Apuntes De Clase Semana 5 Inteligencia Artificial

Manuel Alejandro Rodríguez Murillo

I. Introducción

En esta clase se revisaron conceptos clave sobre entrenamiento de modelos de machine learning, incluyendo los conjuntos de entrenamiento, validación y prueba, así como técnicas para evitar el overfitting y mejorar la capacidad de generalización. Finalmente, se exploró la regresión logística y su relación con la regresión lineal para la clasificación binaria.

II. Noticias

Se menciona sobre la conversación de dos inteligencias artificiales que al momento de percatarse de que ambas son IA deciden cambiar la forma de comunicarse a una más eficiente para ellas.

A. Transformers without Normalization Paper

Las Transformers para las LLM que tenemos hoy en día lo que hacen es que normalizan las salidas de cada una de las capas. Es una técnica que se aplica desde hace mucho y funciona bien, pero es cara por lo que ellos se ponen a investigar y se percatan que cada salida de las capas donde ellos están trabajando tienen forma de S.

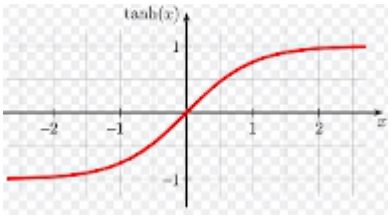

Figura 1: Tanh

Entonces proponen que se cambie la parte de normaliation para aprovechar esa forma de S, incorporando Dynamic Tanh. Outputs que están teniendo:

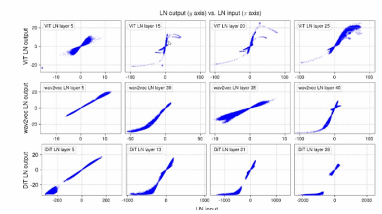

Figura 2: Salidas

B. Boston Dynamics

Boston Dynamics — Atlas utilizó motion para la captura de movimiento de las personas y uso estos datos para enseñarle a un robot utilizando aprendizaje por reforzamiento.

C. Pytorch

Están pensando en mover Pytorch a Linux fundation a lo que meta respondió que reduciría la contribución que hacían hacia esa plataforma, pero ahora estaría abierta a toda la comunidad por lo que otras empresas podrían querer contribuir.

III. Repaso

A. Training Set

Cuando entrenamos nuestro modelo y esperamos mejorar los parámetros a partir de los datos observados y la idea es que el modela sea capaz de predecir datos que jamás ha visto de una forma correcta. Básicamente lo que se hace es hallar patrones entre las entradas y salidas.
Esta tarea se debe realizar de manera tal que no provoque overfitting, que no se aprenda de memoria lo que esta en el set de entrenamiento, sino que sea capaz de generalizar dicha información. Overfitting: La incapacidad de generalizar adecuadamente las observaciones nuevas. Testing Set: Es el set que se va utilizar para probar el modelo.
• No se debe entrenar con este.
• Es independiente.
• Se calculan las métricas para saber si mi modelo esta correcto o no.
Caso Overfitting: La idea es tratar de evitar el overfitting de una temprana forma. Para esto se puede agregar un nuevo set de datos.

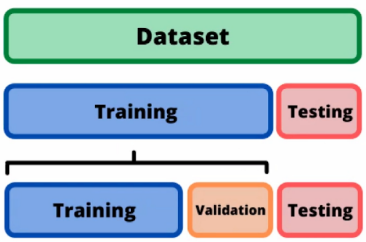

Figura 3: Validación

Permite monitorear el progreso que estamos teniendo durante el entrenamiento. Igual nunca se debe entrenar el modelo con este set. Es parecido al test, solo que lo vamos a ejecutar durante el entrenamiento para detectar el overfitting de una manera temprana. Validation Set:
• Permite valorar la capacidad de generalización de mi modelo a datos no vistos.
• Permite tomar decisiones del proceso de entrenamiento.
• Esencial para el ajuste de hiperparámetros.

