Untitled

June 5, 2025

```
[1]: """

Project_LPI_World_Bank
___Linear Interpolation

"""

import numpy as np
import pandas as pd
from scipy import interpolate

# Singapore population data
years = np.array([2007, 2010, 2012, 2014, 2016, 2018, 2023])
population = np.array([4588599, 5076732, 5312437, 5469724, 5607283, 5638676,
    5917648])

# Create complete year range for interpolation
complete_years = np.arange(2007, 2024)

# Cubic Spline Interpolation
cubic_interp = interpolate.CubicSpline(years, population)
cubic_predictions = cubic_interp(complete_years)

print("=== SINGAPORE POPULATION CUBIC SPLINE ESTIMATES ===")
print("Copy this data directly into Excel:")
print("\nYear\tPopulation (Cubic Spline)")
print("-" * 35)

# Create Excel-ready data
excel_data = []
for i, year in enumerate(complete_years):
    pop_estimate = int(round(cubic_predictions[i]))
    excel_data.append([year, pop_estimate])

    # Mark actual vs estimated
    if year in years:
        actual_pop = population[np.where(years == year)[0][0]]
```

```python
        marker = f" (ACTUAL: {actual_pop})"
    else:
        marker = " (ESTIMATED)"

    print(f"{year}\t{pop_estimate:,}{marker}")

print("\n" + "="*60)
print("EXCEL IMPLEMENTATION GUIDE")
print("="*60)

print("\n1. COPY THE DATA ABOVE INTO EXCEL:")
print("    - Column A: Years (2007-2023)")
print("    - Column B: Population estimates")

print("\n2. CAGR FORMULA FOR COUNTRY MATCHING:")
print("    Place this in your CAGR column:")

# Create the CAGR formula for Excel
cagr_formula = """=IF(OR(ISERROR(VLOOKUP(Y3,'2023'!$Y$3:$Z$141,2,FALSE)),
        ISERROR(VLOOKUP(Y3,'2007'!$A$4:$U$153,21,FALSE))),
    "Data Missing",
    POWER(VLOOKUP(Y3,'2023'!$Y$3:$Z$141,2,FALSE)/
        VLOOKUP(Y3,'2007'!$A$4:$U$153,21,FALSE), 1/16) - 1)"""

print(cagr_formula)

print("\n3. ALTERNATIVE CAGR FORMULA (if you have helper columns):")
alt_formula = """=IF(OR(Z3="", [2007_Population_Column]=""), "No Data",
    POWER(Z3/[2007_Population_Column], 1/16) - 1)"""
print(alt_formula)

print("\n4. FORMAT THE CAGR COLUMN:")
print("    - Select CAGR column")
print("    - Format as Percentage with 2 decimal places")

print("\n" + "="*60)
print("CUBIC SPLINE DATA FOR SINGAPORE (Excel Format)")
print("="*60)

# Create a clean DataFrame for export
df_export = pd.DataFrame({
    'Year': complete_years,
    'Singapore_Population': [int(round(x)) for x in cubic_predictions],
    'Data_Type': ['Actual' if year in years else 'Estimated' for year in
  complete_years]
})
```

```
print(df_export.to_string(index=False))

print("\n" + "="*60)
print("MISSING YEARS FILLED (Ready for Excel)")
print("="*60)

missing_years_df = df_export[~df_export['Year'].isin(years)]
print("These are the NEW estimates you can add to your data:")
print(missing_years_df[['Year', 'Singapore_Population']].to_string(index=False))

print("\n" + "="*60)
print("VERIFICATION - SINGAPORE CAGR CALCULATION")
print("="*60)

singapore_2007 = 4588599
singapore_2023 = 5917648
singapore_cagr = ((singapore_2023 / singapore_2007) ** (1/16) - 1) * 100

print(f"Singapore 2007 Population: {singapore_2007:,}")
print(f"Singapore 2023 Population: {singapore_2023:,}")
print(f"Singapore CAGR (2007-2023): {singapore_cagr:.2f}%")

print("\n" + "="*60)
print("STEP-BY-STEP EXCEL IMPLEMENTATION")
print("="*60)

print("""
STEP 1: Create Singapore Complete Dataset
1. Open a new Excel sheet called 'Singapore_Complete'
2. Copy the Year and Population data above into columns A and B
3. This gives you ALL years from 2007-2023 with no gaps

STEP 2: Add CAGR Column to Your Main Sheet
1. Go to your main sheet with country analysis
2. Add a new column header: "CAGR (2007-2023)"
3. In the first data row, paste this formula:

=IF(OR(ISERROR(VLOOKUP(Y3,'2023'!$Y$3:
 ↪$Z$141,2,FALSE)),ISERROR(VLOOKUP(Y3,'2007'!$A$4:$U$153,21,FALSE))),"No␣
 ↪Data",POWER(VLOOKUP(Y3,'2023'!$Y$3:$Z$141,2,FALSE)/VLOOKUP(Y3,'2007'!$A$4:
 ↪$U$153,21,FALSE),1/16)-1)

STEP 3: Copy Formula Down
1. Copy the CAGR formula down to all rows
2. Format the column as Percentage with 2 decimal places

STEP 4: Verify with Singapore
```

