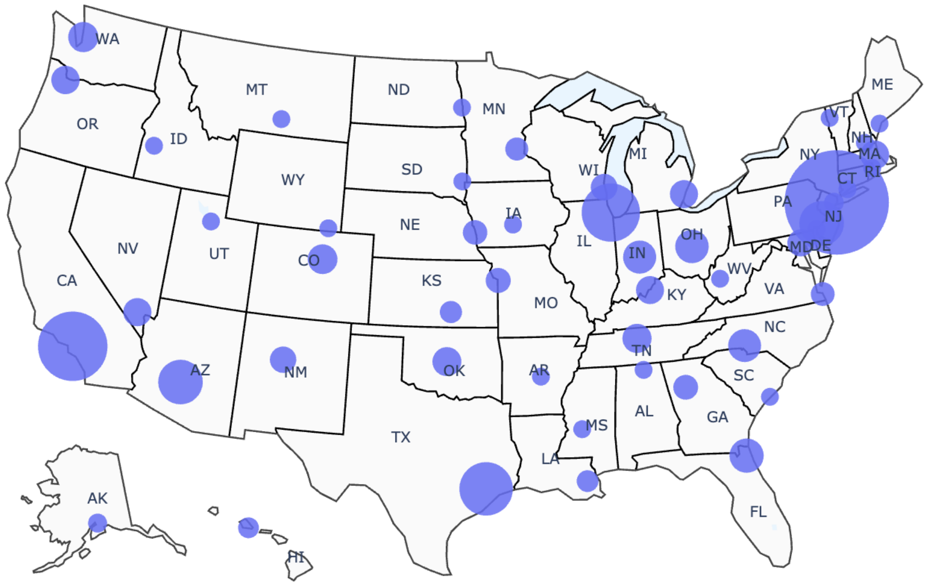
Precalc AI Lab

Power Functions to Predict City Populations



In this Lab you will write Python code and run it. You can write the code yourself or get an LLM to do it. Run the code in Google Colab. If it does not run correctly, rewrite until it works correctly.

Purpose of Lab

- Recognize the shape of power functions of the form $f(x) = a/x^b$ with positive constants a and b.
- Create models using functions of the form $f(x) = a/x^b$.

Context for Lab

List the cities of a US state by population from largest to smallest. Give the largest city rank $r = 1$, the next largest rank $r = 2$, and so on. A surprising pattern shows up for Nevada: the 2$^{\text{nd}}$-ranked city, Henderson, has about half the population of the largest city, Las Vegas. Also, the 3$^{\text{rd}}$-ranked city, Reno, has about one-third the population of Las Vegas. In symbols, we have

$$\text{Population of } r^{\text{th}} \text{ largest city in Nevada} = P(r) \approx \frac{641{,}903}{r}$$

This is an example of Zipf's law, which allows you to estimate the population of a city in a state by using just its rank. More generally, Zipf's law says that there for a given region, like a state, there are constants A and b such that

$$\text{Population of } r^{\text{th}} \text{ largest city in the region} = P(r) \approx \frac{A}{r^b}$$

The constants A and the b depend on the region, and b is often close to 1.

Part 1: Creating a model for Oregon

Let's consider Oregon and ask an LLM to give us a model using Zipf's Law.

