



IIC2115 – Programación como Herramienta para la Ingeniería (I/2020)

## Laboratorio 5 - Análisis y Visualización de datos

### Objetivos

- Aplicar los contenidos de análisis y visualización de datos para procesar información y tomar decisiones en base a esto.

### Entrega

- **Lenguaje a utilizar:** Python 3.6
- **Lugar:** repositorio privado en GitHub. Recuerde incluir todo en una carpeta de nombre **L05**.
- **Entrega:** Domingo 28 de junio a las 23:59 hrs.
- **Formato de entrega:**
  - Archivo python notebook (**L05.ipynb**) con su código. Utilice múltiples celdas de texto y código para facilitar la revisión de su laboratorio.
  - Archivo python (**L05.py**) con el mismo código entregado en el notebook.
  - Todos los archivos deben estar ubicados en la carpeta **L05**. No se debe subir ningún otro archivo a la carpeta. Los archivos **.ipynb** y **.py** deben contener la misma solución. No incluya las bases de datos en esta entrega.
- **Descuentos:** El descuento por atraso se realizará de acuerdo a lo definido en el programa del curso. Además de esto, tareas que no cumplan el formato de entrega tendrán un descuento de 0.5 pts.
- **Laboratorios con errores de sintaxis y/o que generen excepciones serán calificadas con nota 1.0.**

- Si su laboratorio es entregado fuera de plazo, tiene hasta el **Lunes 29 de junio a las 23:59 hrs** para responder el formulario de **entregas fuera de plazo** disponible en el Syllabus.
- Las discusiones en las *issues* del Syllabus en GitHub son parte de este enunciado.
- El uso de librerías externas que sean estructurales en la solución de los problemas no podrán ser utilizadas. Solo se podrán utilizar las que han sido aprobadas en las *issues* de GitHub.
- **Todos los comandos e instrucciones importantes de cada solución deben ser implementados usando funciones de *pandas*, *matplotlib* y/o *sklearn*. Esto significa que no está permitido iterar sobre los DataFrame, calculando con funciones básicas de Python lo requerido.**

## Introducción: En búsqueda de una herramienta urbana parte 2

Debido al éxito del concurso para el desarrollo de la herramienta urbana definitiva, el Gobierno ha abierto una nueva convocatoria que busca potenciar aún más el análisis de los datos disponibles. En particular, se busca el uso de herramientas especializadas en el análisis y la visualización de datos.

Usted ha visto el nuevo anuncio y ha decidido participar. Para ello, utilizando el conocimiento y los datos correspondientes a la herramienta anterior, para cumplir una nueva serie de misiones, esta vez relacionadas con el uso del módulo *pandas* de Python.

## Misiones para la herramienta

Para completar requerimiento solicitado por el Gobierno, usted deberá crear una herramienta que sea capaz de realizar las tareas descritas en cada una de las misiones siguientes:

**M1. Integración de datos:** Su primera misión será traspasar los datos utilizados en el **L04** a DataFrames de *pandas*. Para cumplir con esto, deberá realizar siguientes dos pasos:

- Importe los datos generados en la **M1** del **L04**, directamente desde las tablas en SQLite a DataFrames, generando uno para Predios y otro para líneas de construcción. **(0,3 puntos)**
- Importe la información de los archivos **.csv**, incorporándola directamente en los DataFrames. **(0,3 puntos)**

- M2. **Limpieza y depuración:** Analice la existencia de valores por defecto, nulos, incompletos y outliers. Para cada uno de estos casos, mantenga, complete o modifique la información de acuerdo a algún criterio debidamente justificado. **(0,6 puntos)**
- M3. **Comprobación de compatibilidad:** Repita las misiones M4.(a), M5.(a) y M5.(b) del L04, esta vez utilizando las operaciones de *pandas* sobre **DataFrames**. Retorne para cada pregunta un **DataFrame** con la información requerida. **(1,2 puntos)**
- M4. **Rankeando los predios:** Construya y visualice un ranking de los predios en base a su densidad de construcción promedio. Para esto, investigue sobre la operación **merge** en *pandas* y haga las agregaciones y modificaciones necesarias en los **DataFrames** para poder usarla. **(1,2 puntos)**
- M5. **Completando la tabla comunas parte 2:** Construya un **DataFrame** con la misma información de la tabla **Comunas** de la misión M6 del L04. Posteriormente, construya y visualice un ranking de las comunas, en base a la antigüedad promedio de sus construcciones. Utilice las funciones de agregación de *pandas* para completar los campos requeridos. **(1,2 puntos)**
- M6. **Predicción del avalúo fiscal:** Construya una función que permita predecir el avalúo fiscal de un predio en base a un conjunto de características de este. Construya estas características a partir de la información disponible para cada predio y utilice las funciones de *scikit-learn* para construir el predictor. **(1,2 puntos)**

## Corrección

Es importante que deje ejecutado todo su trabajo antes de subirlo, de lo contrario se le aplicará un descuento al puntaje total. Para la corrección de este laboratorio, se revisarán los procedimientos desarrollados para responder las diferentes misiones propuestas y la estructura de como utiliza los módulos *pandas*, *matplotlib* y/o *sklearn* en ellos. Dado lo abierto de las misiones, se espera que las respuestas incluyan análisis exploratorios y visualizaciones que permitan justificar las decisiones tomadas.

## Política de Integridad Académica

*“Como miembro de la comunidad de la Pontificia Universidad Católica de Chile me comprometo a respetar los principios y normativas que la rigen. Asimismo, prometo actuar con rectitud y honestidad en las relaciones con los demás integrantes de la comunidad y en la realización de todo trabajo, particularmente en aquellas actividades*

*vinculadas a la docencia, el aprendizaje y la creación, difusión y transferencia del conocimiento. Además, velaré por la integridad de las personas y cuidaré los bienes de la Universidad.”*

En particular, se espera que mantengan altos estándares de honestidad académica. Cualquier acto deshonesto o fraude académico está prohibido; los alumnos que incurran en este tipo de acciones se exponen a un procedimiento sumario. Ejemplos de actos deshonestos son la copia, el uso de material o equipos no permitidos en las evaluaciones, el plagio, o la falsificación de identidad, entre otros. Específicamente, para los cursos del Departamento de Ciencia de la Computación, rige obligatoriamente la siguiente política de integridad académica en relación a copia y plagio: Todo trabajo presentado por un alumno (grupo) para los efectos de la evaluación de un curso debe ser hecho individualmente por el alumno (grupo), sin apoyo en material de terceros. Si un alumno (grupo) copia un trabajo, se le calificará con nota 1.0 en dicha evaluación y dependiendo de la gravedad de sus acciones podrá tener un 1.0 en todo ese ítem de evaluaciones o un 1.1 en el curso. Además, los antecedentes serán enviados a la Dirección de Docencia de la Escuela de Ingeniería para evaluar posteriores sanciones en conjunto con la Universidad, las que pueden incluir un procedimiento sumario. Por “copia” o “plagio” se entiende incluir en el trabajo presentado como propio, partes desarrolladas por otra persona. Está permitido usar material disponible públicamente, por ejemplo, libros o contenidos tomados de Internet, siempre y cuando se incluya la cita correspondiente.