Universidad Tecnológica Centroamericana
Facultad de Ingeniería

CC414 - Sistemas Inteligentes
Docente: Kenny Dávila, PhD

Tarea #5 (4% Puntos Oro)

Para completar esta tarea es requerido usar **Python 3** y la librería **Scikit-Learn**. También se recomienda el uso de las librerías **Numpy** para manejar los datos y Matplotlib para la creación de plots con los datos resultantes.

Parte 1. Regresión Lineal (1.0%)

En esta parte debe aplicar regresión lineal para predecir el Puntaje total (de 0 a 5 puntos) que un usuario le podría dar a una película que no ha visto. Para esto, se le da un dataset con una serie de atributos como ser: rating global, genero favorito, casting, advertising y longitud. El rating global se refiere a un puntaje que otros usuarios le han dado a la película (de 0 a 10 estrellas). El género favorito se refiere a si el usuario gusta de este género de películas (0=disgusto total, 1=genero favorito). El casting se refiere a que tanto le gustan los actores principales de esta película al usuario (0=no le gusta ninguno, 1=le gustan todos). El advertising se refiere a que tan buena y adecuada ha sido la publicidad hecha a la película (0=mala publicidad incluyendo trailers engañosos, 5 = buena publicidad muy atractiva). Finalmente, longitud es la duración de la película (entre 60 a 180 minutos).

Usando regresión lineal, usted deberá encontrar los diferentes pesos que se le pueden dar a cada atributo (**más el intercepto**) a fin de predecir el puntaje final otorgado por el usuario. Se le pide experimentar con dos tipos de regresión lineal: **mínimos cuadrados** y **Lasso con regularización L1**. Como lo vimos en clase, la regresión Lasso recibe un parámetro adicional que controla el **peso de la complejidad**. Por defecto, su valor es de 1.0, el cual es un valor **muy elevado** para este dataset. **Debe experimentar con diferentes valores decimales hasta encontrar uno que produzca resultados satisfactorios**, es decir, que su MSE de entrenamiento no sea tan alto en comparación con el de la regresión lineal original. Note también que el valor óptimo para este parámetro cambia con el rango de los datos, y tendrá que hacer múltiples pruebas tanto con los datos originales como con los datos normalizados.

Ambos métodos deben ser entrenados usando el dataset de entrenamiento ("regression_train.csv"), y su rendimiento deberá ser evaluado usando el dataset de pruebas ("regression_test.csv"). Deberá calcular métrica de evaluación promedio del error al cuadrado (MSE), y también debe imprimir los pesos que se obtiene para cada atributo y el bias en cada uno de los regresores. Deberá repetir estos pasos usando la **versión original de la data** y la **versión normalizada** de la data (usando la clase StandardScaler de Scikit-learn).

En su **reporte**, se le pide contestar las siguientes preguntas:

1. ¿Cuáles son los valores de MSE y pesos del modelo de regresión por mínimos cuadrados usando tanto los datos originales como también los normalizados? (reporte tanto en datos de entrenamiento como en datos de pruebas)
2. ¿Cuáles son los valores de MSE y pesos del modelo de regresión de Lasso usando tanto los datos originales como también los normalizados? (reporte tanto en datos de entrenamiento como en datos de pruebas)
3. ¿Qué valores Alpha utilizó en cada dataset para obtener los resultados reportados para Lasso?
4. ¿Cuál de los dos modelos funciona mejor a su criterio y por qué?
5. En base a datos no normalizados, ¿Cómo ordenaría la importancia de cada uno de los 5 atributos en la decisión final? Su respuesta debe estar **basada en datos** (no invente).
6. En base a datos normalizados, ¿Cómo ordenaría la importancia de cada uno de los 5 atributos en la decisión final? ¿hay cambios con respecto al punto anterior? ¿Por qué si o porque no? Su respuesta debe estar **basada en datos** (no invente).

Parte 2. Regresión Logística (1.0%)

El objetivo de este este ejercicio es utilizar la implementación de Regresión Logística provista en la librería Scikit-Learn.

