
multnomah_vap_2020 <- get_decennial(
  geography = "county",
  state = "OR",
```

```

county = "Multnomah",
variables = "P4_001N",
year = 2020
) %>%
pull(value) # pull() the single value

# -----
# 4. Calculate and display the final turnout rate
# -----
dat_voter_turnout <- tibble(
  year = 2020,
  total_votes = multnomah_votes_2020,
  voting_age_population = multnomah_vap_2020,
  voter_turnout_rate = (total_votes / voting_age_population) * 100
)

knitr::kable(
  dat_voter_turnout,
  digits = 2,
  caption = "Voter Turnout in Multnomah County (2020 Presidential Election)",
  col.names = c("Year", "Total Votes Cast", "Voting-Age Population", "Voter Turnout Rate (%)")
)

```

3. Census Response Rate

- **Data Source:** The **U.S. Census Bureau** publishes final self-response rates for the 2020 Census.
- **Most Recent Data:** 2020.
- **Process:** We can download the official response rate file, load it into R, and look up the rate for Multnomah County. This is a straightforward lookup.

Part 1: Download the Response Rate Data

1. Navigate to the Census Bureau’s Data Page

- The data is available on the “2020 Census Self-Response Rates” page. You can find the final data files here: <https://www.census.gov/library/visualizations/interactive/2020-census-self-response-rates.html>

2. Find and Download the File

- Look for the link to the **final** response rates, usually available as a single downloadable file containing data for all counties.
- Download the file and save it into your `data/` subfolder. Let’s assume you’ve named it `2020_census_response_rates.csv`.

Part 2: Code to Extract the Response Rate

This R code will load the file, clean the column names, and then filter down to the exact value for Multnomah County.

```
# -----  
# 1. Load the downloaded Census response rate data  
# -----  
# Make sure the file path is correct for where you saved the CSV.  
response_rate_raw <- read_csv("data/2020_census_response_rates.csv")  
  
# -----  
# 2. Filter for Multnomah County and get the final rate  
# -----  
# The FIPS code for Multnomah County is 41051. The column with the final  
# cumulative self-response rate is often named "CRRALL".  
dat_census_response <- response_rate_raw %>%  
  # Standardize column names for easier use  
  janitor::clean_names() %>%  
  # Filter to the correct county using its unique FIPS ID  
  filter(county_fips == 41051) %>%  
  # Select the key columns for our report  
  select(  
    county = county_name,  
    final_response_rate = crrall  
  )  
  
# -----  
# 3. Display the final data table  
# -----  
knitr::kable(  
  dat_census_response,  
  digits = 2,  
  caption = "Final Self-Response Rate for Multnomah County (2020 Census)",  
  col.names = c("County", "Final Self-Response Rate (%)")  
)
```

4. Non-profit Density

- **Data Source:** The **National Center for Charitable Statistics (NCCS)**, as noted in the original study. This data is based on IRS records of tax-exempt organizations.
- **Most Recent Data:** Data is available for recent years (e.g., 2022 or 2023).
- **Process:** This requires downloading the IRS Business Master File for exempt organizations from the NCCS website. We would then load this large file and filter it for non-international organizations located in Multnomah County, then calculate the rate per 10,000 people.

This measure captures a broad range of community organizations that the previous establishment counts might miss. The authoritative source for this is the **National Center for Charitable**

Statistics (NCCS), which compiles data from IRS records on all registered tax-exempt organizations.

Part 1: How to Get the Non-profit Data

1. Navigate to the NCCS Website

- Go to the NCCS data download page: <https://nccs.urban.org/data-statistics/get-data>

2. Create a Free Account

- You will need to register for a free account to access the datasets.

3. Find and Download the File

- Look for the **IRS Business Master File (BMF)**. This is a complete list of all active tax-exempt organizations.
- Download the most recently available version for the **entire U.S.** It will be a very large CSV file.
- Save this file into your **data/** subfolder. For our example, we'll assume the file is named **bmf-2024.csv**.

Part 2: Code to Calculate Non-profit Density

This R code will load the large BMF file, perform the necessary filtering, get the population for the same year, and calculate the final density rate. The original study excluded non-profits with an “international approach,” which we’ll do by filtering out organizations with an NTEE code starting with “Q” (International, Foreign Affairs).

