

## Nombre y Apellido:

Nicolás Olate Orellana

## Parte 1: Respuestas Preguntas de Lectura

1. Como se analizó en la lectura, podemos evidenciar este caso distorsiones de tipo '*The need to explore*'. Para realizar predicciones en machine learning, es necesario entrenar los datos previos, ya que siempre dependen de las acciones que ha tomado el algoritmo anteriormente. Es decir, que para aprender de manera efectiva es necesario explorar cada escenario, donde muchas veces se toman decisiones no óptimas para la búsqueda de mas datos y parámetros.  
Esto es lo que ocurre en el ejemplo dado, ya que, para generar una herramienta de machine learning que permita predecir si un paciente debe recibir tratamiento médico o no, se necesitan generar mas datos, y para realizar esto deben realizar una serie de exámenes, siendo algunos invasivos. Esto nos hace plantearnos si es que realmente es necesario tales costos individuales con el fin de explorar o es inmoral considerar acciones que creemos que son subóptimas. Siendo un problema a tratar.
2. El artículo menciona dos definiciones de fairness: Statistical definition of fairness e Individual definitions of fairness. Statistical definition of fairness es en la cual se basan la mayor parte de la literatura, esta trata (mediante estadística) simplemente en que de una manera general fijan a los grupos protegidos (por ejemplo, por raza) y se crea una paridad aproximada mediante una medida estadística en todos estos tipos de grupos. Esta definición es atractiva, al ser fácil de obtener y es lograda sin hacer suposiciones sobre los datos, verificándose fácilmente. El problema es que no brinda garantías significativas a los grupos protegidos en sí, sino que estas garantías son en promedio (no sabemos que pasa en cada caso si no en promedio). Por ejemplo, como vimos en clases existía una herramienta de machine learning que creaba un sistema de reincidencia que se equivocaba más con gente de raza negra que con blancos debido al algoritmo, si bien se aplicará esta estadística podríamos saber que en promedio se arreglaría este problema con este grupo protegido, pero aún así en casos específicos podrían existir injusticias.  
Por otro lado, Individual definitions of fairness trata en pedir restricciones en lo que se unan a pares específicos de individuos, más que a una cantidad en que se promedian los grupos. En el cual utilizando una métrica específica los individuos que sean similares serán tratados de manera similar. Esta definición resulta más

atractiva debido a que no toma los datos en promedio, si no que es eficaz con las minorías. El principal problema que enfrenta esta definición es el hecho de que para que funcione es necesario hacer suposiciones importantes. Volviendo al ejemplo anterior, en lugar de aplicar una estadística en promedio, podría verlo aplicando una métrica a pares cosa que no queden datos sesgados, pero deberíamos suponer o asumir distintas características de los datos o constantes que influenciaría el aprendizaje posterior.

3. En el artículo se mencionan distintos papers como el de Kears y Hebert-Johnson ya que estos piden que la Statistical definition of fairness no solo estén en grupos pequeños protegidos (como lo indicaba la definición), si no que sea de una clase exponencial o infinita de grupos definidos por alguna clase de funciones de complejidad limitada. Este enfoque es asertivo y prometedor ya que se están pidiendo nociones estadísticas de fairness, las cuales no requieren de hacer suposiciones sobre los datos ni necesitar de algún conocimiento externo. Además, retiene de manera correcta el concepto de *interseccionalidad*.

La interseccionalidad es un término que se usa para describir como los distintos tipos de discriminación pueden agravarse e interactuar para individuos de clases protegidas. La *interseccionalidad* es importante ya que es un enfoque en que se relacionan distintas categorías, por lo que nos deja ciertas dudas y dificultades. Ya que haciendo *fair* la herramienta de machine learning, directamente se verían afectadas otras categorías que se interrelacionan, generando resultados que afectarían a lo que realmente es. Por ejemplo, como señala el artículo, al relacionar estilos de ropa con raza o género.

## Parte 2: Respuestas Preguntas de Lectura

1. *Fair representation learning* es un proceso de desvanecimiento de datos que producen transformaciones de los datos originales que contienen la mayor cantidad posible de información relevante para predecir, en que anteriormente eliminan la información sobre atributos sensibles o protegidos. Esto significa que se busca la transformación de datos sesgados en que para saber si pertenece o no a un grupo. Esto puede inferirse a partir de sus otras características en que los atributos protegidos son independientes estadísticamente.

