

Fecha de Entrega: **Marzo 20 de 2015 antes de las 21:59 COT**

Instrucciones de Entrega

Todo el código fuente y los datos se debe encontrar en un repositorio público en github con un commit final hecho antes de la fecha de entrega. El nombre del repositorio debe ser `CM20151_HW4_NombreApellido`, por ejemplo yo debo crear un repositorio llamado `CM20151_HW4_SebastianPerez`. El link al repositorio lo deben enviar a través de **sicuplus** antes de la fecha/hora límite.

En cada parte del ejercicio se entrega 1/3 de los puntos si el código propuesto es razonable, 1/3 si se puede ejecutar y 1/3 si entrega resultados correctos.

1. 35 pt **Calentamiento Global?**

- (a) 15 pt Cree un script de R que descargue los datos históricos de temperatura para 5 ciudades de Colombia: Bogotá, Cali, Bucaramanga, Barranquilla e Ipiales ¹. El script debe generar como salida una tabla en representación limpia ² llamada `temperaturas.csv` con las siguientes columnas: `año,mes,fecha,ciudad,temperatura`. Asimismo, debe generar una gráfica en formato `.png` de calidad de artículo que muestre las series de tiempo para las 5 ciudades.
- (b) 13 pt Cree un script de python llamado `interp_temperaturas.py` que haga y grafique una interpolación lineal, polinómica y por splines (escoja polinomio y orden de los splines) para cada una de las ciudades anteriores. Cuál método es mejor? Justifique estadísticamente su escogencia.
- (c) 7 pt Cree un html (ya sea usando un cuaderno `.Rmd` o `ipython`) en el que establezca y justifique estadísticamente su posición con respecto al calentamiento global en referencia a los datos de las 5 ciudades anteriores.

¹Puese usar los datos de http://data.giss.nasa.gov/gistemp/station_data/ para ese propósito

²Tidy Data: vita.had.co.nz/papers/tidy-data.pdf

2. **30 pt** **Estudiando Rios y Oceanos** En este ejercicio, queremos ver el impacto de variables atmosféricas y térmicas en el comportamiento de rios y oceanos.
- (a) **10 pt** A partir de este catálogo de descargas hidrográficas ³, cree un script de python llamado `top_300_rios.py` que genere un archivo llamado `top_300_rios.csv` que solo contenga los con mayor tasa de flujo.
 - (b) **10 pt** Usando el módulo `Basemap`, cree un script de python que genere una gráfica en formato `.png` con la tasa de flujo de los principales 150 rios sobre un mapa.
 - (c) **10 pt** , cree un script de python llamado `interp_mapa.py` que grafique un archivo `.nc` con las temperaturas medias del aire ⁴. Haga una interpolación *nearest neighbors* de esos valores, genere un gráfico y comente.
3. **35 pt** **El poder de su voz**
- (a) **5 pt** Escriba un script en C llamado `grabar_mi_nombre.C` que grabe un archivo `.wav` en el que diga su nombre completo (con dos apellidos).
 - (b) **10 pt** Escriba un script llamado `grafica_mi_voz.py` que lea este `.wav` y grafique en un `mi_voz.png` la señal modulada de su voz. Tenga en cuenta las leyendas, nombres de los ejes y estética del gráfico.
 - (c) **10 pt** Escriba un script llamado `fft_de_mi_voz.py` que ejecute una transformada rápida de fourier para su nombre, y encuentre el armónico más grande. Este script debe generar una imagen llamada `mivoz_fft.png` Describa el método utilizado para encontrar dicho armónico.
 - (d) **10 pt** Haga un Makefile que genere las salidas descritas anteriormente en el orden adecuado y genere todas las salidas.

³<http://www.cgd.ucar.edu/cas/catalog/surface/dai-runoff/coastal-stns-byVol-updated-oct2007.txt>

⁴<ftp://ftp.cdc.noaa.gov/Datasets/ncep.reanalysis.derived/surface/air.mon.ltm.nc>