3

```
1. Find Singapore in your list
2. The CAGR should show approximately 1.64%
3. If it matches, your formula is working correctly!

STEP 5: Sort and Analyze
1. Sort by CAGR column to see fastest/slowest growing countries
2. Use conditional formatting to highlight high/low growth rates
""")

print("\nFormula Breakdown:")
print("- VLOOKUP(Y3,'2023'!$Y$3:$Z$141,2,FALSE) = Gets 2023 population")
print("- VLOOKUP(Y3,'2007'!$A$4:$U$153,21,FALSE) = Gets 2007 population")
print("- POWER(2023_pop/2007_pop, 1/16) - 1 = Calculates CAGR")
print("- IF statements handle missing data gracefully")
```

=== SINGAPORE POPULATION CUBIC SPLINE ESTIMATES ===
Copy this data directly into Excel:

```
Year    Population (Cubic Spline)
---------------------------------
2007    4,588,599 (ACTUAL: 4588599)
2008    4,764,090 (ESTIMATED)
2009    4,928,092 (ESTIMATED)
2010    5,076,732 (ACTUAL: 5076732)
2011    5,206,138 (ESTIMATED)
2012    5,312,437 (ACTUAL: 5312437)
2013    5,395,444 (ESTIMATED)
2014    5,469,724 (ACTUAL: 5469724)
2015    5,546,300 (ESTIMATED)
2016    5,607,283 (ACTUAL: 5607283)
2017    5,634,640 (ESTIMATED)
2018    5,638,676 (ACTUAL: 5638676)
2019    5,636,779 (ESTIMATED)
2020    5,646,339 (ESTIMATED)
2021    5,684,745 (ESTIMATED)
2022    5,769,385 (ESTIMATED)
2023    5,917,648 (ACTUAL: 5917648)
```

==
EXCEL IMPLEMENTATION GUIDE
==

1. COPY THE DATA ABOVE INTO EXCEL:
    - Column A: Years (2007-2023)
    - Column B: Population estimates

2. CAGR FORMULA FOR COUNTRY MATCHING:

```
   Place this in your CAGR column:
=IF(OR(ISERROR(VLOOKUP(Y3,'2023'!$Y$3:$Z$141,2,FALSE)),
        ISERROR(VLOOKUP(Y3,'2007'!$A$4:$U$153,21,FALSE))),
    "Data Missing",
    POWER(VLOOKUP(Y3,'2023'!$Y$3:$Z$141,2,FALSE)/
        VLOOKUP(Y3,'2007'!$A$4:$U$153,21,FALSE), 1/16) - 1)

3. ALTERNATIVE CAGR FORMULA (if you have helper columns):
=IF(OR(Z3="", [2007_Population_Column]=""), "No Data",
    POWER(Z3/[2007_Population_Column], 1/16) - 1)

4. FORMAT THE CAGR COLUMN:
   - Select CAGR column
   - Format as Percentage with 2 decimal places
```

```
==============================================================
CUBIC SPLINE DATA FOR SINGAPORE (Excel Format)
==============================================================
```

Year	Singapore_Population	Data_Type
2007	4588599	Actual
2008	4764090	Estimated
2009	4928092	Estimated
2010	5076732	Actual
2011	5206138	Estimated
2012	5312437	Actual
2013	5395444	Estimated
2014	5469724	Actual
2015	5546300	Estimated
2016	5607283	Actual
2017	5634640	Estimated
2018	5638676	Actual
2019	5636779	Estimated
2020	5646339	Estimated
2021	5684745	Estimated
2022	5769385	Estimated
2023	5917648	Actual

```
==============================================================
MISSING YEARS FILLED (Ready for Excel)
==============================================================
These are the NEW estimates you can add to your data:
```

Year	Singapore_Population
2008	4764090
2009	4928092
2011	5206138
2013	5395444
2015	5546300
2017	5634640

```
2019              5636779
2020              5646339
2021              5684745
2022              5769385
```

```
============================================================
VERIFICATION - SINGAPORE CAGR CALCULATION
============================================================
Singapore 2007 Population: 4,588,599
Singapore 2023 Population: 5,917,648
Singapore CAGR (2007-2023): 1.60%
```

```
============================================================
STEP-BY-STEP EXCEL IMPLEMENTATION
============================================================
```

