PREGUNTAS TEORIA MACHINE LEARNING 3

1. Explica con tus palabras el significado del segundo argumento utilizado en el objeto de sklearn mostrado en el siguiente código de ejemplo:
   KMeans(n_clusters=k, init='random')

*En el código proporcionado, **KMeans(n_clusters=k, init='random')**, el segundo argumento **init='random'** especifica el método de inicialización de los centroides al comenzar el algoritmo de k-means. Este parámetro determina cómo se seleccionan las posiciones iniciales de los centroides antes de que el algoritmo comience a iterar para minimizar la varianza dentro de los clusters. Cuando se establece en **'random'**, los centroides iniciales se eligen al azar de entre los puntos de datos del conjunto de datos. Este método es uno de varios disponibles para la inicialización y puede afectar la convergencia y el resultado final del algoritmo de k-means.*

2. En el algoritmo de clustering aglomerativo (HAC), ¿qué objetivo principal tiene el enlace Ward (linkage='ward')?

El enlace Ward en el clustering aglomerativo busca minimizar la varianza dentro de los clústeres, fusionando los que incrementen menos esta suma de diferencias cuadráticas.

3. Dada la función de reconstrucción de PCA, ¿por qué queremos maximizar o elegir los autovalores más altos y qué hipótesis acerca de la media muestral hemos tenido que asumir?

Elegimos los autovalores más altos en PCA porque representan las direcciones de máxima varianza en los datos, maximizando la información retenida. Se asume que la media muestral de los datos es cero.

4. ¿Cuál es el rango de valores posibles para el coeficiente de silueta (Silhoutte score)? ¿Qué valor es el óptimo?¿Para qué tipo de algoritmo?

$$sc(i) = \frac{b_i - a_i}{\max(a_i, b_i)}$$

El coeficiente de silueta varía de -1 a 1, donde 1 es óptimo, indicando clústeres bien separados y cohesivos. Se utiliza en algoritmos de clustering como k-means o clustering jerárquico.

5. ¿Cuál es el propósito de calcular un perfil de log-likelihood en PCA? ¿Nos interesa el mínimo o el máximo en la gráfica?"

El propósito de calcular un perfil de log-likelihood en PCA es determinar el número de componentes principales a retener; nos interesa el punto donde el log-likelihood alcanza su máximo, indicando la mejor representación de los datos con menos componentes.

6. Dada la siguiente línea de código: **Z = scipy.cluster.hierarchy.linkage(cityDist, 'complete')** ¿Qué está almacenado en Z[i, 2]?

En Z[i, 2] se almacena la distancia entre los clústeres que se fusionaron en el paso i del algoritmo de clustering jerárquico utilizando el enlace completo.

7. Menciona un método relevante en el procesamiento del lenguaje natural (NLP) para reducir la dimensionalidad que pueda considerarse como alternativa al PCA.

Una alternativa relevante al PCA en el procesamiento del lenguaje natural para reducir la dimensionalidad es el análisis semántico latente (LSA, por sus siglas en inglés), que utiliza la descomposición en valores singulares (SVD) para identificar patrones en matrices de términos y documentos.

8. Identifica una variable fuertemente asociada a MntWines y otra que muestre escasa o ninguna asociación con MntWines.

feature_names	PC1	PC2
ID	-0.005422	-0.006155
Year_Birth	-0.090931	-0.625856
Income	0.410626	0.134237
Kidhome	-0.363594	-0.225476
Teenhome	-0.086597	0.661024
Recency	0.003690	0.032621
MntWines	0.405831	0.153844
MntFruits	0.394024	-0.166981
MntMeatProducts	0.447731	-0.146189
MntFishProducts	0.403682	-0.168723

En la imagen proporcionada, la variable "Income" está fuertemente asociada a "MntWines" en el componente principal 1 (PC1) con un valor de 0.40631. En contraste, la variable "ID" muestra escasa o ninguna asociación con "MntWines", dada su baja magnitud en ambos componentes principales, PC1 y PC2.

9. En la siguiente línea de código, ¿qué métrica debería seleccionarse para este objeto?

AgglomerativeClustering(linkage='ward', metric=_____)

En el objeto AgglomerativeClustering con linkage='ward', no es necesario especificar una métrica porque el enlace Ward sólo funciona con la métrica euclidiana, que es utilizada por defecto. Por lo tanto, el parámetro metric no debería incluirse o debería dejarse como 'euclidean' si se especifica.

10. Si disponemos de 750 imágenes de 8x8 píxeles y deseamos aplicar PCA para reducir a 5 dimensiones, ¿qué dimensión tendrá la matriz de loadings (W)?