```
# -----  
# 1. Load the downloaded NCCS Business Master File  
# -----  
# This file is large and may take a moment to load.  
# Make sure the filename matches what you downloaded.  
bmf_raw_data <- read_csv("data/bmf-2024.csv")  
  
# -----  
# 2. Filter, count non-profits, and exclude international orgs  
# -----  
# The FIPS code for Multnomah County is 41051.  
# NTEE codes starting with "Q" are for international organizations.  
multnomah_nonprofits <- bmf_raw_data %>%  
  filter(  
    FIPS == 41051,  
    !str_starts(NTEE_CD, "Q") # Exclude if NTEE code starts with Q  
  ) %>%  
  summarize(non_profit_count = n()) %>%  
  pull(non_profit_count)
```

```

# -----
# 3. Get the 2023 population for the denominator
# -----
# We'll use the most recent population data to match the BMF data.
multnomah_population_2023 <- get_acs(
  geography = "county",
  state = "OR",
  county = "Multnomah",
  variables = "B01003_001",
  year = 2023,
  survey = "acs1"
) %>%
  pull(estimate)

# -----
# 4. Calculate and display the final non-profit density
# -----
dat_nonprofit_density <- tibble(
  year = "2023-2024",
  non_profit_count = multnomah_nonprofits,
  population = multnomah_population_2023,
  rate_per_10k = (non_profit_count / (population / 10000))
)

knitr::kable(
  dat_nonprofit_density,
  digits = 2,
  caption = "Non-profit Density in Multnomah County",
  col.names = c("Data Period", "Non-profit Count", "Population", "Non-profits per 10,000 People")
)

```

5. Let's build the final index.

To do this using **Principal Component Analysis (PCA)**, we first need to build a basis for comparison. PCA works by finding the main patterns in a dataset with multiple entries. Therefore, to calculate a meaningful score for Multnomah County, we first need to gather the data for **all 36 counties in Oregon**.

This process will create a “ruler” against which we can measure Multnomah County. The final code block below is comprehensive; it will:

1. Gather the four factors for every county in Oregon.
2. Combine and clean the data.
3. Perform the PCA to create the index.
4. Show you where Multnomah County ranks in the state.

```

# -----
## Part 1: Gather Data for All Oregon Counties

```

```

# -----
# We'll re-run our previous steps, but for all counties in Oregon (geography="county", state="OR")

### 1a: Association Density (2022)
association_naics_codes <- c("813110", "813410", "813910", "813940", "813920", "813930", "713910")
or_associations_raw <- get_cbp(geography = "county", state = "OR", year = 2022, naics = association_naics_codes)
or_population_2022 <- get_acs(geography = "county", state = "OR", variables = "B01003_001", year = 2022)

factor1_associations <- or_associations_raw %>%
  group_by(GEOID, NAME) %>%
  summarize(total_associations = sum(ESTAB, na.rm = TRUE), .groups = 'drop') %>%
  left_join(select(or_population_2022, GEOID, population = estimate), by = "GEOID") %>%
  mutate(association_rate = (total_associations / (population / 1000))) %>%
  select(GEOID, NAME, association_rate)

### 1b: Voter Turnout (2020)
election_raw_data <- read_csv("data/1976-2020-president.csv")
or_vap_2020 <- get_decennial(geography = "county", state = "OR", variables = "P4_001N", year = 2020)

factor2_turnout <- election_raw_data %>%
  filter(year == 2020, state_po == "OR") %>%
  group_by(GEOID = county_fips, NAME = county_name) %>%
  summarize(total_votes = sum(candidatevotes, na.rm = TRUE), .groups = 'drop') %>%
  mutate(GEOID = as.character(GEOID)) %>%
  left_join(select(or_vap_2020, GEOID, vap = value), by = "GEOID") %>%
  mutate(turnout_rate = (total_votes / vap) * 100) %>%
  select(GEOID, turnout_rate)

