

Razonamiento (II)

Razonamiento Automático (GII)

Práctica 2

Machine Learning

Defensa de trabajos (en grupo): Miércoles, 21 de enero de 2015 a las 09:00h, aula L16.

1. Realizar una presentación, con duración máxima de **12 minutos**, explicando todos los detalles de la tecnología desarrollada.
2. La presentación debe incluir:
 1. Algoritmos de Machine Learning implementados
 2. Estado final de implementación
 3. Coste temporal de desarrollo de cada tecnología
 4. Diseño técnico de la implementación
 5. Decisiones de diseño tomadas y motivación
 6. Medidas de depuración tomadas para comprobar el funcionamiento de las tecnologías.
 7. Experimentos realizados y resultados obtenidos.
3. La nota máxima a obtener en la defensa de la tecnología será **1.25 puntos**.
4. La presentación deberá ser entregada **subiéndola** a la sección **Archivos de Cloud**. La fecha límite para la entrega de la presentación es la hora de comienzo de la defensa de trabajos.

Entrega Machine Learning (en grupo): Miércoles, 14 de enero de 2015 a las 23:59h.

Condiciones:

1. Cada grupo debe hacer una entrega de un fichero ***nombregrupo_MachineLearning.zip*** que contenga lo siguiente:
 1. **Código fuente** e instrucciones de compilación para **Debian/Ubuntu 64 bits**.
 2. Documento PDF con:
 1. Descripción de todas las tecnologías implementadas y estado operativo.
 2. Manual de utilización del software diseñado para poder utilizar las tecnologías (entrenamiento de bots y uso posterior de los mismos).
 3. Descripción de experimentos realizados para evaluar el funcionamiento de las tecnologías y resultados obtenidos
 4. Conclusiones obtenidas respecto al trabajo realizado.
2. La entrega se realizará subiendo el fichero a la sección archivos de **Cloud**.
3. La nota máxima a obtener por el apartado de Machine Learning será **3.00 puntos**. Cada grupo optará a un máximo total de nota en función de la tabla 3.
4. Los criterios para la evaluación son los siguientes:
 1. **[75%]** Estado de funcionamiento de los algoritmos implementados. Si funcionan correctamente en todos los casos, se obtiene el máximo. Si no funcionan correctamente, en función de los problemas que tengan se obtendrá una ponderación sobre el máximo, que quedará a criterio del profesor.
 2. **[25%]** Calidad técnica de la implementación. Se valorará que la implementación tenga un diseño flexible, que permita un fácil uso de los algoritmos desarrollados y sea fácilmente manejable ante cambios y extensiones. La correcta separación entre lógica y datos del software desarrollado será importante en la valoración.

Entrega BOTs Inversión (individual): Martes, 20 de enero de 2015 a las 23:59h.

Condiciones:

1. Con un tiempo de antelación mínimo de 6 días (14 de enero de 2015), los alumnos recibirán los datos históricos de un valor de mercado o de una cartera real (en adelante, El Stock), extraídos de fuentes de inversión fiables.
2. Los datos de inversión se darán en formato CSV, con la misma estructura que provee Google Finance, salvo por el tiempo, que vendrá expresado en [TIEMPO UNIX](#). Por tanto, cada línea del fichero contendrá los datos: unixtime (long int), open (double), high (double), low (double), close (double), volume (double).
3. Con los datos de inversión, los participantes deberán entrenar bots que sean capaces de ganar dinero comprando y vendiendo unidades del Stock.
4. Se entregará **código fuente** e instrucciones de compilación para **Debian/Ubuntu 64 bits**.
 1. La entrega se realizará subiendo a la sección archivos de **Cloud** un fichero **nombrecompletoalumno_bot.zip** que contenga el código fuente en un directorio y un fichero readme.txt con la siguiente información: *Nombre completo de autor, tecnologías utilizadas, instrucciones de compilación, otra información relevante*.
 2. Además, se puede entregar un ejecutable, compilado en 64 bits y con las librerías enlazadas estáticamente (o incorporadas en el pack utilizando LD_LIBRARY_PATH).
 3. Tras compilar, el fichero **ejecutable deberá llamarse** *username.robot*. username será el nombre de usuario del alumno en **cloud.i3a.ua.es**.
 4. Se debe incluir todo lo necesario para compilar, así como una lista de librerías necesarias del sistema, en su caso. Preferentemente, deben adjuntarse las librerías en una carpeta, en 64 bits.
5. Estará **prohibido** leer y escribir en ficheros en tiempo de ejecución. Toda la información que deba manejar un bot deberá estar embebida en el ejecutable en tiempo de compilación.
5. Los bots entregados deben hacer uso **obligatoriamente** de la tecnología desarrollada por el grupo.
6. Cómo **mínimo** debe haber 1 tecnología funcionando en el módulo común del grupo.
7. Criterios de evaluación:
 1. **[0.50 puntos]** Bot funcional, compilando y cumpliendo las condiciones de entrega.
 2. **[1.50 puntos]** La puntuación obtenida saldrá de esta fórmula (pos: posición final, numbots: total de bots compitiendo):
$$\text{puntos} = 1.50 * ((1 - (\text{pos} - 1)/\text{numbots}))$$

Campeonato de bots de inversión (individual): A realizar del 22 al 30 de enero

1. El campeonato de bots de inversión tendrá una única prueba: *All vs All*.
2. Todos los bots participantes participarán en una o varias simulaciones de mercado, utilizando el simulador [invSim64](#) facilitado en clase de teoría.
3. Las simulaciones de mercado harán que los bots reciban información “en tiempo real de simulación” sobre las variaciones del precio en el mercado simulado. En respuesta a esta información, los bots deberán enviar ordenes de compra o venta cuando crean oportuno.
4. Las simulaciones durarán un tiempo determinado. Al finalizar el tiempo de simulación, ganará el bot cuya suma entre su dinero disponible y el valor final de sus Stocks sea mayor.
5. Cada simulación será independiente de las anteriores: todos los bots comenzarán siempre con una misma cantidad predeterminada de dinero.
6. Finalmente, se sumarán los resultados de todas las simulaciones y ganará el bot que mayor cantidad de dinero haya conseguido obtener, como suma de su dinero y el valor sus Stocks finales entre todas las simulaciones.

Tabla 1: Dificultad de Modelos de Machine Learning

- Fáciles:
 - Perceptrón
 - Regresión Lineal
- Medios:
 - Regresión Lineal con transformaciones no lineales
 - Regresión Logística
 - Redes Neuronales
- Difíciles:
 - Redes Neuronales con Algoritmos Genéticos
 - Support Vector Machines
 - Radial Basis Functions con Kernel Methods

Tabla 2: Añadidos a los Modelos de Machine Learning

- Regularización
- Validación
- Preprocesado
 - Algoritmo de las K-medias (Clústering no-supervisado)
 - Mixture of Gaussians (EM)
 - Principal Component Analysis (PCA)

Tabla 3: Puntuaciones según grupos y algoritmos elegidos

- **Grupos de 3 o 4 personas:**
 - [**<= 2.00 puntos**] 2 Modelos fáciles + 2 Medios + 1 Añadido
 - [**<= 2.50 puntos**] 2 Modelos fáciles + 1 Medios + 1 Difícil + 1 Añadido
 - [**<= 3.00 puntos**] 1 Modelos fácil + 2 Medios + 1 Difícil + 2 Añadidos
- **Grupos de 5+ personas:**
 - [**<= 2.00 puntos**] 2 Modelos fáciles + 3 Medios + 2 Añadidos
 - [**<= 2.50 puntos**] 2 Modelos fáciles + 2 Medios + 1 Difícil + 2 Añadido
 - [**<= 3.00 puntos**] 2 Modelos fáciles + 2 Medios + 2 Difíciles + 3 Añadidos