**Laboratorio 3**

**Seccion 1**

Cargar el dataset con los numeros de muertes por año

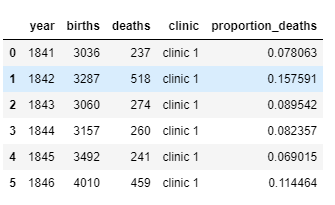
* Importe la librería pandas, com pd.
* Leer datasets/yearly\_deaths\_by\_clinic.csv  y asignarlo en la variable yearly.
* Imprimir yearly.



**Seccion 2**

Calcular la proporcion annual de muertes.

* Calcular la proporción de deaths por numero de births y almacene el resultado en una nueva columna llamada proportion\_deaths.
* Extraiga las filas de Clinic 1 en clinic\_1 y las filas de Clinic 2 en clinic\_2.
* Imprima clinic\_1.



**Seccion 3**

Graficar la proporcion de muertes de ambas clinicas.

* Importe matplotlib.pyplot como plt.
* Grafique proportion\_deaths por year para cada una de las clinicas en una sola gráfica. Use el método DataFrame.plot() .
  + Genere un grafico de líneas y utilizando el argumento label como argumento de .plot()etiquétela.
  + Cambie la etiqueta del eje y a "Proportion deaths" utilizando ylabel como parámetro en su segundo llamado a la función .plot().
* Salve los Axes como objeto retornado por el método plot  en la variable ax.

Para graficar, la manera más fácil es utilizer el método plot que está embebida en los DataFrames. Para conseguir dos lineas de la misma gráfica veremos una manera no explicada en clases. Si df1 y df2 son dos DataFrames puede graficar la data en ambas graficas como:

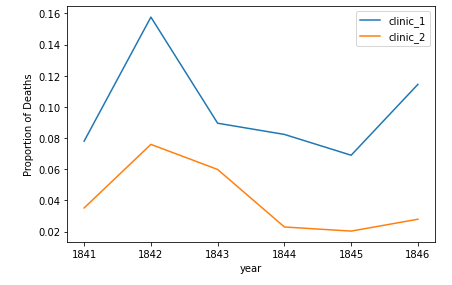
ax = df1.plot(**x**="col\_a", **y**="col\_b",

**label**="df1")

df2.plot(**x**="col\_a", **y**="col\_b",

**label**="df2", **ax**=ax, **ylabel**="Y Axis Label")

Capturando el objeto ax y dandolo como argumento en la siguiente gráfica agregamos graficas al grafico actual.



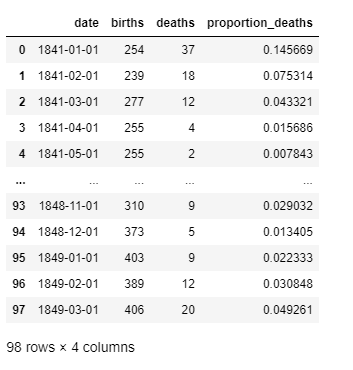
**Seccion 4**

Cargue el dataset con las muertes mensuales de la clinica 1.

* Leer el dataset datasets/monthly\_deaths.csv y asignelo en la variable monthly. Asegurese de que read\_csv use la columna date como argumento de parse\_dates en read\_csv.
* Calcule la proporcion de muertes deaths por numero de nacimientos births y almacene el resultado en la nueva columna monthly["proportion\_deaths"].
* Imprima las tres primeras filas de monthly usando el método .head().

La función read\_csv() no detecta automaticamente las columnas que son de format fecha. Se le puede notificar a read\_csv() esta dando una lista de fechas como la proporción de largumento parse\_dates. Por ejemplo, si datasets/my\_data.csv es un archive csv con la columna date entonces se debe leer de la siguiente manera:

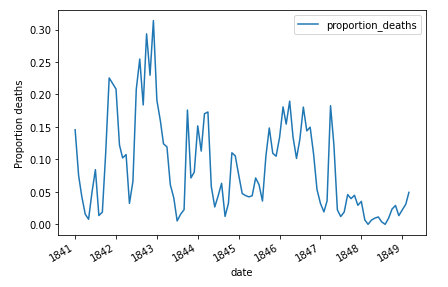
my\_df = pd.read\_csv("datasets/my\_data.csv", parse\_dates=["date"])



**Seccion 5**

Grafique la proporcion de muertes de la clinica 1.