STEP 1: Create Singapore Complete Dataset
1. Open a new Excel sheet called 'Singapore_Complete'
2. Copy the Year and Population data above into columns A and B
3. This gives you ALL years from 2007-2023 with no gaps

STEP 2: Add CAGR Column to Your Main Sheet
1. Go to your main sheet with country analysis
2. Add a new column header: "CAGR (2007-2023)"
3. In the first data row, paste this formula:

=IF(OR(ISERROR(VLOOKUP(Y3,'2023'!$Y$3:$Z$141,2,FALSE)),ISERROR(VLOOKUP(Y3,'2007'
!$A$4:$U$153,21,FALSE))),"No Data",POWER(VLOOKUP(Y3,'2023'!$Y$3:$Z$141,2,FALSE)/
VLOOKUP(Y3,'2007'!$A$4:$U$153,21,FALSE),1/16)-1)

STEP 3: Copy Formula Down
1. Copy the CAGR formula down to all rows
2. Format the column as Percentage with 2 decimal places

STEP 4: Verify with Singapore
1. Find Singapore in your list
2. The CAGR should show approximately 1.64%
3. If it matches, your formula is working correctly!

STEP 5: Sort and Analyze
1. Sort by CAGR column to see fastest/slowest growing countries
2. Use conditional formatting to highlight high/low growth rates

Formula Breakdown:
- VLOOKUP(Y3,'2023'!$Y$3:$Z$141,2,FALSE) = Gets 2023 population
- VLOOKUP(Y3,'2007'!$A$4:$U$153,21,FALSE) = Gets 2007 population
- POWER(2023_pop/2007_pop, 1/16) - 1 = Calculates CAGR

- IF statements handle missing data gracefully

```
----------------------------------------------------------------
CSV FORMAT (Copy to .csv file for Excel import):
----------------------------------------------------------------
Year,Singapore_Population,Data_Type
2007,4588599,Actual
2008,4764090,Estimated
2009,4928092,Estimated
2010,5076732,Actual
2011,5206138,Estimated
2012,5312437,Actual
2013,5395444,Estimated
2014,5469724,Actual
2015,5546300,Estimated
2016,5607283,Actual
2017,5634640,Estimated
2018,5638676,Actual
2019,5636779,Estimated
2020,5646339,Estimated
2021,5684745,Estimated
2022,5769385,Estimated
2023,5917648,Actual
```

[5]:
```python
"""

Project_LPI_World_Bank
___ CAGR Quintiles

"""

import pandas as pd

# Example CAGR data
df = pd.read_xlsx('LPI_Worldbank')


# Create quintile categories (labels 1 to 5)
df['Quintile'] = pd.qcut(df['CAGR'], 5, labels=False) + 1

# If you want custom labels (e.g., 'Q1', 'Q2', ...)
df['Quintile_Label'] = pd.qcut(df['CAGR'], 5, labels=['Q1', 'Q2', 'Q3', 'Q4',
  'Q5'])
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
```

```
Cell In[5], line 11
      8 import pandas as pd
     10 # Example CAGR data
---> 11 df = pd.read_xlsx('LPI_Worldbank')
     14 # Create quintile categories (labels 1 to 5)
     15 df['Quintile'] = pd.qcut(df['CAGR'], 5, labels=False) + 1

AttributeError: module 'pandas' has no attribute 'read_xlsx'
```

```python
"""

Project_LPI_World_Bank
___  CAGR Quintiles

"""

import pandas as pd

# Example CAGR data
df = pd.read_xlsx('LPI_Worldbank', encoding='latin-1', errors='ignore')
# If you have international characters
df = pd.read_csv('LPI_Worldbank', encoding='cp1252')

# Create quintile categories (labels 1 to 5)
df['Quintile'] = pd.qcut(df['CAGR'], 5, labels=False) + 1

# If you want custom labels (e.g., 'Q1', 'Q2', ...)
df['Quintile_Label'] = pd.qcut(df['CAGR'], 5, labels=['Q1', 'Q2', 'Q3', 'Q4',
  'Q5'])
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[7], line 11
      8 import pandas as pd
     10 # Example CAGR data
---> 11 df = pd.read_xlsx('LPI_Worldbank', encoding='latin-1', errors='ignore')
     12 # If you have international characters
     13 df = pd.read_csv('LPI_Worldbank', encoding='cp1252')

AttributeError: module 'pandas' has no attribute 'read_xlsx'
```

[ ]: