Trabajo Práctico 1: Generación de Datos, Ajuste de Modelos y Visualización

Introducción

Para comenzar a trabajar con datos en Python 3 realizaremos el trabajo de codear una serie de generadores de datos con distribuciones específicas, que nos serán útiles a lo largo de la materia. Luego continuaremos poniendo en práctica el ajuste y la evaluación de algunos modelos de aprendizaje sencillos con datos generados por los generadores propios. Finalmente realizaremos algunas exploraciones visuales sobre un par de datasets.

Utilizaremos en la materia Jupyter Notebooks, que permiten entremezclar código Python 3, imágenes generadas por código y texto Markdown, muy útiles para realizar exploraciones de datos y mostrar resultados.

Se puede descargar el entorno Jupyter para Python 3 y correr el código y el renderizado de los notebooks en la máquina local, o se puede utilizar Google Colab que provee un entorno de programación, ejecución de código y renderizado de notebooks en la nube. Sin embargo se debe tener en cuenta que para futuros trabajos prácticos puede ser impráctico el uso de Colab ya que las sesiones tienen un límite de tiempo de uso, y se deberán correr localmente o en un servidor Jupyter alternativo, sin límites de tiempo.

Se pide entregar uno o más Notebooks (recordar que el archivo .ipynb guarda internamente el output generado al correr el código, las figuras y el texto). Sería deseable que se pueda correr entero sin lanzar errores.

Ejercicio 1

Prepare código Python que genere conjuntos de datos (de longitud dada n y tomando parámetros adicionales) de acuerdo a las siguientes descripciones:

a) **Diagonal**
   Los datos tienen d inputs, todos valores reales, correspondientes a la posición de un punto en un espacio d-dimensional. El output es un factor binario correspondiente a la clase a la que pertenece el ejemplo.
   La clase 1 corresponde a puntos generados al azar, provenientes de una distribución normal con en centro en el $(1,1, ..., 1)$ y matriz de covarianza diagonal (todas las variables son independientes entre sí), con desviación estándar igual a $C * \mathrm{sqrt}(d)$.
   La clase 0 tiene la misma distribución, pero está centrada en el $(-1, -1, ..., -1)$.
   Los parámetros a ingresar son d (número de dimensiones) y n (número de ejemplos), ambos enteros, y C (parámetro de la desviación estándar) número real.
   De los n puntos generados, $n/2$ deben pertenecer a cada clase (se puede asumir a n

Trabajo Práctico 1: Generación de Datos, Ajuste de Modelos y Visualización

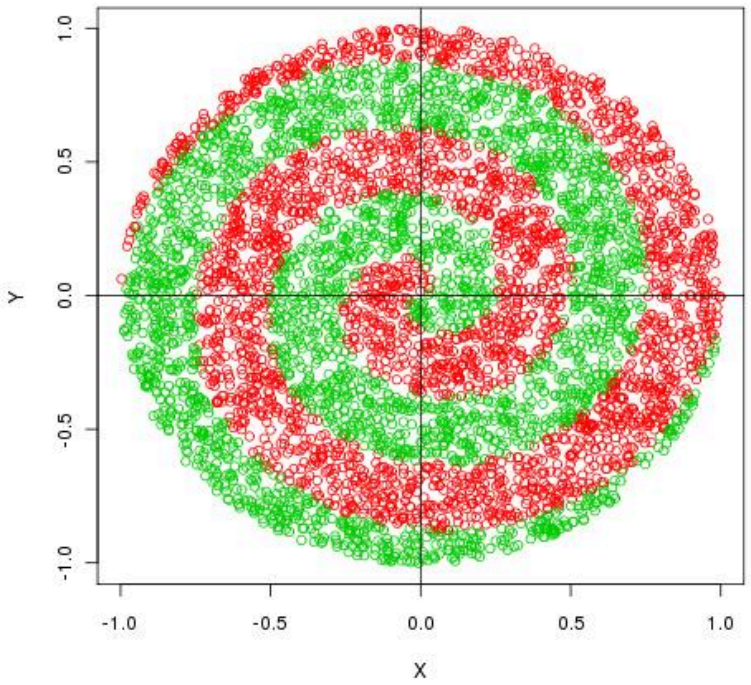
par, o agregar o restar un punto a la generación).
La función debe generar como salida un dataframe con $d+1$ columnas y n filas.

Se puede encontrar información sobre Gaussianas multidimensionales y el caso especial de una matriz diagonal en http://cs229.stanford.edu/section/gaussians.pdf (secciones 1 y 3).

b) **Espirales Anidadas**
Los datos tienen 2 inputs, x e y, que corresponden a puntos generados al azar con una distribución uniforme dentro de un círculo de radio 1. El output es un factor binario correspondiente a la clase a la que pertenece el ejemplo.
La clase 0 corresponde a los puntos que se encuentran entre las curvas

$\rho = \theta \,/\, (4*\pi)$ y $\rho = (\theta + \pi) \,/\, (4*\pi)$ (en coord. polares)

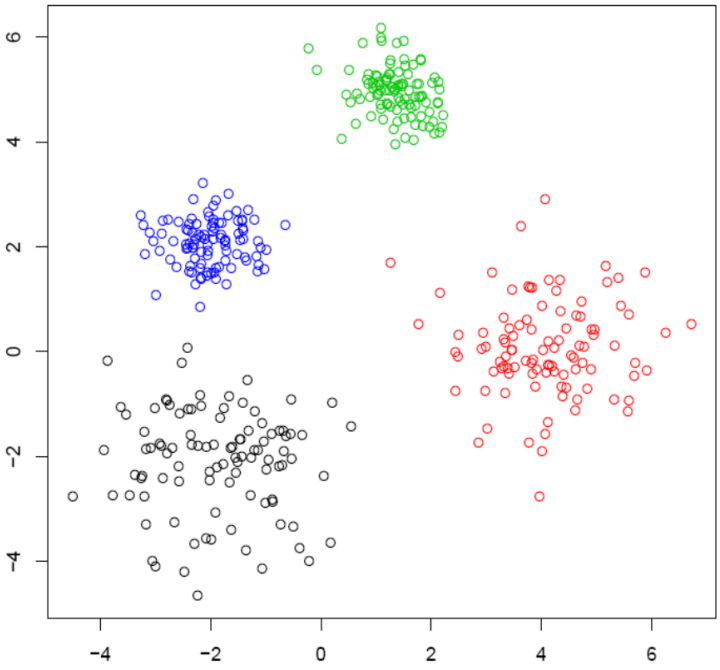
La clase 1 corresponde a los puntos restantes.
De los n puntos generados, $n/2$ deben pertenecer a cada clase (se puede asumir a n par, o agregar o restar un punto a la generación).
La función debe generar como salida un dataframe con 3 columnas y n filas.



c) **N-Gausianas** (opcional)
Los datos tienen d inputs, todos valores reales, correspondientes a la posición de un

Trabajo Práctico 1: Generación de Datos, Ajuste de Modelos y Visualización

punto en un espacio d-dimensional. El output es un factor n-ario correspondiente a la clase a la que pertenece el ejemplo.
Se provee una lista de N centros d-dimensionales, y N valores de desvío estándar. Cada clase corresponde a puntos generados al azar, provenientes de una distribución normal d-dimensional con origen en el centro que corresponda a dicha clase, y con el desvío estándar dado correspondiente (usamos gaussianas independientes como las del ejercicio a).



Ejercicio 2

Genere un conjunto de datos para entrenamiento chico (300 puntos) y uno para test grande (10000 puntos) con cada uno de los generadores anteriores (los datos que faltan están en el notebook base).

Ajuste un clasificador de árboles y uno de k-vecinos para cada uno de los conjuntos de entrenamiento, y mida el error de test con el conjunto correspondiente.
Repita el procedimiento usando únicamente el conjunto de entrenamiento y estimando el error con 5-fold cross-validation.

Compare los resultados obtenidos entre ambos métodos de estimación de error.

Trabajo Práctico 1: Generación de Datos, Ajuste de Modelos y Visualización

Ejercicio 3

Visualize los datasets Iris y Titanic utilizando los métodos listados a continuación. ¿Qué dimensiones piensa que separan mejor los datos para cada uno? ¿Se pueden ver fácilmente outliers con algún método? ¿Cómo afecta el método de visualización al análisis de estos factores?

- Biplots de Scatterplots
- Coordenadas Paralelas
- Caras de Chernoff (Chernoff's Faces)
- Gráficos Estrella (Star Plots / Radar Charts)

Recuerde preprocesar los datos. En particular piense en normalizar valores, codificar las variables categóricas, y tener en cuenta valores vacíos o NaNs.