### 1c: Census Response Rate (2020)
response_rate_raw <- read_csv("data/2020_census_response_rates.csv")
factor3_response <- response_rate_raw %>%
  janitor::clean_names() %>%
  filter(str_starts(county_fips, "41")) %>% # 41 is Oregon's state FIPS code
  select(GEOID = county_fips, response_rate = crrall) %>%
  mutate(GEOID = as.character(GEOID))

### 1d: Non-profit Density (~2024)
bmf_raw_data <- read_csv("data/bmf-2024.csv")
or_population_2023 <- get_acs(geography = "county", state = "OR", variables = "B01003_001", year = 2023)

factor4_nonprofits <- bmf_raw_data %>%
  filter(STATE == "OR", !str_starts(NTEE_CD, "Q")) %>%
  group_by(GEOID = FIPS) %>%
  summarize(non_profit_count = n(), .groups = 'drop') %>%
  mutate(GEOID = as.character(GEOID)) %>%
  left_join(select(or_population_2023, GEOID, population = estimate), by = "GEOID") %>%
  mutate(nonprofit_rate = (non_profit_count / (population / 10000))) %>%

```



```

select(GEOID, nonprofit_rate)

# -----
## Part 2: Combine and Clean the Data
# -----
# Join all four factors into a single data frame by county GEOID.
all_factors <- factor1_associations %>%
  left_join(factor2_turnout, by = "GEOID") %>%
  left_join(factor3_response, by = "GEOID") %>%
  left_join(factor4_nonprofits, by = "GEOID") %>%
  na.omit() # Remove counties with any missing data for the PCA

# -----
## Part 3: Perform Principal Component Analysis (PCA)
# -----
# 1. Select only the numeric factor columns for the PCA
pca_data <- all_factors %>% select(association_rate, turnout_rate, response_rate, nonprofit_rate)

# 2. Standardize the data (set mean to 0, standard deviation to 1)
scaled_data <- scale(pca_data)

# 3. Run the PCA
pca_result <- prcomp(scaled_data)

# 4. Extract the first principal component (PC1) as our index
all_factors$social_capital_index <- pca_result$x[, "PC1"]

# -----
## Part 4: View the Results
# -----
# Rank counties in Oregon by their social capital score.
social_capital_rankings <- all_factors %>%
  select(NAME, social_capital_index) %>%
  mutate(rank = rank(-social_capital_index)) %>%
  arrange(rank)

knitr::kable(
  social_capital_rankings,
  digits = 2,
  caption = "Oregon Social Capital Index Rankings by County",
  col.names = c("County", "Social Capital Index", "Rank")
)

```

Explanation of the PCA

1. `scale(pca_data)`: This is a crucial step that **standardizes** the four factors. It converts each value into a Z-score, telling us how many standard deviations it is from the state-wide mean.

This ensures that a factor with a large scale (like `turnout_rate` near 80) doesn't dominate a factor with a small scale (like `association_rate` near 5).

2. `prcomp(scaled_data)`: This is the function that runs the PCA. It analyzes the relationships between the four factors and creates new, uncorrelated variables called "principal components."
3. `pca_result$x[, "PC1"]`: We extract the first principal component (**PC1**). This component is the one that captures the **largest possible amount of shared information** across all four of our input factors. It is, in essence, the strongest underlying pattern in the data, which we are defining as our social capital index. A higher score means the county tended to score higher across all the input measures.

Adverse Childhood Experiences (ACEs) Score

This score measures a range of childhood traumas. High ACE scores in a population are a strong predictor of poor behavioral and physical health outcomes later in life. This is a foundational, causal measure for long-term health.

- **Primary Source: Behavioral Risk Factor Surveillance System (BRFSS)** from the CDC. Many states, including Oregon, use an ACEs module in their BRFSS surveys.
- **How to Access in R:** The CDC provides public-use data files. We can use the `survey` and `srvyr` packages in R to analyze this complex survey data correctly. We would look for Oregon's specific BRFSS data to see if county-level estimates are available.

