Probabilidades y Estadística **16 de mayo 2022**

Tarea 1: Variables Aleatorias, Vectores Aleatorios y Estadística Descriptiva

Profesores: *Elisa Irarrázaval, Ignacio Montegú*

1. Introducción

En la presente tarea, Ud. deberá aplicar sus conocimientos del curso para responder algunas preguntas prácticas. Para ello, deberá modelar un conjunto de problemas, ayudarse de librerías de `Python` que hagan parte del trabajo por usted, y elaborar un informe con análisis de los resultados obtenidos.

El trabajo debe efectuarse en grupos de 3 personas. Tomen en cuenta que este grupo será el mismo para resolver la Tarea 1 y la Tarea 2.

"Un tema es tener los datos, otro tema es saber qué hacer con ellos". El objetivo de esta tarea es que aprendan a trabajar con una base de datos *DataFrame* en `Python`, para que así puedan interactuar con variables y sus muestras, aplicando los conocimientos vistos en clases e investigando. También especificamente aprender a graficar e interpretar un set de datos para trabajar con ellos y darles interpretaciones y visualizaciones de interés.

2. Desarrollo

"Kaggle.[es] una página web especializada en temas de Data Science, que entrega una serie de bases de datos y estudios estadísticos de interés. Para esta tarea ustedes deben descargar la base de datos de la siguiente página www.Kaggle.com/Coronavirus, y utilizar el archivo llamado "COVID-19 Coronavirus.csv". Dentro del mismo enlace podrán encontrar la descripción de las variables. Deben hacerse un usuario para poder acceder al contenido, es recomendable entrar con su correo institucional. La base de datos a utilizar, junto con la descripción de las variables, está en la sección "Data". *Hint:* En la sección Çode"pueden encontrar una serie de estudios hechos con la misma base de datos, por lo que se pueden ahorrar muchos pasos de aprendizaje y al mismo tiempo indagar en temas estadísticos más complejos (recuerde usar referencias si es que utiliza algo de ahí, para que no sea plagio).

En base a sus conocimientos, responda las siguientes preguntas ejecutándo los comandos necesarios en `Python`:

(1) Grafique, para los 20 países con más contagios por millón de habitantes, el total de casos por país y luego, por país la cantidad de casos por millón de habitantes. Comente sus resultados.

(2) Genere un gráfico que permita ver la distribución de los datos que entregan la información sobre la cantidad de fallecimientos por millón de habitantes por continente. Comente sus resultados.

(3) Genere un gráfico que le permita ver la despersión entre la cantidad de casos por millón de habitantes y la cantidad de fallecidos por millón de habitantes. Comente sus resultados.

(4) Genere una tabla de frecuencia para la cantidad de casos por millón de habitantes para el continente europeo, junto con su gráfico de cantidades respectivo. Haga un proceso similar, pero ahora para Latinoamerica y el Caribe. Comente (y compare) los resultados destacando algo de interés.

(5) Con respecto a lo hecho en el ejercicio (4), btenga las principales medidas descriptivas de la variable "Casos por Millón de habitantes"¿Cómo es la simetría de la variable? Comente (y compare) los resultados destacando algo de interés.

(4) Genere un gráfico de interés que no haya realizado en items anteriores y comente al menos 2 observaciones que pueda obtener a partir de él.

3. Sobre la entrega

En un archivo comprimidio en `zip` deberá entregar:

- Un informe en `pdf` con portada y hasta diez páginas de contenido (incluyendo figuras). En la portada debe explicitarse el nombre completo y RUT de cada uno de los integrantes del grupo. En el informe, deben agregar todos los "outputs"del código (gráficos, tablas, etc.), respuestas a las preguntas y comentarios requeridos. No agreguén los códigos ejecutados.
- El código en `Python` utilizado en cada pregunta. El nombre de archivo deberá tener de la forma: `P` + *nro. de pregunta* + `.py` o `.ipynb` (p.ej., `P2.py`). Recuerden que cualquier código extraído de "Kaggle" o de otra fuente debe ser especificado, caso contrario será considerado como plagio. También recuerde que otro grupo no se considera como fuente válida de referenciar.

El archivo comprimido deberá tener por nombre `Tarea1_` + *Primer apellido de cada integrante del grupo* + `.zip`. P.ej., `Tarea1_SanchezVidalBravo.zip`. El incumplimiento de estas restricciones de nombres será sancionado con un descuento de un punto en la nota final de la tarea. El archivo comprimido deberá entregarse via buzón de Canvas. El plazo para entregar vence impostergablemente el **Martes 31 de Mayo a las 23:59 hrs.**

Aquél equipo que no entregue alguno de los documentos anteriormente específicados, o que entregue en plazo posterior al determinado, o que sus archivos no compilen con el dataset **original**, será evaluado con nota mínima 1,0. Cualquier caso de copia será calificado de la misma forma y será repasado a las autoridades correspondientes sujeto a sanciones adicionales.

4. Librerías de Python

En esta sección se entrega una ayuda introductoria para el uso de dos librerías que necesitará para esta tarea: `pandas` y `matplotlib` Finalmente se explica cómo instalarlas, si no están disponibles en el computador en que Ud. va a trabajar. Se asume que `Python` sí está instalado, junto con algún entorno de programación.

Si necesita más ayuda, recuerde que en internet puede encontrar tutoriales de ambas librerías y de `Python` en general para todos los niveles de dificultad, y que también puede preguntar a los profesores y ayudantes del curso.

4.1. Pandas para trabajar bases de datos

Necesitaran de esta libreria para trabajar de manera más ágil y fácil con las bases de datos. Se importa de la siguiente manera:

`import pandas as pd`

Si este paso arroja un mensaje de error, vea la sección 4.4.

Como norma general se le asigna la abreviación "pd". La recomendación es que mantengan esa norma porque se les hará más fácil encontrar bibliografía de esta manera. Para llamar a la librería solo deben usar `pd.` seguido de la función de pandas que quieran ocupar.

Dentro de las ventajas de usar esta librería es que permite crear tablones a partir de archivos y luego crear nuevas variables al aplicar funciones comunes de `python` en sus columnas manteniendo el orden original de la base de datos. Además, encontrarán de utilidad que la librería permite realizar filtros tanto a nivel de variables como de observaciones (filas), dándoles la oportunidad de crear nuevos DataFrames más pequeños y fáciles de manejar.

En este enlace encontrarán un resumen con los comandos más conocidos de `pandas`

4.2. Para cargar la base de datos

Una vez que se importe `pandas`, podrán importar desde un csv (archivo de valores separado por comas) la base de datos y crear un objeto DataFrame para poder seguir usandola.

Este comando es uno de los que cumple esa función:

`bdd = pd.read_csv("Mi_Archivo.csv")`

Para que no haya errores, es importante que `Mi_Archivo.csv` se encuentre en la misma carpeta que el archivo `python` (`.py` o `.ipynb` si utilizan Jupyter Notebook)

4.3. Gráficos

La librería `matplotlib` es la más completa y simple de usar gracias a su fácil integración con pandas. Se importa de la siguiente manera:

`import matplotlib as mpl`

Si este paso arroja un mensaje de error, vea la sección 4.4.

En este enlace encontrarán un resumen de sus comandos más conocidos.

4.4. Instalación de librerías

Si una librería no está instalada junto a `Python` en un computador dado, la consola entregará un mensaje de error de la forma `ImportError: No module named 'pepito'`. Para instalarla, escriba las siguientes dos líneas de código, reemplazando "pepito" por el nombre de la librería, entre comillas:

```
import pip
pip.main(["install", "pepito"])
```