Abstract: This project models U.S. population growth with three simple differential-equation models: exponential, logistic, and Gompertz. Let P(t) be population. I estimate each model's parameters from historical U.S. data and compare the fits. In all cases, the growth rate r has units of 1/year (per year). The logistic model has a carrying capacity K, and the Gompertz model has an upper level L; both K and L are measured in people.

Results show the exponential model can match short-run growth but tends to predict too high in the long run, because it never slows down. The logistic and Gompertz models both slow as population rises. Logistic is symmetric around K/2, while Gompertz slows earlier and often fits long spans of U.S. data better. However, estimates of K or L depend a lot on which years are used, since the series does not clearly reach a stable plateau. Error checks over time show patterns linked to real-world forces that these simple models ignore: immigration and emigration, and life-expectancy rates, or any wide-spread disasters or wars. These issues break the constant rate of growth we look for in the US as well as and closed-population assumptions. Overall, these models are useful to summarize trend and make short-term projections, with Gompertz and Logistical often being the most accurate curves.

Introduction

Let $P(t)$ denote the population at time t (people). The total growth rate is

$$\frac{dP}{dt} \quad \text{(people per unit time).}$$

To compare models on the same footing, define the *per-capita (per-unit) growth rate*

$$g(P) \;=\; \frac{1}{P}\frac{dP}{dt} \quad \text{(per unit time).}$$

Throughout, r is a rate with units 1/time (e.g., per year if t is in years). The parameters K and L are long-run size scales (people). All three models below assume a closed population and constant parameters over the fitted period.

Exponential model.

$$\frac{dP}{dt} = rP \qquad \Longrightarrow \qquad g(P) = r \text{ (constant).}$$

Interpretation: each person contributes the same expected increment to growth regardless of current size, so there is no built-in slowdown. This is useful over short spans but over-predicts long-run levels.

Logistic model.

$$\frac{dP}{dt} = r\Big(1 - \frac{P}{K}\Big)P \qquad \Longrightarrow \qquad g(P) = r\Big(1 - \frac{P}{K}\Big) \text{ (linear, decreasing in } P).$$

Here K is the carrying capacity. Total growth $\frac{dP}{dt}$ is S-shaped: it peaks at $P = K/2$ with height $rK/4$ and is (approximately) symmetric around $K/2$. Per-capita growth falls to zero as $P \to K$.

Gompertz model.

$$\frac{dP}{dt} = r \ln\!\left(\frac{L}{P}\right) P \qquad \Longrightarrow \qquad g(P) = r \ln\!\left(\frac{L}{P}\right) \text{ (logarithmic decline in } P).$$

Here L is an upper level. The total growth curve is an asymmetric S: it bends earlier than logistic, peaks at $P = L/e$ with height rL/e (where e is Euler's number), and then slows smoothly as P approaches L.

Summary. Exponential keeps $g(P)$ constant (no slowdown). Logistic makes $g(P)$ fall linearly to zero at $P = K$. Gompertz makes $g(P)$ decline more gently and earlier via $\ln(L/P)$. These differences in $g(P)$ explain why exponential fits short-run growth, while logistic and Gompertz better capture long-run deceleration.

Methods. To find my parameters L (Carrying capacity) and r (growth rate) I conducted 2 separate tests. For L, I thought of other countries that may be reaching their carrying capacity, China came to mind. The good thing about China as well, is they have technological advancements similar to the United States and we both have similar economies (Top 2 GDP from https://www.worldometers.info/gdp/gdp-by-country/). So what I did to approximate the United States carrying capacity, is I built a ratio with China's carrying capacity, to its geographical size, and I solved it comparing the United States geographical size.

$$\frac{\text{China Carrying capacity}}{\text{China Size}} = \frac{\text{US Carrying capacity}}{\text{US Size}}$$

From this, I found the United States carrying capacity to be about 1.543 billion people. So this will serve as our L. Now we must find our r value. What I did to find out the best r is test a load of r's and using the sklearn.metrics r2_score function. I calculated the R^2 value for each of these different r's in each of the different models. I managed to find the best r value for each model uusing our given initial condition $y_0 = 3,929,214$ Below is my table I generated from my code with the r's and R^2's.