Este tema se ha abordado por distintos enfoques (distintos autores según señala el artículo). Dentro de estos nuevos enfoques se señalan ideas como: procedimientos para preservar el rango para reparar características y reducir o eliminar la dependencia entre atributos protegidos. También, se presenta un enfoque basado en la probabilidad que puede manejar atributos protegidos continuos, características discretas promoviendo independencia de las características transformadas y atributos protegidos.

Finalmente, podemos interpretar que sigue siendo un gran desafío la búsqueda de representaciones justas ya que, queramos o no, estos atributos protegidos influyen en otros. Por lo que, al usar este proceso de desvanecimiento, estas transformaciones implican una pérdida de información. Aún así es prometedor ver que existen nuevos enfoques o nuevas prácticas para las cuales quitando o reduciendo estos sesgos (atributos protegidos) se obtenga un menor costo en el rendimiento del algoritmo.

2. Por lo que entiendo tras leer el artículo es algo parecido a lo que hizo en Chile con la 'Discriminación positiva' para la ley de cuotas de género. Esto trata, generalizando, en aplicar ciertas medidas con el fin de aumentar la participación de grupos minoritarios por distintas categorías. Esto, es algo que ha estado en alza los últimos años, por ejemplo, con el género. En que, como nombré anteriormente la ley de cuotas de género en que un porcentaje arriba del 40% de los candidatos por partido político debían ser mujeres. También, he sabido que en la Universidad de Chile se ha aplicado esta lógica para carreras como Ingeniería dando ciertos cupos solo destinados a mujeres.

En mi opinión, siento que si bien debido a la cultura en que hemos vivido en que en un principio la mujer era la que debía hacer las cosas del hogar, cuidar a los niños, no iban a la universidad ni tenían los mismos privilegios que el hombre. Creo que estos procesos evolucionan según la cultura y la historia y no deben ser

forzados a darse. Hoy, en el 2021 sin obligar a votar por mujeres hemos visto en que se votan por igual a los hombres y no ha sido por las cuotas de género. De hecho, para las elecciones de constituyentes ocurrieron casos de que las ternas era de mayoría mujeres y debieron sacar a algunas para incluir a hombres, siendo tal vez mas calificadas las mujeres en ese cargo y viceversa. Por lo tanto, no considero que este correcto la reserva de cupos para mujeres ya que para entrar a la universidad nos medimos bajo el mismo sistema, que bueno o malo es el sistema que existe, por lo que al preferir a una mujer por un hombre que tuvo un puntaje menor para ingresar a la universidad provoca una injusticia en la persona que, por ser hombre, no obtendrá el ingreso aún siendo mas calificado que la que ingresó por el hecho de ser mujer. Esto llevándolo al artículo podemos ver que no se consideran los sistemas sociales. Para ver la causa de un cambio en el algoritmo no se ven estas nociones de justicia. Afectando también en las predicciones.

3. Si fuera fiscalizador, para verificar si efectivamente el banco discrimina a la hora de otorgar créditos lo primero que haría sería verificar si los datos disponibles que tiene el banco se encuentran contaminados con sesgos. Ya que estos datos podrían haber sido producto de procesos sociales e históricos que operaron en desventajas de otros grupos. Por ejemplo, género o raza, desfavoreciéndolos solo debido a los datos que tienen. Esto lo haría enseguida ya que es uno de los problemas de *fairness* que más aparecen, ya que, al entrenar estos datos sesgados, pueden reproducir o acrecentar los rasgos. Al analizar esto obligaría a el banco a aplicar alguna definición de *fairness* para corregir este problema en que se observará directamente reducciones en la predicción, pero con el fin de generar un sistema no discriminatorio sería oportuno de realizar.

