# Actividad 2 - Explorando nuestro mundo (primera parte)

ExactasPrograma - Datos

Invierno 2020

En esta actividad vamos a explorar nuestro mundo a través de algunos datos abiertos. En particular, nos vamos a enfocar en algunos sets de datos recopilados por Gapminder, como PBI per cápita de los países durante los últimos 200 años, y posibles asociaciones entre este y otros indicadores que describen a los países.

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

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

1. **Obtener los datos**. Guardemos la base de datos de PBI per cápita de los países del mundo en un *dataframe* llamado `pbi`. Estos datos están disponibles en Gapminder. En el seleccionador de indicadores, seleccione Income y elija "Download as csv". También están disponibles en este URL.

   ```
   pbi = pd.read_csv("http://bit.do/ep2020inv-pbi", index_col=0)
   ```

   El parámetro `index_col=0` indica que la primera columna (recuerden que en python empezamos siempre con el 0) contiene etiquetas para las filas, los índices.

   ¿Qué información guarda cada fila y cada columna? ¿Cuántos países hay representados? ¿A qué fechas corresponden los valores de PBI? ¿Cuál es el primer año con datos registrados? y ¿el último?

   **Ayuda:** Explorar los comandos `pbi.columns`, `pbi.index`, `pbi.head()`, `pbi.tail()`, `pbi.shape`, `pbi.columns.min()`, `pbi.columns.max()`

2. Hacer un gráfico de PBI per cápita vs tiempo para Argentina entre 1900 y 2018, y agregar las curvas de otros dos países.

   **Ayuda:** `pbi[['1800','1850','1900']]` selecciona las columnas correspondientes a los años 1800, 1850 y 1900. En general `pbi_algunas_columnas = pbi[lista_de_columnas]` selecciona el subgrupo del *dataframe* `pbi` cuyas columnas se llaman como los elementos de la `lista_de_columnas`.

3. Leer otra base de datos de Gapminder con información geográfica de los países, y guardarlo en otro *dataframe* llamado `geo`. Explorar la información contenida en este *dataframe*.

   ```
   geo = pd.read_csv("http://bit.do/ep2020inv-geo", index_col='name')
   ```

4. **Combinando datos.** Implementar una función `agregar_region(df)` que recibe un *dataframe* con países en su índice y agrega al mismo una nueva columna 'region' con la región de cada país, según la clasificación de Gapminder en base a la columna `four_regions` de `geo`.

   - Verificar que hay 54 países de Africa y 57 países de Asia en el dataframe `pbi`.
   - ¿Cuál es el país que no tiene región asignada?

   **Ayuda:** Ver el comando `pbi.groupby('region').count()` ó el comando `pbi['region'].value_counts()`

5. ¿Cuáles son los 10 países con mayor pbi per cápita en al año 2000?

   **Ayuda:** Ver `pbi.sort_values(by=['2000'])` para ordenar el dataframe `pbi`.

6. Realizar un diagrama de barras que muestre el pbi per cápita de los 10 países más pobres y los 10 más ricos para un año elegido (por ejemplo 2000). Usar colores de las barras de acuerdo a la región del país.

7. Implementar una función `seleccionar_extremos(df, n, anio)` que recibe un *dataframe* `df` que está estructurado como `pbi`, un número entero `n`, y un año. Esta función debe ordenar `pbi` según la columna de PBI para el año elegido (`anio`) y devolver las primeras y últimas `n` filas en un nuevo dataframe. Mostrar en un mismo diagrama de barras los 10 países más ricos y los 10 más pobres para un año elegido (por ejemplo 2000).

   **Ayuda:** quizás es útil usar una escala logarítmica para el eje $y$, debido a las escalas tan distintas en el pbi per cápita de los países.

8. Graficar la distribución del PBI per cápita para el año 2000 como un histograma. Agregar a este gráfico una línea vertical en el valor promedio del PBI. ¿qué características tiene esta distribución?

