Universidad del Valle de Guatemala
Facultad de Ingeniería
Inteligencia Artificial – Sección 10

Laboratorio 3

Task 1 – Teoría

Responda con criterio y análisis de ingenierx. No se esperan definiciones de libro, sino que analice las consecuencias de las decisiones de diseño.<

La Mentira de la Independencia (Naive Bayes)

En la diapositiva 6 se menciona que Naive Bayes "Asume independencia entre observaciones". Con esto en mente, considere que usted está construyendo un filtro de spam y su modelo analiza la frase "Cuenta Bancaria". En el mundo real, la probabilidad de que aparezca la palabra "Bancaria" aumenta drásticamente si ya apareció la palabra "Cuenta". Responda

- Si Naive Bayes trata estas dos palabras como eventos independientes (lanzar dos monedas separadas) en lugar de eventos dependientes, ¿está el modelo subestimando o sobreestimando la probabilidad real de la frase conjunta? Justifique su respuesta basándose en la fórmula $P(A \cap B)=P(A)\cdot P(B)$ vs. la realidad. ".

El modelo subestima la probabilidad real de la frase conjunta. Esto ocurre porque Naive Bayes trata a 'Cuenta' y 'Bancaria' como eventos independientes cuando en realidad son eventos dependientes. En la realidad, si aparece 'Cuenta', la probabilidad de que aparezca 'Bancaria' es mayor, por lo que $P(\text{Cuenta} \cap \text{Bancaria})$ es mayor que lo que la fórmula de independencia $P(A)\cdot P(B)$ sugiere. Al asumir independencia incorrectamente, Naive Bayes calcula un producto de probabilidades que es menor que la verdadera probabilidad conjunta observada en los datos, resultando en una subestimación.

La Economía de los Datos (SVM)

Refierase a las slides correspondientes al tema de SVM. Con esto en mente considere, usted entrena un SVM con 1 millón de datos de partidas de League of Legends. El modelo resultante identifica 5,000 "Vectores de Soporte". Su jefe le dice que para ahorrar espacio en la base de datos, va a eliminar los otros 995,000 puntos de datos que no son vectores de soporte y re-entrenar el modelo solo con los 5,000 restantes. Responda:

- Matemáticamente, ¿cambiará la frontera de decisión (el hiperplano) al hacer esto? ¿Por qué sí o por qué no?

  Matemáticamente no cambia, ya que, siempre se tiene que en este caso, la formula de predicción, solo depende de los SV, en este caos los otros puntos tienden a a=0, y como no contribuyen, entonces no cambiaría la frontera.

- Explique la eficiencia de memoria de este algoritmo frente a uno como KNN}

  Es mucho más eficiente comparándolo con un KNN, esto debido a que, empezando en el tema de memoria, solo guarda los 5,000 datos que le van a servir de frontera, mientras que el KNN guardaría todos y seria mas memoria ocupada, de igual manera al realizar la predicción el SVM, pasaría de manera más rápida por todos los datos, mientras que, un KNN pasaría por todos los datos y llevaría más tiempo.

La Miopía de los Árboles (Algoritmo Greedy)

En las diapositivas se menciona que la construcción del árbol es un "algoritmo greedy" (codicioso/avaro). Con esto en mente considere, Un algoritmo greedy toma la mejor decisión posible en el paso actual sin preocuparse por el futuro. Responda:

- Si el árbol elige el "Mejor Feature" para dividir el nodo raíz porque reduce la impureza drásticamente ahora, ¿garantiza esto que el árbol final será el más óptimo/pequeño posible?

  No garantiza que el árbol final sea el óptimo/pequeño posible. El algoritmo greedy toma la mejor decisión localmente en cada paso, pero no considera cómo esa decisión afectará las divisiones futuras. Es posible que elegir un feature 'sub-óptimo' al inicio permita divisiones más eficientes después, resultando en un árbol más pequeño y simple al final. El algoritmo greedy encuentra un óptimo local**,** no necesariamente el óptimo global**.**

- ¿Dibuje o describa un escenario lógico donde elegir una división “sub-óptima” al inicio podría llevar a un mejor árbol al fina Qué nombre técnico recibe este fenómeno?

  NO garantiza el árbol óptimo. Por ejemplo, al clasificar animales como peligrosos o seguros, el algoritmo greedy podría elegir dividir primero por 'Es_Carnívoro' porque reduce más la impureza inmediatamente, pero esto podría mezclar leones con gatos (ambos carnívoros) requiriendo múltiples divisiones posteriores. En cambio, dividir primero por 'Tamaño' (sub-óptimo inicialmente) podría separar mejor los grupos y resultar en un árbol más simple al final. Este fenómeno se llama optimización local vs. global: el greedy encuentra un óptimo local, no necesariamente el óptimo global.

Repositorio donde se encuentra el archivo de las demás tasks:

https://github.com/Emadlgg/uvg-ia-2026/tree/b010f56b8947f4ea9f4d28e5011f62d22b36e170/labs/lab3