```
# -----
# 1. Load your libraries and data
# -----
# pacman::p_load(tidyverse, srvyr) # Ensure srvyr is loaded
# Load your full BRFSS data file here
# aces_microdata <- read_csv("data/your_brfss_data_file.csv") # Example

# -----
# 2. Filter and prepare the data for analysis
# -----
# This code assumes you have the 'aces_microdata' object loaded.
multnomah_aces_data <- aces_microdata %>%
  # !!! ACTION: Replace 'county_fips' with the actual county variable name in your data
  filter(county_fips == 41051) %>%

  # Recode 'Don't know' (77) and 'Refused' (99) in ACETOTAL to NA
  mutate(acetotal_clean = if_else(ACETOTAL %in% c(77, 99), NA_real_, ACETOTAL)) %>%

  # Create the final ACEs categories
  mutate(
    ace_category = case_when(
      acetotal_clean == 0 ~ "0 ACEs",
      acetotal_clean == 1 ~ "1 ACE",
      acetotal_clean %in% c(2, 3) ~ "2-3 ACEs",
```

```

    acetotal_clean >= 4 ~ "4 or more ACEs",
    TRUE ~ NA_character_ # All other cases (including NA) become NA
  )
) %>%
# Remove rows where our category is NA, as they can't be analyzed
filter(!is.na(ace_category))

# -----
# 1. Create the survey design object
# -----
# This tells srvyr about the complex survey design (strata, clusters, and weights)
aces_survey_design <- multnomah_aces_data %>%
  as_survey_design(
    ids = `_PSU`,      # Primary Sampling Unit (cluster)
    strata = `_STSTR`, # Stratum
    weights = `_LLCPWT` # The final weight
  )

# -----
# 2. Calculate the weighted percentages and confidence intervals
# -----
dat_multnomah_aces <- aces_survey_design %>%
  # Group by our newly created ACEs categories
  group_by(ace_category) %>%
  # Calculate the weighted mean (proportion) and 95% CI for each group
  summarize(
    percentage = survey_mean(vartype = "ci", level = 0.95),
    n_unweighted = unweighted(n()) # Also get the unweighted sample size
  ) %>%
  # Reorder the factor for a clean table
  mutate(ace_category = fct_relevel(ace_category, "0 ACEs", "1 ACE", "2-3 ACEs")) %>%
  arrange(ace_category)

# -----
# 3. Display the final results table
# -----
knitr::kable(
  dat_multnomah_aces,
  digits = 1,
  caption = "Weighted Prevalence of ACEs in Multnomah County (2020 BRFSS)",
  col.names = c("ACEs Category", "Percent (%)", "Lower 95% CI", "Upper 95% CI", "Sample Size (n)")
)

```

Outcome-based Behavioral Health Measures

These measures reflect the results or consequences of behavioral health issues and the effectiveness of the system in place.

Fatal Overdose Rate (by substance, e.g., fentanyl)

This is a critical and highly-impactful measure for Multnomah County, where fentanyl-related deaths have seen a dramatic increase. It directly reflects the severity of the opioid crisis.

- **Primary Source: CDC WONDER (Wide-ranging Online Data for Epidemiologic Research).** This is a powerful tool for querying mortality data from death certificates. The **Oregon Health Authority (OHA)** also has public data dashboards that are often more current.
- **How to Access in R:** There is not an official R package for CDC WONDER, but it's possible to write scripts to query the web form and parse the results. For OHA, we would check their open data portal for an API or downloadable files.
- **Key Variables:** We would query by cause of death (using ICD-10 codes for overdose or suicide), county, year, and demographic characteristics.

```
## -----  
## (Assuming your raw, merged vital records object 'dat_vr' is loaded)  
## -----  
  
## Define the ICD-10 codes for drug poisonings (overdoses)  
overdose_codes <- c(  
  "X40", "X41", "X42", "X43", "X44", # Accidental  
  "X60", "X61", "X62", "X63", "X64", # Intentional (suicide)  
  "X85",                               # Assault  
  "Y10", "Y11", "Y12", "Y13", "Y14" # Undetermined intent  
)  
  
## -----  
## Filter data for all drug overdoses  
dat_overdose_all_raw <- dat_vr %>%  
  
  ## Filter to Multnomah County  
  filter(dplacecountyfips == "051") %>%  
  
  ## Filter for ICD-10 poisoning codes in the underlying cause of death  
  filter(str_trim(dmca_acme) %in% overdose_codes) %>%  
  
  ## Create deathdate to filter for the last five years  
  mutate(deathdate = lubridate::make_date(ddodyear, ddodmonth, ddodday))  
  
five_year_period <- max(dat_overdose_all_raw$deathdate) - years(5)  
  
dat_overdose_all <- dat_overdose_all_raw %>%  
  filter(deathdate >= five_year_period)  
  
cat("Data filtered for all fatal drug overdoses.\n")  
  
## -----
```

```
## Recode variables (The rest of your script follows here)
# This section for recoding sex, race, age, and cleaning addresses
# can be applied directly to the `dat_overdose_all` object.
# The result is a clean line-list of all overdose deaths.
## -----
```

Suicide Rates (by age group, gender, and race)

Suicide is a tragic and direct outcome of severe behavioral health distress. Analyzing the data by specific demographics can help identify the most vulnerable populations.

- **Primary Source: CDC WONDER (Wide-ranging Online Data for Epidemiologic Research).** This is a powerful tool for querying mortality data from death certificates. The **Oregon Health Authority (OHA)** also has public data dashboards that are often more current.
- **How to Access in R:** There is not an official R package for CDC WONDER, but it's possible to write scripts to query the web form and parse the results. For OHA, we would check their open data portal for an API or downloadable files.
- **Key Variables:** We would query by cause of death (using ICD-10 codes for overdose or suicide), county, year, and demographic characteristics.

```
## -----
## (Assuming your 'dat_vr_raw' or similar object is loaded and merged)
## Let's start from the combined raw data object before filtering.
# dat_vr <- bind_rows(death_recent, death_decade) |>
#   clean_names() |>
#   mutate(dyyyyysfn = as.numeric(dyyyyysfn))

# For this example, let's assume `dat_vr` is the raw, combined dataset.

## -----
## Define the ICD-10 codes for suicide
suicide_codes <- c(
  "X60", "X61", "X62", "X63", "X64", "X65", "X66", "X67", "X68", "X69",
  "X70", "X71", "X72", "X73", "X74", "X75", "X76", "X77", "X78", "X79",
  "X80", "X81", "X82", "X83", "X84", "Y870"
)

## -----
## filter data for suicides
dat_suicide_raw <- dat_vr %>%

  ## filter to Multnomah County
  filter(dplacecountyfips == "051") %>%

  ## !!! CHANGE: Filter for ICD-10 suicide codes in the underlying cause of death
  filter(str_trim(dmca_acme) %in% suicide_codes) %>%
```

```

## filter for last five years
mutate(deathdate = lubridate::make_date(ddodyear, ddodmonth, ddodday))

five_year_period <- max(dat_suicide_raw$deathdate) - years(5)

dat_suicide <- dat_suicide_raw %>%
  filter(deathdate >= five_year_period)

cat("Data filtered for suicide deaths.\n")

## -----
## recode variables (This section is identical to your original script)
## recode sex, race, age, dates, and select final variables...
# (The rest of your recoding and cleaning script follows here)
# For brevity, I'll assume the same recoding logic is applied to `dat_suicide`.
# The output will be a clean dataframe `dat_suicide` with the same structure
# as your final fentanyl dataset.
## -----