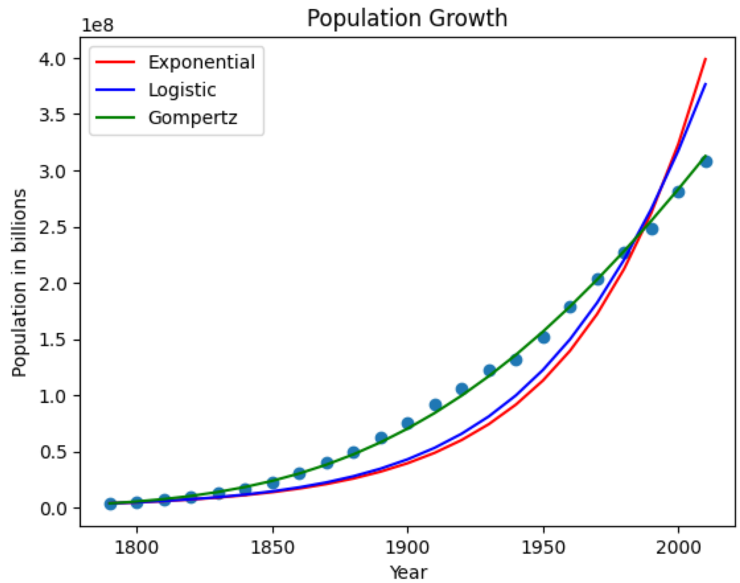
Optimizing r:

```python
r_values = [0.001, 0.002, 0.003, 0.004, 0.005,
print("r value  Exponential   Logistic    Gompe
for i in r_values:
  r = i
  Gomp_sol = odeint(gompertz_model, y0, t)
  Expo_sol = odeint(expo_model, y0, t)
  Log_sol = odeint(logistic_model, y0, t)
  R_expo = r2_score(pop_data, Expo_sol)
  R_log = r2_score(pop_data, Log_sol)
  R_Gomp = r2_score(pop_data, Gomp_sol)
  print(f"{r}     {R_expo:.4f}      {R_log:.4f}
```

r value	Exponential	Logistic	Gompertz
0.001	-1.1141	-1.1141	-0.9920
0.002	-1.0944	-1.0945	-0.7213
0.003	-1.0708	-1.0711	-0.2904
0.004	-1.0427	-1.0431	0.2567
0.005	-1.0090	-1.0098	0.7689
0.006	-0.9689	-0.9701	0.9985
0.007	-0.9209	-0.9229	0.6578
0.008	-0.8638	-0.8668	-0.5070
0.009	-0.7959	-0.8004	-2.6572
0.01	-0.7154	-0.7222	-5.8390
0.018	0.5317	0.4467	-52.8167
0.019	0.7178	0.6285	-59.1526
0.02	0.8514	0.7851	-65.2801
0.021	0.8738	0.8917	-71.1659
0.022	0.6890	0.9158	-76.7906
0.023	0.1423	0.8174	-82.1454

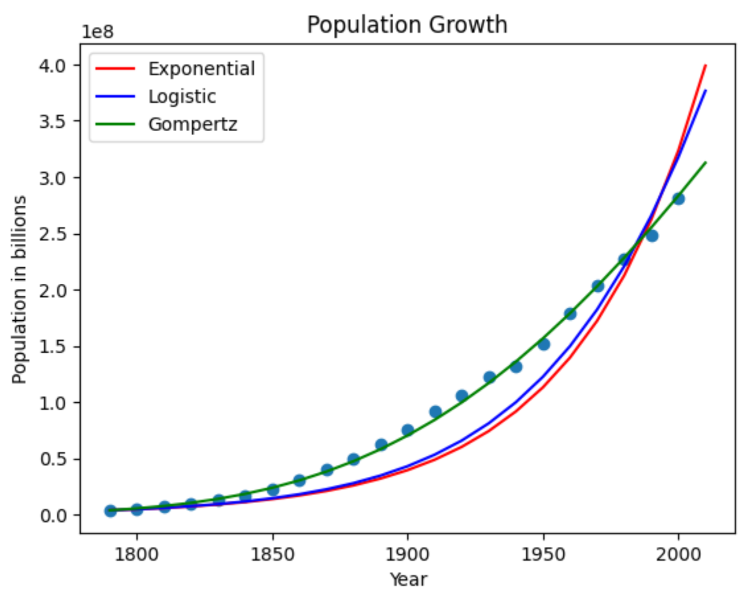
So, for my exponential model, the best r is 0.021, since that is the closest the R^2 gets to 1. For the logistic model, my best r is 0.022 with an R^2 of 0.9158. Finally, the best r value happens to be for the Gompertz model, with an r value of 0.006 we get an R^2 of 0.9985, which is really close to 1. So that means our model nearly perfectly fits for the values of the data. Because our model fits so well, we could be able to possibly predict the population in future years.

