

Actividad 2 - Explorando nuestro mundo (segunda parte)

ExactasPrograma - Datos

Seguimos explorando nuestro mundo a través de datos recopilados por Gapminder. Nos vamos a centrar en la distribución de la riqueza entre países.

Seguiremos usando los módulos `pandas` y `seaborn`, recuerden importarlos en el entorno de `Python`:

```
import pandas as pd
import seaborn as sns
```

1. Curva de Lorenz

Vamos a suponer que en nuestro mundo hay un individuo por cada uno de los países representados en el dataframe, cuya riqueza es el pbi per cápita del país que representa. De esta forma, la riqueza total del mundo en un dado año es la suma del PBI per cápita de todos los países en ese año. La curva de Lorenz es una herramienta que permite visualizar cuán cerca (o lejos) está una dada distribución respecto de la igualdad.

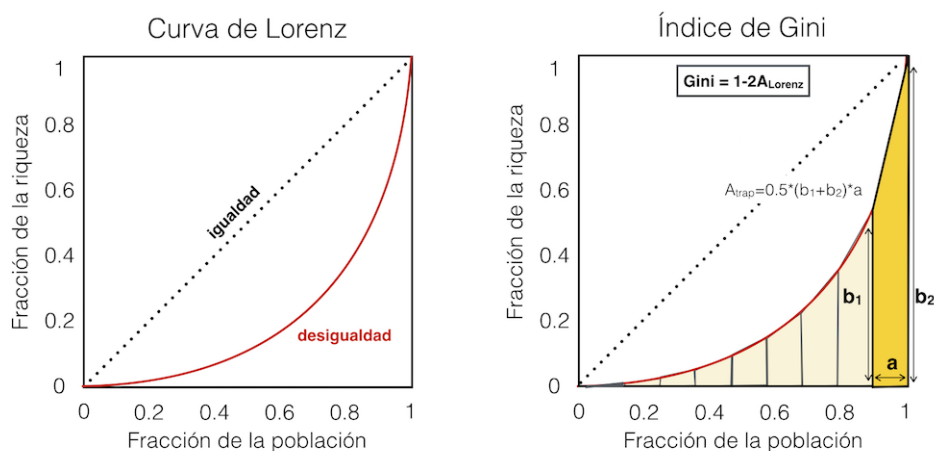


Figura 1: La **curva de Lorenz** muestra qué fracción de riqueza posee cada fracción de la población (ordenada de menor a mayor riqueza). La línea punteada corresponde a una distribución igualitaria y la curva roja a una distribución con mayor desigualdad.

Sigamos los siguientes pasos para construir la curva de Lorenz de distribución de PBI para el año 2000:

- Primero ordenar los países, según el PBI, de menor a mayor para ese año.
- Graficar la fracción acumulada de los individuos en el eje x, y la fracción de la riqueza acumulada en el eje y.

Si fuera una distribución igualitaria, esperaríamos ver que, por ejemplo, el 20 % de la población posee el 20 % de la riqueza total, el 50 % de la población, el 50 % de la riqueza, y así sucesivamente, obteniendo la línea de 45 grados (Figura 1).

Cuanto más desigual sea la distribución, esta curva se vuelve más panzona hacia abajo y la derecha, y podría incluir, por ejemplo el punto $(x = 0,8, y = 0,2)$, indicando que el 80 % de la población (ordenados de menor a mayor PBI per cápita) solamente posee el 20 % de la riqueza total.

1. Implementar una función `graficar_curva_lorenz(pbi, anio)` que recibe un dataframe con datos de riqueza por país y un año determinado, y genera la curva de Lorenz, con el agregado de la línea de referencia $y = x$.

2. Índice Gini

El índice de desigualdad de Gini intenta justamente resumir la esencia de la curva de Lorenz en un sólo número (entre 0 y 1), calculado de la siguiente manera:

- Partimos de los mismos datos de fracción de población y fracción de riqueza que graficamos en la curva de Lorenz.
 - Calculamos el área A bajo la curva de Lorenz, sumando las áreas de los trapecoides (Figura 1)
 - El índice de desigualdad de Gini se define como $B/(A+B) = 1-2A$, siendo B el área entre la recta de identidad y la curva de Lorenz y A el área bajo la curva de Lorenz, que da lugar a un índice entre 0 (igualdad) y 1 (máxima desigualdad). Es decir, cuanto más mayor sea el índice, mayor es la desigualdad.
2. Implementar una función `calcular_gini(df, anio)` que recibe un dataframe con datos de riqueza por país, un año, y calcula el índice de desigualdad de Gini entre países para ese año. Calcule el índice para el año 2000.
 3. Graficar la evolución del índice de Gini en los últimos 200 años. ¿Cuál es el momento en el que la desigualdad global fue máxima?
 4. Realizar un gráfico de líneas con el índice de Gini del mundo desde 1800 hasta 2014.
 5. Calcular y graficar en un mismo gráfico las curvas de evolución del índice de Gini correspondientes a cada uno de los continentes. Cuál es el continente que al día de hoy es más igualitario? Cuál el que tuvo el máximo histórico de desigualdad?
- Ayuda:** quizás sea útil que construya un nuevo *dataframe* `mundo` con información sintetizada para los años de interés.
6. Graficar la evolución de la riqueza promedio en los últimos 200 años. Distinguir las curvas de riqueza promedio para cada uno de los continentes.

3. Quiero más: adicionales

7. (*Optativo*) Para poder ver cómo ha cambiado esta desigualdad en los últimos 200 años, podríamos graficar sucesivamente para cada año una curva de Lorenz. Usando la función `graficar_curva_lorenz(pbi, anio)` puede hacer una animación de la

evolución en el tiempo de la curva de Lorenz para la desigualdad de riqueza entre países entre los años 1800 y 2020, visualizando por ejemplo las curvas cada 10 años.

Ayuda: Para cada año, puede generar una figura con los datos de ese año en particular y guardarlo en un archivo tipo .png, por ejemplo "Lorenz_1900.png" usando la función `savefig()` del paquete `savefig`. Luego, usando el paquete `imageio` (usado en el desafío de avalancha de Exactas Programa) pueden armar la animación "Lorenz_evolucion.gif" con la función `hacer_video(lista_nombres, archivo_salida)` que recibe una lista de nombres de archivos del tipo .png (por ejemplo `lista_nombres=["Lorenz_1800.png", "Lorenz_1900.png"]`) y el nombre de un archivo de salida (por ejemplo "Lorenz_evolucion.gif") y genera la animación. Por otro lado, para visualizar la animación pueden usar el paquete `Image` de `IPython.display`.

```
from pylab import savefig
import imageio
from IPython.display import Image

def hacer_video(lista_nombres, archivo_salida):
    lista_fotos = []
    for nombre_imagen in lista_nombres:
        lista_fotos.append(imageio.imread(nombre_imagen))
    imageio.mimsave(archivo_salida, lista_fotos, fps=800,**kargs)

... (plot) ...
savefig("Lorenz_1800.png")
...
... (plot) ...
savefig("Lorenz_1900.png")

kargs = { 'duration': 1 }
imagenes=["Lorenz_1800.png", "Lorenz_1900.png"]
hacer_video(imagenes, "Lorenz_evolucion.gif")
Image("Lorenz_evolucion.gif", format='png', height=500)
```

8. (*Optativo*): Armar algún gráfico inspirado en esta nota