# Example of what the final cleaned data would look like
# (Assuming the full recoding pipeline is run on `dat_suicide`)
# kable(head(dat_suicide_final))

```

Emergency Department Visits for Behavioral Health Crises

This measure indicates the number of people in acute distress who are not being served by the regular, community-based mental health system. A high rate suggests a fragmented or insufficient system of care.

- **Primary Source: Oregon Health Authority (OHA).** They collect hospital discharge data, which includes ED visits.
- **How to Access in R:** This data is often sensitive and may not be available via a public API. We would start by exploring OHA’s public data portals. If not available, this would likely require a formal data request.

Part 1: Guidance for Querying ESSENCE

To get data on Emergency Department (ED) visits for behavioral health crises, you’ll want to build a query that captures visits for mental health conditions, substance use, overdoses, and suicide-related behaviors. Here’s a recommended approach:

1. Use Pre-built Syndromes First:

- Check if your jurisdiction’s ESSENCE has pre-built syndromes like “Mental Health,” “Substance Use,” or “Suicide and Self-Harm.” These are often the most reliable and have been validated locally.

2. Query the Chief Complaint (CC) and Discharge Diagnosis (DD) Fields:

- If you build a custom query, use a combination of keywords in the chief complaint and ICD-10 codes in the diagnosis fields.

3. Define Your Terms:

- **For Mental Health Crises:**
 - **Keywords:** anxiety, depression, psychosis, psychiatric, behavioral health, bipolar, schizophrenia, homicidal ideation
 - **ICD-10 Codes:** The F00-F99 block (“Mental, Behavioral and Neurodevelopmental disorders”) is your primary tool. You can select the whole block or specific ranges like F10-F19 (substance use disorders), F20-F29 (schizophrenia), F30-F39 (mood disorders), F40-F48 (anxiety disorders).
- **For Suicide and Self-Harm:**
 - **Keywords:** suicide, suicidal, self harm, cutting, SI (suicidal ideation)
 - **ICD-10 Codes:** T14.91 (suicide attempt), X60-X84 (intentional self-harm).
- **For Overdoses:**
 - **Keywords:** overdose, OD, ingestion
 - **ICD-10 Codes:** T36-T50 (poisoning by drugs).

4. Set Query Parameters:

- **Time Period:** Select your desired date range (e.g., the last five years).
- **Geography:** Filter for hospitals located in Multnomah County.
- **Export:** Run the query and export the results as a time-series (e.g., weekly counts). Be sure to **export it as a CSV file**.

Part 2: R Code for Analyzing the Exported Data

Once you have the CSV file from ESSENCE, this R code provides a template to load, clean, and visualize it. You may need to adjust the column names in the code to match your specific export.

```
# -----
# 1. Load the exported CSV data
# -----
# Place your exported file in your 'data/' folder and update the filename.
essence_raw <- read_csv("data/ESSENCE_Behavioral_Health_Export.csv")

# -----
# 2. Clean and prepare the data
# -----
dat_ed_visits <- essence_raw %>%
  # Standardize column names (e.g., "Data Date" becomes "data_date")
  janitor::clean_names() %>%
  # !!! ACTION: Rename the core columns to be generic: 'date' and 'visits'
  # Update 'data_date' and 'count' to whatever they are named in your file.
```

```

rename(
  date = data_date,
  visits = count
) %>%
# Convert the date column to a proper date format
mutate(date = mdy(date)) %>% # Use mdy() or ymd() depending on your date format
# Ensure visits are numeric
mutate(visits = as.numeric(visits)) %>%
# Arrange the data chronologically
arrange(date)