*La matriz de loadings (W) tendrá una dimensión de 64 x 5, ya que cada imagen de 8x8 píxeles se aplana a un vector de 64 elementos y se reducirá a 5 componentes principales. Como XW = Z, la dimensión de W será de 64*5. como XW = Z, la dimensión de W será de 64*5.*

TEST II – RECOMMENDATION SYSTEMS

1. **En muchos sistemas de recomendación, especialmente aquellos que se basan en interacciones implícitas, se utiliza el concepto de "unary rating" para representar las interacciones de los usuarios con los items. Explica qué es un "unary rating" y cómo difiere de los sistemas de calificación explícita tradicionales**

   Respuesta:
   Un unary rating es un tipo de valoración en la que la única información que se registra es la interacción del usuario con un item, sin graduar el nivel de preferencia. Ejemplos como: Like, Visto, Comprado. A contrario de los datos detallados y los grados de preferencia con una escala que presentan los sistemas de calificación tradicionales.

2. **Explica cada uno de los términos de esta fórmula, donde la variable de la izquierda es el predicted rating.**

$$\boldsymbol{y_{ui} = U\Sigma V^T + \mu + b_u + c_i}$$

   Respuesta:
   Esta fórmula representa la reconstrucción de la matriz de valoraciones mediante el uso de Singular Value Decomposition. SVD descompone la matriz original en tres matrices, una rotación (Matriz ortogonal – U) seguido por un estiramiento (Matriz Diagonal - Σ) seguido por una rotación (Matriz ortogonal - V^T). Esto se representa en las primeras matrices.

   Los siguientes términos son el sesgo global de la matriz representado por μ, el sesgo de los ratings del usuario u - (b_u) y el sesgo de los ratings del item i - (c_i)

3. **Explica la diferencia entre Weighted Matrix Factorisation y Matrix Factorisation, detallando como se tratan los ratings no-observados (NAN) en cada uno de los modelos.**

   Respuesta:
   La diferencia entre Weighted Matrix Factorisation y Matrix Factorisation es el hecho que WMF intenta hacer predicciones sobre datos no observados (NaN), mientras que Matrix Factorisation suele descartar datos no observados por completo.

   WMF considera los datos no observados asignándoles un coeficiente tal que:

$$c_{ij} = \begin{cases} a \; si \; rating_{ij} > 0 \\ b \; if \; rating_{ij} = 0 \end{cases}$$

   Donde a > b, de esta manera se aprovechan mejor los datos disponibles y hay una reducción de sesgo de muestreo.

4. **¿Cómo afecta el Long Tail 4problem a la diversidad y precisión en sistemas de recomendación y qué método de filtrado colaborativo memory-based ayuda a mitigarlo?**

   Respuesta:
   El Long Tail problema se refiere a la distribución de popularidad de los ítems donde una pequeña cantidad tienen una alta frecuencia de interacciones, mientras que la gran cantidad de ítems (el 'long-tail') tiene pocas interacciones. En términos de **diversidad**, esto hará que se favorezcan los ítems populares, reduciendo la gama de productos y para la **precisión,** si la mayoría de los elementos tienen menos datos de interacción, es más difícil para los algoritmos predecir con precisión la relevancia de estos ítems.

   En memory-based tenemos item-based o user-based collaborative filtering, para resolver el problema, nos será más útil enfocarnos en los usuarios, que en los items, ya que la gama de productos que alcancen será más amplia.

5. **Disponemos de un conjunto de datos que consta de 5000 registros distribuidos en 3 columnas, correspondientes a 'userId','itemId', y 'rating'. Se identifican 4978 usuarios únicos y 1340 ítems únicos en el conjunto. Al proceder con la construcción de la matriz de ratings y aplicar la Descomposición en Valores Singulares (SVD), ¿cuáles son las dimensiones de la matriz V en la siguiente fórmula, donde M es la matriz de ratings? ¿Qué tipo de matriz es la matriz Sigma?**

$$M = U\Sigma V^T$$

   - $U = (m \times k)$ → Donde m es el número de usuarios.
   - $\Sigma = (k \times k)$ → Donde k es el latent space.
   - $V^T = (k \times n)$ → Donde n es el número de ítems.

   M = 4978
   N = 1340

   Al no haberse definido una dimensión del latent space, asumimos que se acepta toda la matriz de usuarios y ítems, por lo tanto U tendrá dimensión m x m, Sigma m x n, y V^T n x n.

   Por lo tanto, la matriz V tendrá dimensión (1340, 1340) y la matriz Σ es una matriz diagonal.