## Parte 3: Respuestas Preguntas de Materia

1. Como vimos en clases un proceso de negocio se define como un conjunto de actividades lógicamente relacionadas que usan los recursos de la organización para proveer resultados definidos con el fin de alcanzar objetivos del negocio. Bajo esta definición no creo que todo software en una organización equivale a un proceso de negocio, esto debido a que existen distintos tipos de software. Por ejemplo, un software de sistema como Windows que no comprende completamente la definición de proceso de negocio ya que este tipo de software si bien realiza una serie de tareas dadas por el usuario, no busca llegar en sí a un objetivo concreto, sino que coordina el hardware con la interfaz del usuario. Mientras que software de otro tipo, como de aplicaciones cumplen efectivamente con la definición ya que utilizando los datos y ejecutando las instrucciones brindadas por el usuario realiza una serie de tareas pedidas, obteniendo los objetivos pedidos. Por ejemplo, el software Excel efectivamente se ajusta a la definición dada, ya que con los datos dados que tenga puedo pedirle tareas específicas.
2. A grandes rasgos la gran diferencia entre Deep Learning y las técnicas tradicionales de aprendizaje automático con relación a la ingeniería de características de los datos de entrada radica en que el Deep learning en lugar de tomar una información o una data tal como es (que la mayoría son millones de posibilidades de datos), se aprenden representaciones de los datos, resumiéndolo en un vector que tenga menos valores. Esto mediante, el aprendizaje de representaciones que puede ser tanto con matrices como con 'redes neuronales'. Mientras que en las técnicas tradicionales se toman los datos tal cual cualquiera sean sus dimensiones.  
Podemos ver un ejemplo concreto de esto al generar una herramienta de machine learning que identifique fotos de perros y gatos. Mediante Deep learning lo que se hará será buscar una representación de los datos que tienen las características de una cara de un perro y por otro lado una representación de la cara de un gato mediante las técnicas mencionadas anteriormente, para así entrenar los datos mediante algún algoritmo. Mientras que para un aprendizaje automático tradicional costaría mucho generar de manera tan sencilla cada parte de la cara que tienen ambos animales, ya que los datos son muchísimos (píxeles: muchos datos). Costaría bastante que pueda aprender el algoritmo con tales niveles de magnitud, ver con que parámetros puedo reconocer una característica (ojo, nariz, etc). Siendo muchísimo más costoso que usando Deep Learning.

3. Para sumar dos números mediante el lenguaje Prolog, debemos definir lo que representa un número y posteriormente definir la suma.

Podemos definir el concepto de número en Prolog:

**numero(0).**

**numero(s(A)) :- numero(A).**

Esto lo podemos verificar haciendo la siguiente consulta:

**numero(A).**

Obteniendo esto así sucesivamente:

**A = 0**

**A = f(0)**

**A = f(f(0))**

**A = f(f(f(0)))**

**A = f(f(f(f(0))))**

**A = f(f(f(f(f(0)))))**

**A = f(f(f(f(f(f(0))))))**

Podemos analizar que finalmente se crea un enumerador. Además, teniendo un número se puede construir el siguiente número agregando otra  $f(0)$ . Llegando a la conclusión que mediante inducción (caso base, más que es lo que pasa en el siguiente paso) se le enseñó el programa crear un enumerador.

Ahora se busca un predicado que permita sumar dos números.

Primero analizamos que  $\text{suma}(A, B, C)$  funciona cuando  $C$  es la suma de  $A$  y  $B$ .

Para esto iniciamos el caso base que consiste en la suma de 0 con  $B$  me dará  $B$

**suma(0, B, B).**

**suma(f(A), B, f(C)) :- suma(A,B,C).**

Esto lo podemos verificar haciendo la siguiente consulta:

**suma(f(0),f(0),X).**

Esto lo que nos dice es que, si sumamos 1 con 1, cuanto será el resultado. Esto nos entrega:

**X = f(f(0))** que es exactamente el valor asociado.

Gracias al análisis obtenido para calcular la suma, se puede usar la consulta para generar una nueva lógica, como la resta.

Para que funcione la resta solo se debe modificar la consulta de la siguiente manera:

**suma(f(0),X,f(f(0))).**

Lo que nos dice esto es que, si es posible que la suma 1 más un  $X$  me de 2, lo que finalmente se define la resta señalando cuando es la resta de  $2-1$ . Obteniendo lo siguiente:

$$X = f(0)$$

Quedando demostrada la resta bajo el mismo concepto de suma creado anteriormente.

Finalmente, es muy especial lo que tiene Prolog en diferencia al resto de los lenguajes de programación ya que no solo se aplicó una suma como en cualquier otro lenguaje de programación, si no que el programa entendió el concepto de la suma.