B. Técnicas para subdividir el set de validación

• Random Sampling

Se utiliza siempre que el dataset este balanceado, que tenga

un mismo porcentaje. Esto permite que no se agregue ningún tipo de sesgo al seleccionar algún sample para los sub-sets.
• Stratified Sampling
Cuando los datos están imbalanceados se divide de misma manera los datos para el training como para el testing y validation por ejemplo 70-30 en cada uno de estos, permitiendo que por lo menos tengamos un par de ejemplos de cada una de las clases en nuestros subconjuntos.
• K-Fold Cross-Validation
Se toma el set de entrenamiento, se divide en k partes y se utiliza k-1 partes para el entrenamiento, la otra parte sobrante se utiliza para hacer la validación y este proceso se va a continuar, pero rotando los k que se utilizan para la validación.

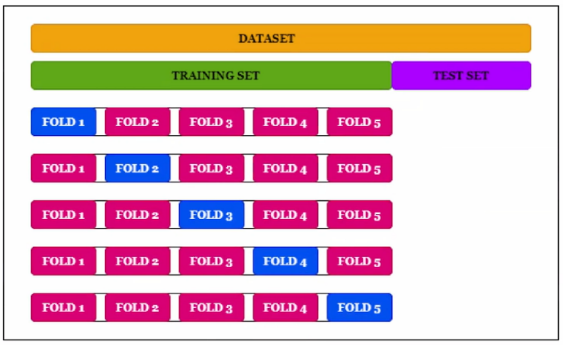

Figura 4: K-Fold

C. Posibles Escenarios

Bajo error en training, bajo error en testing
• Es el escenario ideal.
• Evita el ruido existente de los datos.
• Puede generalizar correctamente.
Bajo error en training, alto error en testing
• Overfitting
• No es capaz de generalizar
• Alta varianza
Alto error en training, alto error en testing
• Uderfitting
• El modelo no esta aprendiendo de los datos.
• Modelo muy simple.
• Alto sesgo.
Bias Variance tradeoff:
• El modelo optimo es aquel que tenga baja varianza y bajo sesgo.

D. Bias Variance tradeoff

Alto Bias
• Comete muchos errores en el training (Underfitting).
• El modelo asume mucho del training set.
• No usa todos los features de modelo.
• El modelo es simple.
Como evitarlo
o Utilizar un modelo más complejo
o Los features del training set no son los adecuados para el problema (No
tienen capacidades predictivas)

Alta Varianza
• El modelo se adapta mucho a los datos de entrenamiento (overfitting)
• No es capaz de generalizar
• Suele pasar con datos con muchas dimensiones y pocos ejemplos.
Que hacer
o Intentar con un modelo más simple
o Reducir dimensionalidad
o Obtener más ejemplos para el trainingset
o Aplicar técnicas de regulación.

IV. Regresión Logística

Regresión Logística ¡= Regresión Lineal
La regresión lineal modela una predicción de valores continuos dependiendo de los features que yo tenga.
La regresión logística va a dar una salida binaria, hace un modelo de clasificación binario donde explica si un evento sucede o no sucede. Da la probabilidad de que ocurra o no ese evento que estoy tratando de observar.

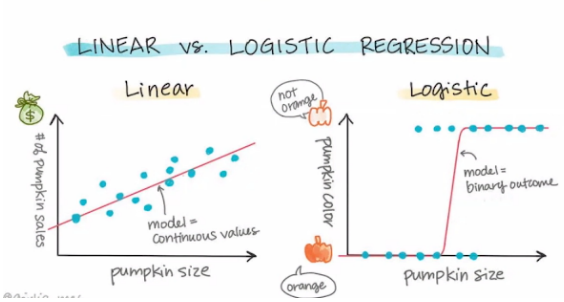

Figura 5: Linear vs Logistic

• Es un evento de clasificación binaria.
• Predice la probabilidad de ocurrencia de un evento binario.
• Distribución de Bernoulli.

$$P(X = k) = p^k \cdot (1 - p)^{1-k}$$

Figura 6: Logistic

Donde k es el valor que puede tomar (0 o 1) y p la probabilidad de que ocurra el evento

