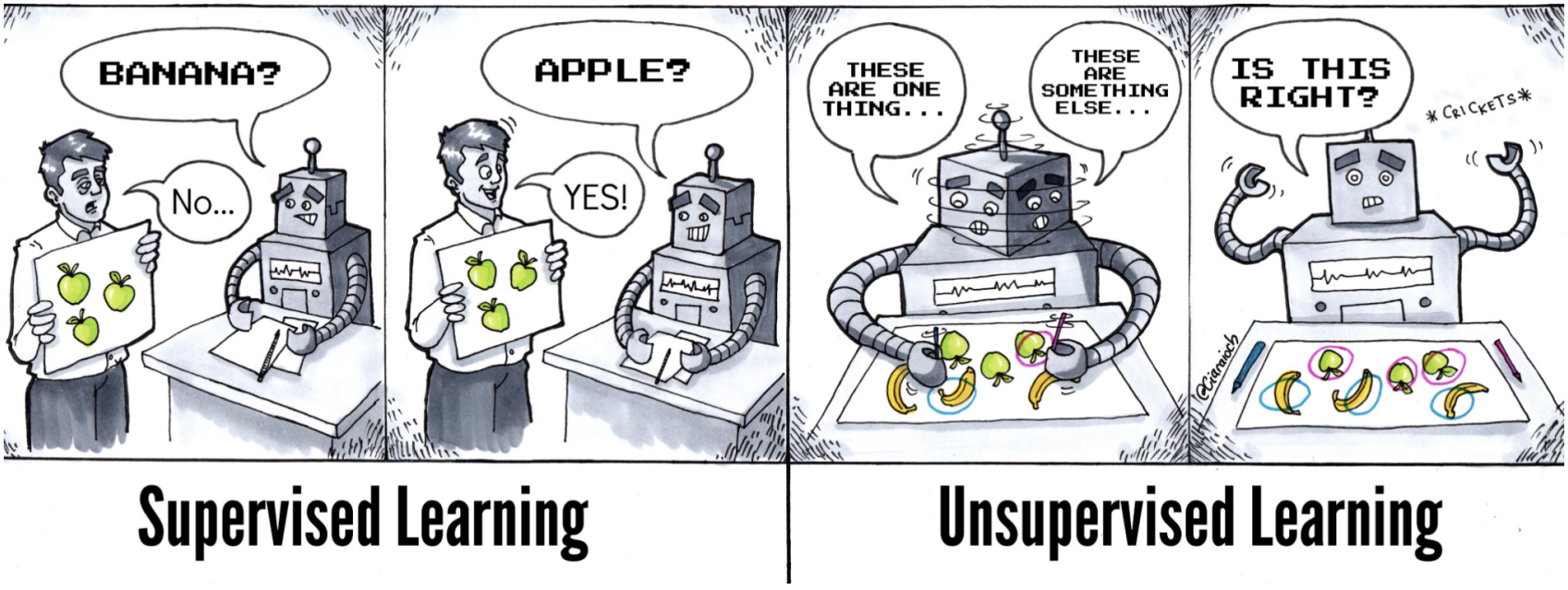
Unsupervised Learning

Fernanda Sobrino

6/22/2021

Introducción

Aprendizaje no supervisado



Por qué es una buena idea?

- Problema: Qué hacemos cuando nuestros datos no tienen labels?
  - solución: aprendizaje no supervisado
- Encuentra patrones desconocidos en los datos
- Ayuda a encontrar características
- No necesitamos entrenar nada

Diferencias entre aprendizaje supervisado y no supervisado

Parámetros	Supervisado	No Supervisado
inputs	hay labels	no hay labels
complejidad	'sencillo'	complejo
accurancy	bastante	un poco menos

Tipos

- ML:
  - Clustering: K-means, hierarchicahl clustering, probabilistic clustering
  - Reducción de dimensiones: PCA, SVD, etc
  - Generative models: intentan conocer la escencia de los datos para poder generar datos similares
- Deep Learning:
  - autoencoders: intentan aprender una aproximación de la función identidad, ayudan a encontrar estructuras difíciles de ver en los datos. Ayuda a reducir la dimensionalidad parecido a un PCA
  - sparce coding