# -----
# 3. Create a time-series visualization
# -----
ggplot(dat_ed_visits, aes(x = date, y = visits)) +
  geom_line(color = "#326195", linewidth = 1) +
  geom_point(color = "#326195", size = 1.5, alpha = 0.5) +
  labs(
    title = "Weekly Emergency Department Visits for Behavioral Health Crises",
    subtitle = "Multnomah County",
    x = "Date",
    y = "Number of Visits"
  ) +
  theme_minimal() +
  scale_x_date(date_breaks = "1 year", date_labels = "%Y")

```

Mental Health-Related Civil Commitment Rates

This measures the number of individuals involuntarily placed in a treatment facility due to a mental health crisis. It can highlight gaps in preventative and community-based care.

- **Primary Source:** This data is highly localized and held by the **Multnomah County Courts** and local law enforcement agencies.
- **How to Access in R:** It's very unlikely this data is available via an API. The most probable path would be a public records request to get aggregated, de-identified data in a flat file (like a CSV).

Criminal Justice System Involvement (mental health-related arrests)

The number of arrests and incarcerations for individuals with mental health disorders indicates a failure to provide effective, community-based support.

- **Primary Source:** This data is highly localized and held by the **Multnomah County Courts** and local law enforcement agencies.
- **How to Access in R:** It's very unlikely this data is available via an API. The most probable path would be a public records request to get aggregated, de-identified data in a flat file (like a CSV).

Substance Use Disorder Treatment Admission and Completion Rates

This is a key measure of the system’s effectiveness. High admission rates show a need for services, while low completion rates can point to barriers in treatment.

- **Primary Source: Substance Abuse and Mental Health Services Administration (SAMHSA).** They manage the **Treatment Episode Data Set (TEDS)**.
- **How to Access in R:** SAMHSA provides public-use data files for TEDS admissions (TEDS-A) and discharges (TEDS-D) that can be downloaded and analyzed in R.

How to Find the Data

1. **Search the OHA Website:** Navigate to the OHA website and search for reports or data dashboards related to “substance use disorder treatment,” “behavioral health data,” or “TEDS report.”
2. **Locate County-Level Data:** Look for tables within these reports that show treatment admissions and discharges (or completions) broken down by county.
3. **Extract the Numbers:** Find the most recent annual counts for Multnomah County for both **total admissions** and **treatment completions**. If a downloadable data file (CSV/Excel) is available, that’s ideal. If not, you can manually pull the numbers from a table in a PDF report.

Example code:

```
# -----
# 1. Manually enter the SUD treatment data from the OHA report
# -----
# Replace the placeholder numbers in the 'count' column with the most
# recent data for Multnomah County from the OHA.
dat_sud_treatment <- tribble(
  ~metric,          ~count,
  "Total Admissions", 8500, # <-- Replace with actual number
  "Treatment Completed", 3900 # <-- Replace with actual number
  # Add other discharge reasons if available, e.g., "Transferred", "Dropped Out"
)

# -----
# 2. Calculate the completion rate
# -----
# Extract the individual values to perform the calculation
admissions <- dat_sud_treatment %>% filter(metric == "Total Admissions") %>% pull(count)
completions <- dat_sud_treatment %>% filter(metric == "Treatment Completed") %>% pull(count)

# Calculate the rate
completion_rate <- (completions / admissions) * 100

# -----
# 3. Display the final summary
```

```
# -----
# Add the calculated rate to our table for display
dat_sud_summary <- dat_sud_treatment %>%
  add_row(metric = "Completion Rate (%)", count = completion_rate)

knitr::kable(
  dat_sud_summary,
  digits = 1,
  caption = "Substance Use Disorder Treatment Summary for Multnomah County",
  col.names = c("Metric", "Value")
)
```

Process and Systems-based Measures

These measures evaluate the functioning of the behavioral health system itself, focusing on how services are delivered and used.

- **Wait Times, Patient Satisfaction, Retention Rates:** Your best bet for these would be to look for reports or publications from the **Oregon Health Authority** or **Multnomah County Health Department**. It's unlikely you'll find a raw, queryable dataset for these.
- **Integration of Behavioral and Physical Health Care:** This is more of a qualitative or composite measure. You might find metrics in OHA reports about the number of certified coordinated care organizations or patient-centered medical homes, but it won't be a simple number you can pull from an API.

Wait Times for Behavioral Health Appointments

Long wait times are a significant barrier to care. This measure directly reflects the accessibility and capacity of the behavioral health system.

Patient/Consumer Satisfaction with Behavioral Health Services

Surveying individuals who use the system can provide valuable qualitative and quantitative feedback on the quality, cultural responsiveness, and effectiveness of care.

Retention Rates in Behavioral Health Programs

This measure tracks how long people stay engaged in treatment. High retention rates suggest that programs are effective and that clients feel supported.

Integration of Behavioral and Physical Health Care

This measures the extent to which mental health, substance use, and physical health services are coordinated. A high level of integration leads to better overall health outcomes. It can be measured by looking at metrics like the number of individuals with co-occurring mental and physical health conditions who are seen by integrated care teams.