Una compañía de seguros desea entender los criterios que usan sus clientes para escoger entre sus dos principales planes de seguro para carro. El primer plan es el “Básico” (Plan B) que es un plan bastante económico y se enfoca en cobertura mínima para accidentes y daños a terceros y no provee otros beneficios adicionales. El otro plan es el “Completo” (Plan C) que es considerablemente más caro (triple del costo de plan B) y aparte de la cobertura básica, también ofrece una serie de servicios al cliente como ser asistencia en carretera 24/7, incluyendo tareas simples como cambios de llantas, e incluso un servicio de carros rentados a precio descontado cuando el carro se encuentra en reparaciones, entre otros beneficios varios.

En base a esto, la compañía elabora un cuestionario de 10 preguntas considerando factores que ellos esperan que afecten la decisión de sus clientes. La idea es que, utilizando estas mismas 10 preguntas, se pueda determinar si una persona nueva escogería el plan B o el plan C. Los factores considerados son los siguientes:

1. **Conductor joven**. Indica si se trata de un(a) conductor(a) de menos de 40 años de edad.
2. **Historial de Accidentes**. Indica si se trata de un(a) conductor(a) que ya ha estado en accidentes (ya sea culpable o no).
3. **Alto millaje.** Determina si el carro que se asegurará tiene un millaje alto (más de 100K millas / 160 mil kilómetros)
4. **Carro viejo.** Indica si el carro que se asegurará se fabrico hace mas de 10 años.

5. **Carro rentado.** Determina si el/la cliente considera útil o importante el poder tener acceso a carros rentados a precio descontado en caso de que su carro asegurado se encuentre en reparación.
6. **Maneja mucho.** Indica si se trata de alguien que conduce al menos 90 minutos diarios en días laborales.
7. **Conductor experimentado.** Indica si la persona lleva al menos 10 años manejando.
8. **Cambiar llantas.** Indica si la persona es capaz de cambiar llantas punchadas sin asistencia.
9. **24_7.** Determina si la persona considera importante poder tener asistencia en carretera a cualquier hora del día, cualquier día de la semana.
10. **Dispuesto a pagar.** Indica si la persona se encuentra dispuesta a pagar más de lo usual por un buen seguro de automóvil.

Debe tomar los datos de clasificación binaria (Plan B vs. Plan C) y evaluar el rendimiento del clasificador basado en Regresión Logística sobre dichos datos. Nótese que será necesario que convierta los atributos binarios en números (0 y 1s) antes de poder usar la regresión logística sobre estos datos.

Se le pide usar los datos de entrenamiento ("seguros_training_data.csv") para entrenar un modelo de regresión logística. Deberá reportar el Accuracy, Recall, Precisión, F1-score (con clase positiva = "Plan C"), y tiempo total de predicción sobre los datos de prueba correspondientes ("seguros_testing_data.csv").

En su reporte deberá incluir lo siguiente:

1. ¿Considera que la regresión Logística funcionó bien para este problema?

2. Reporte los pesos aprendidos para cada atributo por la regresión logística. ¿Qué atributos tienen mayor peso absoluto y que atributos tienen menor peso absoluto?

3. Sorpresas. ¿Son los pesos aprendidos consistentes con lo que usted esperaba ver?

Parte 3. Random Forests (2.0%)

Se le pide utilizar bosques aleatorios (Random Forests) para clasificar películas por género. Los Random Forests cuentan con una serie de parámetros que controlan su efectividad. Su objetivo es tratar de maximizar el promedio del score F-1 por clase, y deberá experimentar con diferentes configuraciones de los parámetros de los árboles de decisión. Debe experimentar con diferentes valores para cada uno de los siguientes parámetros: "criterio", "número de árboles", "profundidad máxima" y "número de atributos". Se le pide probar al menos 15 configuraciones validas distintas de los parámetros del clasificador Random Forests.