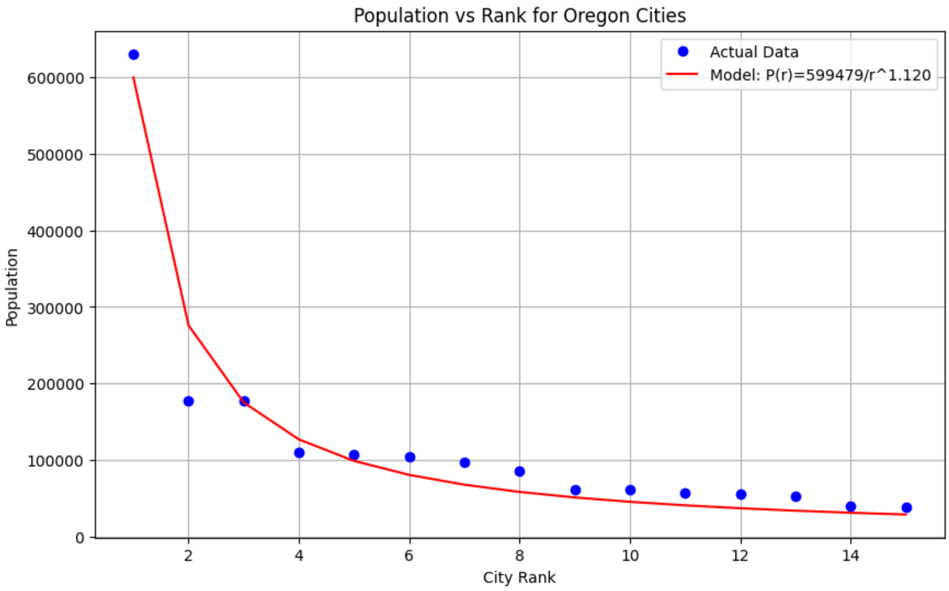
Use an LLM

- Find the population of the 15 largest cities in Oregon.
- Generate python code to find a power-law model of the form $P(r) = A/r^b$, where r is the rank of the city, and plot the function $P(r)$ together with the data. We ask the LLM

- for code that is simple, easy to follow for a Precalculus student, and runs in Colab.
- to give the values of A and b.

Note: If the LLM gives a log-log plot, ask for a regular plot. If it shows A or b in scientific notation (e.g., $1.3 \cdot 10^6$), you can request standard notation if you prefer.

Here is the plot you should get if you use the population of the 15 largest cities in Oregon to be 630498, 177899, 177432, 110685, 107730, 104557, 96945, 85098, 61087, 61085, 57053, 55590, 53362, 39924, 39149. If the LLM uses different data, the model will be slightly different.



This is the formula you should get for Oregon:

$$\text{Population of } r^{\text{th}} \text{ largest city in Oregon} = P(r) \approx \frac{599{,}479}{r^{1.12}}$$

Part 2: What the model tells us

Consider the model for Oregon you found with the LLM. Answer the following questions:

1. What is the value of A? What practical meaning does it have? [Hint: Plug in $r = 1$, what do you get?]
2. What is the value of b?
3. How good is the model at predicting the population of the 5^{th} largest city in Oregon? Of the 10^{th} largest city?
4. If you were to try to find a model for California, what practical meaning would A have? Would it be larger or smaller than the A you found for Oregon? Why?

Part 3: Model for another state

Use an LLM

Do the same thing you did in part 2 but for a different state. That is:

- Plot the populations of the 15 largest cities of a different state by rank.
- Ask the LLM to give a power-law model of the form $P(r) = A/r^b$, where r is the rank of the city. Give the values of A and b.
- Plot the model together with the points. [Note: if the LLM gives you a log-log plot, ask it to generate a regular plot for you]

Answer the following questions:

1. How are A and b different from the Oregon ones? How do you think it affects the shape of the curve? (Plot the two models on the same set of axes)
2. How good is your model at predicting the population of the 5^{th} largest city in this other state? Of the 10^{th} largest city?

Further discussion

What is the shape of the graph of power functions of the form $f(x) = A/x^b$ for positive constants A and b? Answer the following questions to figure it out. Feel free to ask an LLM to help you out with the plots!

1. What is the long-run behavior as x grows?
2. How does changing the value of A affect the shape of the curve? [Try fixing the value of b at 1.1 and plotting the function for several values of A]
3. How does changing the value of b affect the shape of the curve? [Try fixing the value of A at 10 and plotting the function for several values of b – look at the steepness of the drop-off].

Sample Python Code

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit

# Part 1: Oregon data
ranks_oregon = np.arange(1, 16)
populations_oregon = np.array([630498, 177899, 177432, 110685, 107730, 104557, 96945,85098, 61087, 61085,
      57053, 55590, 53362, 39924, 39149])


# Power law function
def power_law(r, A, b):
   return A / (r ** b)


# Fit the model
params, _ = curve_fit(power_law, ranks_oregon, populations_oregon)
A_oregon, b_oregon = params


# Generate model predictions
fitted_oregon = power_law(ranks_oregon, A_oregon, b_oregon)


# Plot the data and model
plt.figure(figsize=(10, 6))
plt.plot(ranks_oregon, populations_oregon, 'bo', label='Actual Data')
plt.plot(ranks_oregon, fitted_oregon, 'r-', label=f'Model: P(r)={A_oregon:.0f}/r^{b_oregon:.3f}')
plt.title('Population vs Rank for Oregon Cities')
plt.xlabel('City Rank')
plt.ylabel('Population')
plt.legend()
plt.grid(True)
plt.show()

# print the values of A and b
A_oregon, b_oregon
```