

# Proyecto Corto #1 - Simulador Votantes

## Inteligencia Artificial

Grupo 1, I Semestre 2018

Un problema significativo al desarrollar agentes de inteligencia artificial radica en la poca disponibilidad de conjuntos de datos de ejemplo. Si bien es cierto existen proyectos abiertos que dan acceso a datos en ciertos dominios, la aplicabilidad de los mismos es debatible y no necesariamente reflejan casos de la vida real.

Una manera de evitar este problema es generar datos sintéticos. Esto es, asumiendo que sabemos la distribución teórica de los datos, crear programas que obtengan muestras extraídas de la distribución para emular la población.

En este proyecto se pide crear un módulo de Python que contiene funciones para crear muestras de ejemplo, basados en los resultados electorales presidenciales de la primera ronda en Costa Rica.

### Módulo simulador

Se pide a los estudiantes crear dos funciones primarias:

- *generar\_muestra\_pais(n)*: retornará una muestra simulada con  $n$  votantes con etiquetas y atributos proporcionales a la distribución de resultados nacionales.
- *generar\_muestra\_provincia(n, nombre\_provincia)*: retornará una muestra simulada siguiendo la distribución de datos para una provincia específica (donde *nombre\_provincia* es uno de "SAN JOSE", "ALAJUELA", "CARTAGO", "HEREDIA", "GUANACASTE", "PUNTARENAS", "LIMON")

El resultado de llamar a cada función será una lista de listas de Python. El tamaño de la lista externa será  $n$ , mientras que cada lista interna tendrá  $[\text{numero\_atributos} + 1]$  elementos. Los atributos corresponden a características cantonales recolectadas en el censo 2011. La etiqueta, que es la columna adicional a los atributos, será la representación textual de un partido político (nombre en mayúscula y de la manera escrita en las actas de escrutinio) o bien las palabras "NULO" o "BLANCO" cuando corresponda.

Para generar las muestras se deberá seguir el algoritmo básico aleatorizado descrito en clase (favor no usar otro algoritmo sin consultar con el profesor). El objetivo básico de dicho algoritmo

es representar la masa poblacional de cada cantón y las preferencias de voto a este nivel geográfico, procurando que las muestras generadas sean fieles a estas distribuciones.

### Fuentes de datos

Existen 3 fuentes de datos primarias:

- **Actas de escrutinio:** Contienen los totales de votaciones por junta receptora y se pueden descargar a través de

[http://www.tse.go.cr/elecciones2018/actas\\_escrutinio.htm](http://www.tse.go.cr/elecciones2018/actas_escrutinio.htm)

- **Mapeo juntas a cantones:** Dado que los resultados de las juntas no contienen explícitamente el cantón, tendremos que reproducirlo con la información contenida en:

<http://www.tse.go.cr/pdf/nacional2018/JRV.pdf>

- **Indicadores cantonales:** Corresponden a los atributos que nos permitirán generar predicciones para cada uno de los ejemplos. Estos se extraen del Censo 2011 y se encuentran almacenadas en

[https://www.estadonacion.or.cr/files/biblioteca\\_virtual/otras\\_publicaciones/Indicadores-cantonales\\_Censos-2000-y-2011.xlsx](https://www.estadonacion.or.cr/files/biblioteca_virtual/otras_publicaciones/Indicadores-cantonales_Censos-2000-y-2011.xlsx)

Los estudiantes estarán a cargo de curar esa información, y generar una serie de archivos CSV, que puede usarse como entrada al módulo.

### Entregables

- Código fuente de las dos funciones mencionadas contenidas en un módulo llamado `tec.ic.ia.pc1.g##`, con `pc1` representado Proyecto Corto 1, y `g##` que se reemplazará por el número de grupo asignado por el profesor. Esto quiere decir, por ejemplo, que una vez entregada la tarea podríamos tener la siguiente cláusula de importación para llamar a las funciones:

```
from tec.ic.ia.pc1.g01 import generar_muestra_pais, generar_muestra_provincia
```

Lo anterior será vital ya que parte de la revisión será realizada asumiendo que se pueden importar las funciones de esa forma. Nótese que esto implica que la estructura de carpetas debe ser esa en específico.

- Los archivos CSV utilizados por las funciones, deben ser agregados al paquete enviado. Estos mismos archivos deberán ser leídos por el código desarrollado. Crear variables globales al módulo que nos permitan cambiarlos fácilmente, es un requerimiento para

todos los archivos (idealmente usamos parámetros a la consola, pero se dejará este nivel para el proyecto).

- Pruebas unitarias. Si bien es cierto no se espera que haya pruebas que revisen con rigurosidad estadística si las muestra simuladas en efecto vienen de la distribución de los datos, si se pide que haya pruebas que puedan revisar las funcionalidades. Por ejemplo, los estudiantes podrían "inyectar" un generador de números aleatorios para pruebas, cuyos valores son determinísticos, permitiéndonos escribir pruebas. Las mismas deben realizarse utilizando pytest.
- Un informe, en formato PDF, sobre la comparación de resultados obtenidos durante múltiples ejecuciones del simulador. Si todo estuviera bien, al correr las funciones muchas veces debería crear, en promedio, muestras que aproximen con relativa certeza. Debe mostrarse análisis nacionales y para cada provincia. El nivel de detalle del análisis tendrá un impacto directo en la calificación de este rubro.

### Detalles de Entrega y Revisión

El proyecto corto deberá realizarse en grupos de 3 personas. Se debe enviar todos los entregables a través de TEC Digital a más tardar el 25 de marzo del año en curso.

El profesor se reserva el derecho a asignar una nota de cero si los requerimientos no son cumplidos de forma tal que impida ejecutar las funciones principales.

### Consideraciones Python

- Se utilizará Python 3 (no 2.7)
- Se utilizará pip para instalar el módulo enviado. Se debe respetar la estructura de directorios apropiada para que la instalación sea exitosa.
- Se debe utilizar pytest para pruebas unitarias
- Seguir recomendaciones en <http://docs.python-guide.org/en/latest/>
- Seguir PEP 20. Una guía rápida con ejemplos está disponible en: [http://artifex.org/~hblanks/talks/2011/pep20\\_by\\_example.pdf](http://artifex.org/~hblanks/talks/2011/pep20_by_example.pdf)
- Utilizar PEP 8 como guía de estilo: <http://pep8.org/>