V. Sigmoid (Estándar Logisitc Function)

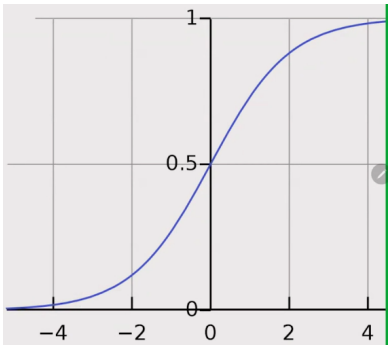

Figura 7: Forma Logística

• Tiene comportamiento no lineal
• Codominio [0,1]

$$\sigma(x) = \frac{1}{1+e^{-x}}$$

Figura 8: Formula Logística

Note que x puede ser cualquier número. Incluso el resultado de otra función. Para combinarlo con regresión lineal a la salida de esta se le coloca una función Sigmoid esto vuelve esa regresión lineal se vuelve un problema de clasificación.
Con esto estamos logrando mantener la combinación lineal entre pesos y features pero se aplica ahora para problemas de clasificación.

• $\sigma(x) = \frac{1}{1+e^{-x}}$
Resultado no lineal.
• $f_{w,b}(x) = wx + b$
Resultado lineal
• $\sigma(f_{w,b}(x)) = \frac{1}{1+e^{-(f_{w,b}(x))}}$

Figura 9: Formula Logística-Lineal

El resultado final obtenido es no lineal.

A. ¿Por qué queremos hacerlo así?

• Calcular una función lineal es super simple (computacionalmente).
• Es un método simple para mantener la relación entre variables y pesos.
• Obtener comportamientos no lineales con una función sencilla como Sigmoid.
• Permite modelar problemas con complejidad mayor.

B. Clasificador

• Para realizar el clasificador debemos definir un umbral.
• Y ¿= 0.5 = 1
• Y ¡ 0.5 = 0
• El umbral es cambiable (0.6, 0.7 . . .) Esto varía dependiendo del problema.
• Ejemplo una moneda cargada, donde esta no tiene la misma probabilidad de que salgan ambos escenarios por lo que se puede subir o bajar el valor del umbral.
Básicamente lo que hay que revisar es la salida y dependiendo del valor se clasifica según el lado del umbral donde se encuentre.
Entre más cerca este del umbral más probable es que se presenten errores, porque puede que perteneciera a una clase, pero fue tomado como la otra.

VI. Regresión Logística y Regresión Lineal

$$f_{w,b}(x) = \frac{1}{1+e^{-(wx+b)}}$$

Figura 10: Formula Logística-Lineal 2

• Resultado es no lineal
• [0,1]
• La relación de los features y pesos se da por la regresión lineal
• Probabilidad de que un evento suceda

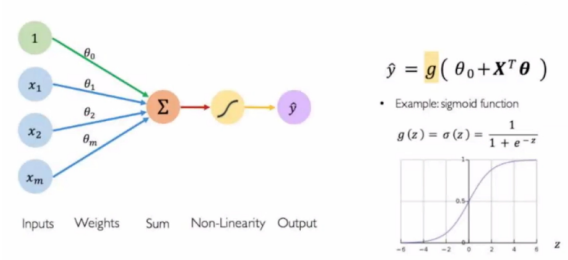

Figura 11:Logística-Lineal

A. Regresión Lineal:

Features: Ciruclos azules y cada uno posee un peso theta.
Bias: theta en la parte verde.
Sumatoria: Producto punto entre XW

B. No-Linealidad:

El resultado de lo anterior pasa por la función Sigmoid [0,1]

Y obtendríamos como resultado el $\hat{y}$

VII. Optimización

Se necesita optimizar los pesos de w y b de la regresión logística.

$$f_{w,b}(x) = \frac{1}{1+e^{-(wx+b)}}$$

Figura 12: Formula Logística-Lineal 3

Pasos:
• Hallar el lost function.
• Hallar las derivadas parciales de lost function con w y b, para utilizar el decenso del gradiante.
• Actualizar los parámetros w y b una vez calculadas las derivadas parciales respecto a lost function.
Debido a que estamos utilizando probabilidades no se puede utilizar la misma función de perdida que se vio anteriormente por lo que se debe utilizar otra.