9. Vamos a seguir explorando el mundo, a través de otros indicadores también recopilados por Gapminder. Guardar en un nuevo dataframe llamado `co2` los datos de las emisiones de dióxido de carbono per cápita y explorarlo (ver URL abajo).

```
co2 = pd.read_csv("http://bit.do/ep2020inv-co2", index_col=0)
```

10. Para un año en particular (por ejemplo 2000), armar un dataframe que tenga por índice a los países, y dos columnas: PBI y consumo de CO2 per cápita para ese año.

    **Ayuda:** Explorar el comando `pd.merge(df1,df2,on="col elegida",how='left')` para unir dos dataframes `df1` y `df2`. Podés construir un ejemplo sencillo con dos dataframes y explorar la función para diferentes valores del argumento `how` (`how='right'`, `how='outer'`)

11. Realizar un gráfico de puntos con las emisiones de PBI (eje x) vs dióxido de carbono (eje y), con colores indicando la región de cada país. Prueben de visualizar con escala logarítmica. ¿se ve una tendencia?

> **Nota:** Detrás de cualquier análisis de datos hay una responsabilidad enorme. Es de suma importancia distinguir entre una *asociación* (una variable provee información sobre otra), una *correlación* (se observa una tendencia creciente o decreciente entre una variable y otra), y *causalidad* (el cambio en una variable se debe al cambio en la otra).

12. Generar un dataframe que para un año elegido contenga los países como índices y columnas con PBI, CO2, Esperanza de Vida al nacer, que pueden descargar de Gapminder.

    **Ayuda en general:** En caso de que un archivo tenga los números decimales separados por comas, pueden especificarlo al leerlo usando el argumento `decimal` de `pd.read_csv`. También pueden especificar el delimitador usado para separar los campos (variables) del archivo original, usando el argumento `delimiter`.

    Por ejemplo: `pd.read\_csv("http://bit.do/ep2020inv-decimales", index\_col=0,decimal=",",delimiter=",")`

13. **Gráficos de Gapminder** Volver a realizar el gráfico de puntos de una variable vs otra (PBI (eje x) vs CO2 (eje y), PBI vs Esperanza de Vida, etc) para un año elegido (por ejemplo 2000), con colores por continentes y con el tamaño de los símbolos de acuerdo a la población del país. Para eso importe también los datos de poblaciones del archivo gapminder ("population_total.csv",

URL: "http://bit.do/ep2020inv-poblacion").

Puede ser útil construir un dataframe `unidos`, en el que el índice son los países y las columnas las variables de `PBI`, `CO2`, `POP` (población) para el año elegido, y la columna `region` con la información del continente.

- Exportar los gráficos (en un formato pdf o png).
- (optativo 1) Incluir una línea de tendencia, asumiendo una relación lineal entre el consumo de CO2 per cápita y el PBI. Aplicar el método de cuadrados mínimos descrito la clase pasada:

  Denotemos con $(x_i, y_i)$, $i = 1, \ldots, n$, a los pares observados. La recta de cuadrados mínimos está dada por $y = m^* x + b^*$, donde $(m^*, b^*)$ minimizan la función (de pérdida)

  $$L(m, b) = \sum_{i=1}^{n} (y_i - (mx_i + b))^2$$

  Se puede mostrar que esta función se minimiza en

  $$b^* = \overline{y}_n - m^* \overline{x}_n \,, \quad m^* = \frac{\sum_{i=1}^{n} (x_i - \overline{x}_n)(y_i - \overline{y}_n)}{\sum_{i=1}^{n} (x_i - \overline{x}_n)^2}$$

  $$\overline{x}_n = \frac{1}{n} \sum_{i=1}^{n} x_i \,, \quad \overline{y}_n = \frac{1}{n} \sum_{i=1}^{n} y_i$$

- (optativo 2) ¿Cuál es la expectativa de vida al nacer promedio por continente para cada uno de los años (entre 1950 y 2014)? Graficar la evolución.
- (optativo 3) Generar una secuencia de gráficos de PBI (eje x) vs Esperanza de vida (eje y) para diferentes años.

  Charla de Hans Rosling:

  ver aquí