Lista de Ejercicios 1

Introducción a Machine Learning para CCSS

13 de octubre, 2024

Instrucciones

1. **Temas abordados:** Esta lista de ejercicios se enfoca en los siguientes temas: Gradient Descent, Trade off Bias Variance & Cross-Validation.
2. **Formación de grupos:** Se permite la formación de grupos de hasta 4 integrantes. La composición de los grupos se mantendrá constante para las listas de ejercicios 2 y 3.
3. **Puntuación de ejercicios:** La lista contiene 4 ejercicios. Cada ejercicio vale 5 puntos.
4. **Formato de entrega:** La resolución de los ejercicios debe presentarse en un archivo jupyter-notebook con todas las celdas ejecutadas.
5. **Fecha límite de entrega:** La fecha límite para la entrega es el Domingo 20 de octubre a las 11:59 pm. Un representante del equipo debe subir su solucionario a la actividad correspondiente en la plataforma Canvas. Los nombres y códigos de todos los participantes deben ser incluidos en el solucionario.

Pregunta 1

En este ejercicio, estimará los coeficientes de un modelo de regresión lineal utilizando el conjunto de datos Boston. Tenga en cuenta que la variable objetivo es 'crim', que representa la tasa de criminalidad per cápita por ciudad. Considere los siguientes pasos:

a) Ajuste un modelo de Mínimos Cuadrados Ordinarios (OLS) y estime los coeficientes mediante el atributo `coef_`. *Hint:* considere la biblioteca Scikit-Learn.

b) Estime manualmente los coeficientes OLS y compárelos con los resultados obtenidos en el paso anterior. Considere que los coeficientes se pueden estimar minimizando la Suma Residual de Cuadrados (RSS). *Hint:* la Suma Residual de Cuadrados se define como:

$$\text{RSS} = (y - X\beta)' (y - X\beta)$$

c) Estime los coeficientes OLS mediante el algoritmo de optimización de Descenso de Gradiente. Experimente con distintos valores para la Tasa de Aprendizaje (*Learning Rate*) y el Umbral de Convergencia (*Convergence Threshold*). Compare los coeficientes obtenidos con aquellos obtenidos en los pasos a) y b). ¿Qué valores de Tasa de Aprendizaje y Umbral de Convergencia proporcionan resultados más cercanos a los obtenidos anteriormente?

Pregunta 2

El objetivo de este ejercicio es ilustrar cómo el trade-off entre sesgo y varianza evoluciona con la creciente complejidad de los modelos polinómicos. Para este propósito, utilice el conjunto de datos Portafolio. Considere los siguientes pasos:

a) **División del conjunto de datos:** Separe los datos en un conjunto de entrenamiento y otro de prueba, asignando el 20% de los datos al conjunto de prueba.

b) **Ajuste de modelos polinómicos:** Implemente un bucle para ajustar modelos de regresión polinomial desde grado 1 hasta grado 8. *Hint:* considere las funciones `PolynomialFeatures` y `LinearRegression` de Scikit-Learn.

c) **Descomposición sesgo-varianza:** Dentro del mismo bucle, calcule la descomposición de sesgo-varianza para cada modelo, tanto en el conjunto de entrenamiento como en el de prueba. *Hint:* considere la librería mlxtend.

d) **Visualización:** Grafique cómo sesgo, varianza y error cuadrático medio (MSE) varían con la complejidad del modelo (grados polinómicos) para ambos conjuntos, entrenamiento y prueba.

Pregunta 3

Sobre k-fold Cross-Validation:

a) Explique cómo se implementa el enfoque k-fold Cross-Validation.

b) Detalle cuáles son las ventajas y desventajas del enfoque k-fold Cross-Validation con respecto a:

I. El enfoque del Conjunto de Validación.

II. El enfoque de Validación Cruzada Dejando Uno Afuera (LOOCV).

Pregunta 4

A continuación implementará Cross-Validation para un conjunto de datos simulado. Considere los siguientes pasos:

a) Genere el conjunto de datos simulado de la siguiente manera:

```
rng = np.random.default_rng(1)
x = rng.normal(size=100)
y = x - 2 * x^2 + rng.normal(size=100)
```

En este conjunto de datos, ¿cuál es n (número de observaciones) y cuál es p (número de variables predictoras)? Escriba el modelo utilizado para generar los datos en forma de ecuación.

b) Establezca una semilla aleatoria y luego calcule los errores LOOCV que resultan de ajustar los siguientes cuatro modelos polinomiales usando mínimos cuadrados:

i. $Y = \beta_0 + \beta_1 X + \varepsilon$

ii. $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \varepsilon$

iii. $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \varepsilon$

iv. $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \beta_4 X^4 + \varepsilon$

c) Repita el punto c) usando otra semilla aleatoria e informe sus resultados. ¿Son sus resultados iguales a los que obtuvo en c)? ¿Por qué?

d) ¿Cuál de los modelos en c) tuvo el error LOOCV más pequeño? ¿Esperaba ese resultado? Explique su respuesta.