Al entender este concepto, bajo la misma lógica se podrán comprender problemas mas complejos como ecuaciones simples o restas, conceptos que lenguajes como Python no tienen. Además, Prolog funciona bajo demostraciones y lógica, por lo que entiende lo que está pasando. Por ejemplo, en la suma mediante Prolog yo puedo invertir el orden o mediante la misma idea de suma puedo generar una ecuación simple, mientras que en Python solo puedo llevarlo acabo como una función en un sentido.

## Parte 4: Respuestas Preguntas de Materia

1. La diferencia entre un algoritmo eficiente de resolver y uno eficiente de verificar depende de la clase de problemas que son. Esto debido a que un problema eficiente de resolver toma un tiempo polinomial y es de una clase de problemas P. Esto quiere decir que el número de pasos que toma el algoritmo sobre esa instancia es siempre menor que el tamaño del input de un polinomio. Mientras que un problema eficiente de verificar si existe un algoritmo que se pueda verificar en tiempo polinomial responden a la pregunta. Esto lo podemos diferenciar ya que un problema eficiente de resolver es obtener un algoritmo que satisfice el problema en un tiempo polinomial, mientras que eficiente de verificar es que dada la respuesta obtenida pueda comprobarlo en un tiempo polinomial.

Esto tiene suma relevancia con los conceptos de problemas P y NP. Esto debido a que un problema de clase NP es un problema en que es eficiente de verificar como por ejemplo el camino Hamiltoniano visto en clases. Mientras que se define como problemas de clase P como los problemas que son eficientes de resolver como por ejemplo el camino Euleriano visto en clase.

2.
  1. No, no es posible. A partir de la salida no es posible reconstruir la entrada ya que como sabemos, las funciones hash tienen como propiedad que el output no nos entrega información acerca del input, por lo que en este caso a partir de la salida no podemos saber nada de la entrada, por lo que sería imposible de reconstruir.
  2. Si, si es posible. Como vimos en clases, gracias a las propiedades que tienen las funciones hash y sus punteros. Es posible verificar la integridad de los datos de un bloque ya que siempre el puntero de hash de un bloque está asociado al puntero de hash que se encuentra en el bloque anterior por lo tanto a partir de el bloque en que estoy puedo verificar mis datos y mediante el hash que está en mi bloque puedo verificar los datos que se encuentran en el bloque anterior.
  3. Lo más costoso en la red bitcoin es la minería. En que los mineros generan nuevos bloques para obtener como recompensa nuevos bitcoins y se le da una comisión en las transacciones. Lo que hacen es generar bloques y encontrar el Nonce (campo del bloque donde se coloca un número). Prueban con distintos Nonce hasta encontrar alguno en que el hash sea menor que cierto número. Debido a esto, gastan mucha energía ya que hoy en día utilizan hardwares especializados en calcular los distintos Nonces.  
Esto tiene como implicancia que cada vez será más difícil obtener bitcoins ya que se necesitarán más y mejores hardwares lo que provocará que los bitcoins pasen a ser una materia prima como el oro, siendo una moneda confiable que no se devaluará fácilmente siendo una moneda efectividad para una futura economía global.



3. En esta parte gracias a la entrada **0011010** calcularemos la salida paso a paso, esto gracias a la tabla que dan:

1. **Paso 0:**

E actual=q0, S actual=0

E nuevo=q1, S nuevo=0

2. **Paso 1:**

E actual=q1, S actual=0

E nuevo=q0, S nuevo=1

3. **Paso 2:**

E actual=q0, S actual=1

E nuevo=q0, S nuevo=1

4. **Paso 3:**

E actual=q0, S actual=1

E nuevo=q0, S nuevo=0

5. **Paso 4:**

E actual=q0, S actual=0

E nuevo=q1, S nuevo=1

6. **Paso 5:**

E actual=q1, S actual=1

E nuevo=q1, S nuevo=0

7. **Paso 6:**

E actual=q1, S actual=0

E nuevo=q0, S nuevo=\_

8. **Paso 0:**

E actual=q0, S actual=\_

Respuesta: NO

La salida de esta máquina de Turing es NO

El que llegue o no al estado final es debido a varios factores. Uno de ellos es el programa, ya que, si el conjunto de instrucciones dado genera una repetición continua de los pasos en las celdas generaría finalmente un loop infinito, repitiendo las instrucciones sistemáticamente, nunca encontrado una respuesta YES/NO.