Results and Conclusions:

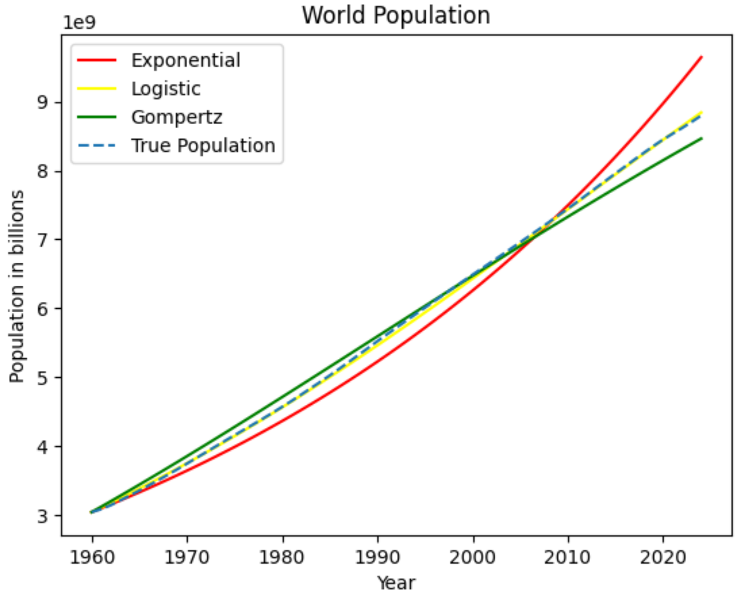


Model including all 3 models with best fitted parameters.

Now, to test how well the models work, I am going to remove the last point from t (2010) and am going to have the models try to predict what the population is during 2010. I noted the predictions and the % errors in the output from my Python code and graph below.



Same US Population dataset with the 2010 point removed.



Now here the same 3 models, but now fitted to the World Population data.

For the world population data, I did the same analysis to find r, however, L was difficult to calculate at this high of a scale, where my calculate L was inaccurate, so I gave us the value of $L = 15,000,000,000$.

Fit		US population (millons)			Reduced Model Predictions	
Model	**P_0**	**L**	**r**	**2010 Data**	**Prediction**	**Error%**
Exponential	3929214		0.021	308,745,538	496,925,188.54	60.95%
Logistic	3929214	1,541,000,000	0.022	308,745,538	317,449,248.77	2.82%
Gompertz	3929214	1,541,000,000	0.006	308,745,538	312,599,331.35	1.25%
Fit		World population (billions)		**2023 Data**		
Exponential	3,046,620,397	15,000,000,000	0.018	8,795,003,197	17,150,596,400	95.00%
Logistic	3,046,620,397	15,000,000,000	0.027	8,795,003,197	8,839,305,400	0.50%
Gompertz	3,046,620,397	15,000,000,000	0.016	8,795,003,197	11,300,986,400	28.49%

Table with all of our found parameters for the US Population and World Population

For the United States in 2010 (target 308,745,538), the Gompertz model was best with 1.25% error. The Logistic model was next at 2.82%. The Exponential model overpredicted badly at 60.95%.

For the World in 2023 (target 8,795,003,197), the Logistic model was best with 0.50% error. The Gompertz model overpredicted at 28.49%, and the Exponential model missed by 95%.

Exponential keeps the per-capita growth rate constant, so it tends to overshoot. Logistic and Gompertz both slow as population rises. Gompertz bends earlier, which matches recent U.S. slowing; Logistic matches the world series around 2023 a bit better.

	R^2 Value			Prediction Error		
Data	Exponential	Logistic	Gompertz	Exponential	Logistic	Gompertz
USA	0.8738	0.9158	0.9985	60.95%	2.82%	1.25%
World	0.9744	0.9996	0.9932	95%	0.50%	28.49%

R^2 and Prediction Error %

This table gives us a good idea on what models worked best, from this we can look at both our R^2 and Prediction error %. From this we can see that the logistic model was pretty good at predicting the next point, however, the Gompertz model manges to have a consistently higher R^2 value than the other models. So from this, we can clearly see that for population growth, exponential is not a good match, but depending on the population size, either logistical or Gompertz are good models for population growth.

When these models are fit on many decades of data, the estimated parameters r (and K, L) lock in the old trend, so the model carries a lot of "memory" from the distant past. If recent birth rate, death rate, or migration have shifted, the fixed curve cannot adapt quickly and the forecasts keep following the outdated S-shape. In this sense the model remembers too much of the past to predict the near future well.

Although these models are good for short-term prediction and summarizing trends, these models ignore some pretty important factors when it comes to the population. Immigration and emigration, and life-expectancy rates, or any wide-spread disasters or wars are all important factors that can increase or decrease the population, and our model completely ignores these. These issues break the constant rate of growth we look for in the US as well as and closed-population assumptions. Overall, these models are useful to summarize trend and make short-term projections, with Gompertz and Logistical often being the most accurate curves.

Bibliography

Worldometers. Worldometer – Real Time World Statistics. 2025, https://www.worldometers.info/ (GDP and Area data)

OpenAI. ChatGPT. 2025, https://chat.openai.com (some Python code and packages)