6. **Compara los sistemas de recomendación de tipo model-based frente a memory-based explicando como cada uno gestiona las recomendaciones para nuevos usuarios (cold-start problem). Explica de qué manera utiliza cada método la información del usuario y del item para generar recomendaciones.**

   Respuesta:
   Los memory-based utilizan técnicas de filtrado colaborativo que utilizan las calificaciones o interacciones previas de los usuarios para hacer recomendaciones. Estos sistemas pueden ser user-based o item-based. *Estos modelos suelen tener dificultades con los nuevos usuarios, ya que dependen completamente de los datos de interacción previa.* Sin datos, el sistema no tiene base para formar una recomendación.

   Los model-based utilizan algoritmos de optimización para aprender y modelar las preferencias de usuarios y características de los ítems a partir de datos existentes como Matrix Factorisation, BPR, Factorisation Machines. Manejan mejor el problema del cold-start, ya que extraen características de los usuarios o ítems en los vectores latentes para hacer predicciones.

7. ***Discute los pros y los contras de utilizar la correlación de Pearson frente a la similitud del coseno para calcular la similitud entre usuarios.*** **Considera en tu respuesta aspecto como la normalización de los datos y la sensibilidad de los datos dispersos.**

   Explica que diferencia hay entre el método 'Cosine' y el 'AdjustedCosine' para el cálculo de la similitud.

   Respuesta:
   La correlación de Pearson esta normalizada, lo que significa que considera tanto la media como la desviación estándar de las clasificaciones. Esto permite compararlas en una escala común y es útil cuando los usuarios tienen distintos estilos de clasificación. Lo malo es que requiere mucho más computo para llevar al cabo.

   La similitud del coseno, calcula directamente la similitud entre dos vectores de usuario considerando únicamente los ángulos entre ellos. Con cosine similarity, los datos no se normalizan, por lo tanto una diferencia de escala le puede afectar de manera negativa. Lo bueno es que lleva mucho menos tiempo de ejecución.

   Cosine vs Adjusted Cosine: Adjusted cosine considera el rating medio del usuario y se lo substrae previo a hacer el cosine similarity, el cual ayuda para mitigar sesgos.

8. **Proporciona ejemplos de dos tipos distintos de sistemas de recomendación, explicando brevemente cómo opera cada uno y en que contexto se suele utilizar.**

Respuesta:
Sistemas de recomendación explícitos:
Estos sistemas utilizan una matriz de calificaciones explícitas entre usuarios e ítems para generar recomendaciones. Operan mediante la identificación de similitudes entre usuarios (user-based) o ítems (item-based), basándose en las calificaciones proporcionadas directamente por los usuarios. Comúnmente se emplean en plataformas donde los usuarios pueden calificar productos o servicios, como sitios de reseñas de películas o tiendas en línea.

Sistemas de recomendación implícitos:
Estos sistemas se basan en las interacciones pasivas de los usuarios, como visualizaciones de páginas o compras, para formar una matriz de interacciones usuario-ítem. Utilizan la presencia o ausencia de interacciones para asignar un valor relativo a los ítems. A partir de esta matriz, se aplican algoritmos de aprendizaje para inferir las preferencias de los usuarios y generar recomendaciones. Son típicos en servicios de streaming de música o vídeo, donde las acciones de los usuarios proporcionan indicaciones implícitas de sus preferencias.

9. **Al aplicar algoritmos de descomposición matricial, como la Descomposición en Valores Singulares (SVD), a menudo nos encontramos con errores de convergencia en las bibliotecas que implementan estos algoritmos. Una causa común de estos errores es la alta proporción de elementos nulos (sparsity) en la matriz de ratings. Esto afecta la convexidad de la función objetivo, haciendo que el mínimo de la función no sea único (no hay una sola solución). Explica con tus palabras por qué no hay solución única cuando esto sucede.**

Respuesta:
Tener tantos elementos vacíos complica el proceso de minimación de la función de error, con multiplicación de matrices como es el caso para SVD, observamos el hecho de que existen muchas posibles soluciones y combinaciones de elementos que nos pueden llegar a minimizar nuestra función de error. Es por ello por lo que para resolver estos problemas presentes típicamente en Model-Based Collaborative filtering debemos de aplicar técnicas como Stochastic Gradient Descent, o Alternating Least Squares – que ajustan los factores de la matriz para optimizar la función de error, navegando por múltiples mínimos locales.

10. **¿Qué distribución se emplea habitualmente como prior para los parámetros de modelos en el Bayesiana Personalized Ranking dentro de los sistemas de recomendación implícitos, y cuáles son las razones que hacen de esta una opción razonable?**

Respuesta:
La distribución normal con media 0 y varianza independiente para cada dimensión del parámetro.
Esto ayuda con:
- Regularización: Previene overfitting empujando los parámetros a 0, lo cual es útil para datos dispersos
- Simplicidad Computacional
- Robustez: La normalidad proporciona un equilibrio entre flexibilidad y control sobre la complejidad del modelo.