* Grafique proportion\_deaths por date del dataset monthly utilizando el método.plot() method.
  + Cambie la etiqueta del eje y-axis a "Proportion deaths"
* Guarde el objeto Axes retornado por el método .plot() en la variable ax.



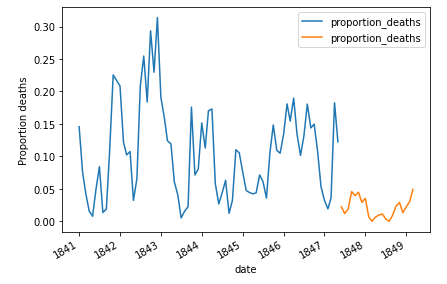
**Seccion 6**

Haga una gráfica que refleje los efectos del lavado de jabón. El código para definirhandwashing\_startesta provisto para usted utilizando pandas to\_datetime()*.*

* Dividir monthly en before\_washing (las filas en monthly antes handwashing\_start) y after\_washing (las filas en monthly y despues de handwashing\_start).
* Use el mismo procedimiento que utilize en la sección 3, grafiqeu proportion\_deaths en before\_washing y after\_washing en la misma gráfica. Nueavmente, use el DataFrame método .plot(), grabe el objeto de eje obtenido por el primer llamado de.plot() en la variable ax.
  + Etiquete las lineas graficadas usando label como argumento de .plot().
  + Cambie la etiqueta del eje y a "Proportion deaths" en su Segundo llamado de .plot().

Como la columna monthly["date"] fue leída como fecha, ahora podemos comparer diferentes fechas utilizando los operadres de comparación (<, >=, ==, etc.). Por ejemplo, para elegir la columna como handwashing\_start podemos escribir:

at\_washing = monthly[monthly["date"] == handwashing\_start]



**Seccion 7**

Calcular el promedio de reducción en proporción a las muertes por no lavarse las manos.

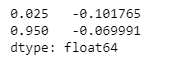
* Seleccione la columna proportion\_deaths en before\_washing y asignela a before\_proportion.
* Haga lo mismo para proportion\_deaths en after\_washing y asignela a after\_proportion.
* Calcule la diferencia en media de proporcion de muertes como: la media de after\_proportion menos la media de before\_proportion.



**Seccion 8**

Realice un análisis de bootstrap de la diferencia en media mensual de la proporción de muertes..

* En un for:
  + boot\_before y boot\_after deben ser muetreadas con remuestreo de before\_proportion y after\_proportion.
  + La diferencia en medias debe ser añadida a la lista boot\_mean\_diff.
* Calcule el 95% de confidence\_interval como el 2.5% y el 97.5% de los cuantiles de boot\_mean\_diff.



Un análisis bootstram es una manera rápida de tener un valor estimado de un valor incierto, en su caso el estimado es la diferencia media mean\_diff que calculó en la tarea 7. Un análisis de bootstrap trabaja para *simular* el no tener que rehacer la recolección de datos aleatoriamente de la data y permitir un valor ser muestreado varias veces. Utilizando un dataframe y la columna en pandas  my\_col (o una serie), esto puede ser realizado de la siguiente manera:

boot\_col = my\_col.sample(**frac**=1, **replace**=True)

El estimado es entonces calculado con boot\_col en ves de my\_col. Este proceso es repetido una gran cantidad de veces y la distribución de la muestra en bootstrap estima y representa la incertidumbre alrededor del estimado original. Si boot\_mean es una lista de estimados de bootstrap, el interval de confidencia al 95% se calcula con pandas:

**pd**.Series(**boot\_mean**).quantile([0.025, 0.975])

* Dada la información recolecatada de Semmelweis, es Verdadero o Falso que los doctores se deben lavar las manos. Explique su respuesta

Congratulations, you've made it this far! If you haven't tried it already, you should **check** your project now by clicking the "Check project" button.

Good luck! :)