Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ingeniería de Transporte y Logística

ICT3115 – Sistemas Urbanos Inteligentes (I/2022)

Tarea 1: redes neuronales para elección de modo

Introducción

En esta tarea tendrá la oportunidad de experimentar con el uso de redes neuronales para el procesamiento de datos tabulados. En particular, deberá entrenar MLPs para un problema de elección de modo, en base a datos numéricos y categóricos, y utilizando *embeddings* cuando corresponda. Para el desarrollo se utilizará el framework Pytorch sobre la plataforma Colab de Google. Dado que existe abundante material disponible en línea relacionado con el desarrollo de la tarea, se espera que todo recurso externo utilizado, sea este código o librerías, esté debidamente indicado.

Set de datos

La fuente primaria de datos para entrenar los modelos será el set "Swissmetro". Este conjunto consta de datos de encuestas recopilados en los trenes entre St. Gallen y Ginebra, Suiza, durante marzo de 1998. Los encuestados proporcionaron información para analizar el impacto de la innovación modal en el transporte, representada por el Swissmetro, un revolucionario tren subterráneo de levitación magnética, frente a los modos de transporte habituales representados por el vehículo particular y el tren.

El conjunto contiene más de 10 mil registros, donde cada uno considera los atributos para cada modo y la elección realizada por el encuestado. Información detallada para cada una de las variables puede ser obtenida acá. Además, en el sitio del curso podrá encontrar ejemplos de código que permiten cargar y filtrar los datos.

Modelos

Para esta tarea, debe utilizar modelos de redes profundas como los descritos en el capítulo 2 del curso, con la exigencia de que debe considerar el uso de *embeddings* para las variables categóricas. Se recomienda revisar la bibliografía ubicada al final del enunciado para esto y además diseñar las capas teniendo en consideración el tipo de dato que procesará (por ejemplo, en el caso de las variables categóricas, el número de categorías es relevante). Considere además preprocesar las entradas y salidas numéricas (revise el ejemplo de código disponible en el sitio del curso). No hay problema en basarse completamente en algún modelo propuesto previamente en la literatura o en tutoriales, siempre y cuando el código sea escrito por ud. En cualquier caso, debe justificar su elección de modelo.

Actividades a realizar

Para la tarea se espera se espera que realice al menos las siguientes actividades:

- **Entrenamiento de MLPs "puros"**: entrene MLPs para predecir el modo elegido por las personas encuestadas, utilizando todas las variables disponibles y evaluando los rendimientos utilizando las métricas que considere adecuadas en un set de prueba independiente. Considere al menos las siguientes configuraciones:
  - MLPs de 1 y 2 capas (0 o 1 oculta y 1 de clasificación), con capa de clasificación con pesos únicos para cada modo (categoría).
  - MLPs igual al anterior, pero con capa de clasificación con pesos compartidos entre los modos.

  Para ambos casos puede (debe) considerar el uso de otro tipo de capas, como dropout, batch normalization, etc. Seleccione el modelo que mejor rendimiento entrega y justifique su elección en base a las características de la arquitectura y del problema.

- **Entrenamiento de MLPs con embeddings categóricos**: repita la actividad del ítem anterior, pero considerando el uso de embeddings categóricos para aquellas variables que considere adecuado. Pruebe al menos las siguientes configuraciones:
  - MLPs de 1 y 2 capas con embeddings (capas de embeddings independientes para cada variable que lo necesite, 0 o 1 oculta y 1 de clasificación), con capa de clasificación con pesos únicos para cada modo (categoría).
  - MLPs igual al anterior, pero con capa de clasificación con pesos compartidos entre los modos.

Para ambos casos puede (debe) considerar el uso de otro tipo de capas, como dropout, batch normalization, etc. Seleccione el modelo que mejor rendimiento entrega y justifique su elección en base a las características de la arquitectura y del problema. Además de esto, se espera que existan visualizaciones para los espacios de embeddings relevantes.

- **Bonus - Autoencoders:** investigue sobre la arquitectura Autoencoder y aprenda representaciones de los datos considerando las siguientes dos configuraciones:
  - Autoencoder alimentado por la mismos datos de las actividades anteriores. La representación aprendida para cada ejemplo será el vector latente generado antes del proceso de decodificación.
  - 3 Autoencoders independientes, cada uno recibiendo como entrada solo los datos pertenecientes a un modo en particular y los comunes. La representación aprendida para cada ejemplo será la concatenación de los vectores latentes generados por cada Autoencoder antes del proceso de decodificación.

  Una vez obtenidas las representaciones, entrene con ellas MLPs similares a los de la primera actividad y seleccione el esquema que mejor rendimiento entrega y justifique su elección en base a las características de la arquitectura y del problema.

Desarrollo y entrega

La tarea puede desarrollarse de manera individual o en parejas, utilizando el framework Pytorch para Python. Se recomienda utilizar la plataforma Google Colab con el fin de facilitar la instalación de librerías. Esta plataforma permite utilizar gratuitamente una GPU para el entrenamiento por intervalos de 12 horas continuos. En el *notebook* desarrollado debe ir tanto el código como un informe (preferiblemente intercalados), donde se expliquen los pasos realizados, se analicen los resultados y se planteen conclusiones. La entrega de la tarea tiene como fecha límite el lunes 16 de mayo a las 23:59, a través del buzón que se habilitará en el sitio del curso. Para fines de corrección, se revisará la última versión entregada.

Referencias

Política de Integridad Académica

Los alumnos de la Escuela de Ingeniería deben mantener un comportamiento acorde al Código de Honor de la Universidad:

"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.