VIII. Derivada Sigmoid

Pasos:
• Se deriva de manera normal
• Se agrega un 0 (1-1)
• Se separa la fracción
• Se simplifica
• Se saca factor común
• Y se nota que algunos de los datos son la propia sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$
$$\sigma'(x) = \frac{1' \cdot (1+e^{-x}) - (1 \cdot (1+e^{-x})')}{(1+e^{-x})^2}$$
$$\sigma'(x) = \frac{0 - 1 \cdot (1' + (e^{-x})')}{(1+e^{-x})^2}$$
$$\sigma'(x) = \frac{-(0-(e^{-x})}{(1+e^{-x})^2}$$
$$\sigma'(x) = \frac{e^{-x}}{(1+e^{-x})^2}$$

Figura 13: Derivada

$$\sigma'(x) = \frac{e^{-x}}{(1+e^{-x})^2}$$
$$\sigma'(x) = \frac{e^{-x}+1-1}{(1+e^{-x})^2}$$
$$\sigma'(x) = \frac{e^{-x}+1}{(1+e^{-x})^2} - \frac{1}{(1+e^{-x})^2}$$
$$\sigma'(x) = \frac{1}{(1+e^{-x})} - \frac{1}{(1+e^{-x})^2}$$
$$\sigma'(x) = \frac{1}{(1+e^{-x})} \cdot (1 - \frac{1}{(1+e^{-x})}) \text{ Factor común}$$
$$\sigma'(x) = \sigma(x) \cdot (1 - \sigma(x))$$

Figura 14: Derivada 2

IX. Verosimilitud vs MSE

- No usamos MSE.
- Necesitamos una función de costo relacionada a probabilidades.
- Verosimilitud.

$$L(\theta|X) = P(X|\theta)$$

Figura 15: Fromula 1

- Probabilidad condicional de observar X dado parámetro theta.
- La probabilidad de observar todos los datos.

$$L(\theta|X) = P(x_1, x_2, .., x_n|\theta) = P(x_1|\theta) \cdot P(x_2|\theta), \ldots, P(x_n|\theta)$$

Figura 16: Fromula 2

Verosimilitud es la probabilidad de observar los datos.

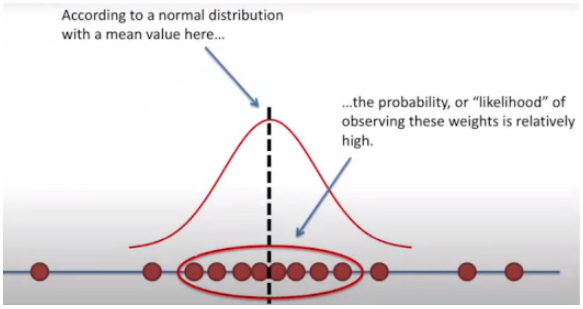

Figura 17: Verosimilitud 1

Conforme yo cambie mis parámetros logro que mi función avance, esto permite que si hago las suficientes pruebas pueda localizar el mejor lugar para observar los datos.

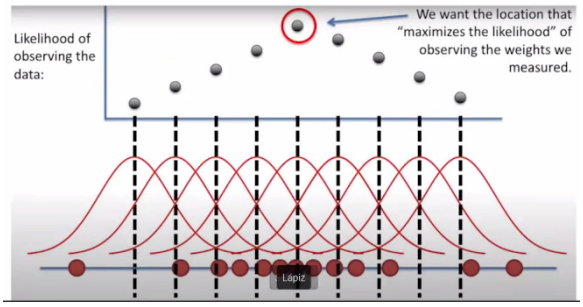

Figura 18: Verosimilitud 2

X. MSE vs Maximize Likelihood

- Regresión lineal minimizamos el MSE.
- En su lugar, vamos a maximizar la verosimilitud.
- Optimizar los parámetros w y b, para obtener la máxima verosimilitud.

XI. Concluciones

Se destacaron los principales desafíos en el entrenamiento de modelos de machine learning, como el equilibrio entre sesgo y varianza, y la importancia de una correcta subdivisión de los datos. Además, se profundizó en la regresión logística, explicando su optimización mediante funciones de pérdida adecuadas y la maximización de verosimilitud en lugar del MSE.