Por cada configuración, debe probar el clasificador utilizando solamente los datos de entrenamiento ("genero_peliculas_training.csv"). Debe dividirlos en 5 sub-conjuntos para ejecutar 5 fold cross-validation. En este método, se entrenan 5 clasificadores distintos por configuración, usando 1 de los 5 sub-conjuntos

como datos de validación y los 4 restantes como datos de entrenamiento. Luego, se obtiene el promedio de los "F-1 por clase promedio" de cada uno de los 5 clasificadores de la configuración actual. Si se prueban 15 configuraciones en 5 data folds, se entrenarán un total de 15 x 5 = 75 clasificadores únicos. Como resultado de este ejercicio, se le pide reportar una tabla como sigue:

					F1-Promedio por Clase					
Conf. Id	**Param 1**	**Param 2**	**...**	**Param N**	**P1**	**P2**	**P3**	**P4**	**P5**	**Promedio**
1	val P1 1	val P2 1	...	val PN 1						
2	val P1 2	val P2 2	...	val PN 2						
...	...	...	...	...	...	...	...	...	...	...
N	val P1 N	val P2 N	...	val PN N						

Debe seleccionar la configuración que obtuvo el **mejor promedio** del **F-1 promedio por clase** sobre los datos de validación y reportar estadísticas más detalladas sobre esta configuración. Esto incluye la matriz de confusión y el accuracy total sobre los datos de validación (5 folds). Note que la validación se ejecuta en 5 partes, por lo que para obtener la matriz de confusión de los datos de validación usted deberá sumar las matrices individuales de cada una de las 5 partes. Luego, es válido calcular el accuracy total en base a esta matriz de confusión combinada.

Posteriormente, deberá entrenar un nuevo modelo de random forest usando el 100% de los datos de entrenamiento disponibles y la mejor configuración de parámetros encontrada en la prueba anterior. Luego, se le pide evaluar el clasificador con el dataset de pruebas. Debe reportar la matriz de confusión, el promedio de F-1 por clase, y la efectividad (accuracy) total sobre los datos de prueba.

En el **reporte** provea lo siguiente:

1. **Tabla** de **resumen de resultados** para todas las configuraciones sobre los datos de **entrenamiento/validación**.
2. Análisis detallado de los resultados de **validación** **para la mejor configuración**.
3. Análisis detallado de los resultados de **prueba** para el modelo entrenado usando **100%** de los datos con **la mejor configuración**.
4. Discusión detallada de los resultados. Debe analizar toda la información colectada hasta ahora sobre este problema y escribir un buen análisis de lo aprendido. Entre otras cosas, debe hacer **comparaciones con los clasificadores probados anteriormente sobre este mismo dataset**.

Parte 4. Reporte.

Debe elaborar un reporte siguiendo los lineamientos de tareas anteriores. En particular, debe proveer una introducción; resultados y respuestas a las preguntas de análisis de las partes 1, 2 y 3; un análisis resumido de lo aprendido a través de esta tarea; un resumen de dificultades entradas; y la conclusión. Note que los resultados y preguntas representan la mayor parte del puntaje de esta tarea.

Otras políticas

1. Esta tarea deberá trabajarse y entregarse **individual** o en **parejas.**
2. La entrega será **un solo archivo comprimido (.zip o .rar)**. Dentro de dicho archivo debe contener la guía completada **en formato PDF**, y también debe contener **los scripts de Python** que se usaron para contestar cada punto.
3. Si se hace en parejas, **ambas** personas **deben subir el mismo archivo**.
4. El **plagio** será penalizado de manera severa.
5. Los estudiantes que entreguen una tarea 100% original recibirán una nota parcial a pesar de errores existentes. En cambio, los estudiantes que presenten tareas que contenga material plagiado recibirán 0% automáticamente independientemente de la calidad.
6. Tareas entregadas después de la fecha indicada solamente podrán recibir la mitad de la calificación final. Por esta razón, es posible que **un trabajo incompleto pero entregado a tiempo termine recibiendo mejor calificación que uno completo entregado un minuto tarde**.