



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

Laboratorio 5 - Herramientas avanzadas de Python

Objetivos

- Aplicar los contenidos de SIG en el análisis y procesamiento de información geográfica para generar nueva información
- Utilizar herramientas de manejo de redes para modificar redes de transporte
- Aplicar herramientas de *web scrapping* y expresiones regulares para complementar información desde la web

Entrega

- **Lenguaje a utilizar:** Python 3.6
- **Lugar:** repositorio privado en GitHub. Recuerde incluir todo en una carpeta de nombre **L05**.
- **Entrega:** Domingo 6 de diciembre 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 7 de diciembre a las 11:59 am** 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*, *geopandas*, *networkx*, *osmnx*, *bs4* y/o *pyrematch*. Esto significa que no está permitido iterar sobre los `GeoDataFrame`, calculando con funciones básicas de Python lo requerido.**

Introducción: Estudio sobre Rancagua, excepto Rancagua

El cuerpo docente está expectante en que usted realice juvenilmente un estudio de transportes en Rancagua, excepto Rancagua. Este estudio busca conocer información de las no existentes vialidades de Rancagua y evaluar su transporte público.

Un estudio se dice que es realizado juvenilmente cuando es llevado a cabo por alguien como usted, donde se utilicen herramientas específicas de sistemas de información geográfica (SIG), manipulación de redes y *web scrapping* con expresiones regulares.

Lamentablemente, no hay datos completos para Rancagua, por lo que el estudio utilizará exclusivamente con información de Santiago. Además, la fuente principal de información es obtenida de la página web de Uber, que almacena tiempos de viaje entre zonas de la ciudad agregados a nivel de hora. Los datos de Uber están compuestos por dos tipos de información:

- **Zonas:** Archivo JSON que contiene las coordenadas de las geometrías poligonales que utiliza Uber para entregar los tiempos de viaje.
- **Tiempos de viaje:** Archivo CSV con una matriz origen-destino de tiempos de viaje entre las zonas definidas.

Misiones para completar el estudio

Para completar el estudio, usted deberá completar cada una de las siguientes misiones.

M1. Manipulación de tiempos de viaje: Su primera misión será crear los DataFrames (un GeoDataFrame y un DataFrame) con la información proporcionada por Uber. Investigue qué es un archivo JSON o GeoJSON (puede comenzar abriéndolo con el editor de texto, quizás note una estructura muy familiar). Para resolver esta misión guíese por los siguientes pasos:

- (a) Cree un GeoDataFrame de zonas a partir del JSON. Genere una visualización de las zonas con el método `plot()`. **(0.2 ptos.)**
- (b) Cree un DataFrame de tiempos de viaje a partir del CSV. Genere una visualización de los datos con el método `head()`. **(0.1 ptos.)**
- (c) Cree un método que reciba el GeoDataFrame de la parte (a), el DataFrame de la parte (b) y el id de una zona origen cualquiera. Su método debe crear una nueva columna en el GeoDataFrame con los tiempos de viaje a todas las zonas desde la zona indicada en el *input* (id de la zona). Genere una visualización de los tiempos de viaje en la zonas con el método `plot()` de **geopandas**. **(0.7 ptos.)**

M2. Velocidades a nivel de red: En esta misión deberá añadir información de velocidades a los arcos de la red vial de Santiago para distintos tramos horarios. Para llevar a cabo la misión, guíese con los siguientes pasos:

- (a) Utilice el archivo Shape de las zonas urbanas de Chile para obtener la zona urbana de Santiago. A partir de la zona urbana obtenida, obtenga mediante la librería de **osmnx** la vialidad contenida dentro del polígono la zona urbana. **(0.2 ptos.)**
- (b) Defina una metodología propia que permita incorporar a la vialidad las velocidades presentes en arcos de la red vial. Las velocidades deben ser obtenidas a partir de la información de tiempos de viaje generada en la **M1**. Debe explicar detalladamente la lógica utilizada y justificarla para que tenga sentido. Puede ayudarse con los métodos disponibles en la librería de **osmnx**, **geopandas** o cualquier otra librería que encuentre útil (referenciando cuando corresponda). Finalizada esta parte, deberá tener un Shape de la red vial en donde cada geometría de vialidad (LineString) incluya información de la velocidad para distintos horarios del día. Ustedes es libre de definir el nivel de agregación temporal en base a los datos, siempre y cuando tenga al menos cuatro horarios

diferentes. Genere una visualización con esquema de color de las velocidades para cuatro horarios diferentes. **(2.3 ptos.)**

M3. Incorporación de una nueva característica: En esta misión deberá buscar por internet algún tipo de información que pueda añadir a sus datos (que sea geo-referenciable de alguna forma). Para completar esta misión deberá hacer uso de *web scrapping* para obtener la información de internet. Ayúdese con el tutorial de expresiones regulares (disponible en el Syllabus) para procesar los datos obtenidos. Genere una visualización de la ciudad de Santiago que incluya esta característica. Es libre de generar la visualización de mapa que desee siempre y cuando incluya relación con la característica obtenida **(1.5 ptos.)**

M4. Indicadores en Transporte Público: En esta última misión deberá utilizar el Shape de los trazados de los servicios de Red (ex-Transantiago) y generar un ranking con ellos. El ranking debe ser realizado a en base a un indicador construido por usted. El único requisito en la construcción del indicador es que se utilice la información obtenida en las misiones **M2** y **M3**. Debe pasar la información a los servicios y determinar el indicador en cada uno de ellos. Finalmente cree una visualización de mapa de este indicador para los servicios de Red. **(1.0 pto.)**

Corrección

Es importante que deje ejecutado todo su trabajo antes de subirlo, de lo contrario se le aplicará un descuento de 1 punto al puntaje total. Para la corrección de este laboratorio, se revisarán los procedimientos desarrollados para responder las diferentes misiones propuestas y los argumentos utilizados. Además se evaluará cómo utiliza los módulos de *pandas*, *geopandas*, *networkx*, *osmnx*, *bs4* y/o *pyrematch*. Dado lo abierto de las misiones, se espera que las respuestas incluyan análisis 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.