Problemas

- como no tenemos labels no sabemos si el algoritmo hizo un buen trabajo o no
- cuando hay clasificaciones estas clases no necesariamente se pueden traducir a algo que los humanos entiendan
- hay que invertir tiempo en interpretar los resultados

PNL no supervisado

Topic Modeling

- algoritmos para descubrir los temas principales de un corpus no estructurado
- no necesitamos ninguna información previa, solo decidir en cuantos temas tenemos K
- los temas se conocen como latentes porque solo aparecen durante el proceso del modelo
- extraen patrones de grupos de palabras y sus frecuencias en los documentos

Por qué queremos esto?

- topic modeling nos da métodos para organizar, entender, buscar y resumir texto de manera automática
- es bueno:
  - descubriendo temas escondidos
  - clasificando los documentos en estos temas escondidos
  - usando esta clasificación par organizar, resumir, etc

Latent Dirichlet Allocation

- uno de los modelos mas populares para hacer topic modeling
- cada documento está conformado de varias palabras y cada tema esta asociado con un conjunto de palabras
- el objetivo es encontrar los temas a los que un documento pertenece utilizando las palabras dentro de el

Latent Dirichlet Allocation

- cada celda representa la probabilidad de que la palabra pertenezca al tema

	Word1	word2	word3	word4	.....
Topic1	0.01	0.23	0.19	0.03	
Topic2	0.21	0.07	0.48	0.02	
Topic3	0.53	0.01	0.17	0.04	

Each topic contains a score for all the words in the corpus.

Palabras representativas para cada tema

- podemos escojer las x palabra con mayor probabilidad para describir cada uno de los temas
- alternativamente podemos decidir un umbral y todas las palabras que lo atraviesen describen al tema

Supuestos

- cada documento es una bolsa de palabras
- podemos eliminar palabras comunes (hasta 80% de las palabras que aparecen en todos los documentos pueden ser eliminadas sin que perdamos mucha información)
- sabemos cuantos temas distintos queremos

Cómo funciona LDA?

- consta de dos partes principales
  1. las palabras que pertenecen a un documento (lo sabemos)
  2. las palabras que pertenecen a un tema o la probabilidad de que pertenezcan a el (no lo tenemos)

Algoritmo para la segunda parte

- asignar aleatoriamente a las palabras de cada documento pertenencia a alguno de los k temas
- para cada documento d calculamos para cada palabra lo siguiente:

Algoritmo para la segunda parte

1. $p(topic = t|document = d)$ la proporción de palabras en el documento d asignadas al tema t
2. $p(word = w|topic = t)$ la proporión de asignaciones al tema t sobre todos los documentos que vienen de la palabra w. Intenta medir cuantos documentos están en el tema t por la palabra w.

Algoritmo para la segunda parte

- LDA representa los documentos como un mezcla de temas
- representa los temas como una mezcla de palabras
- si una palabra tiene una probabilidad alta de estar en un tema
- todos los documentos con esa palabra van a estar mas asociados con t

Algoritmo para la segunda parte

- por último actualizamos la probabilidad

$$p(w,t) = p(t|d) * p(w|t)$$

Problemas con LDA * K fija * LDA no captura correlaciones entonces puede agrupar temas no relacionados * Con datos limitados o textos cortos no lo hace tan bien * Bolsa de palabras * no supervisado

Reducción de dimensiones

La maldición de la dimensionalidad

- Problema:
  - incremento exponencial en el tamaño de los datos causados por un número grande de dimensiones
  - entre mas dimensiones tengan los datos se vuelve más difícil procesarlos
- Solución:
  - reducción de dimensiones

Reducción de dimensiones

Los métodos de reducción de dimensiones reducen el tamaño de los datos extrayendo información relevante y desechando el resto de datos como ruido

- SVD (Singular Value Decomposition)
- PCA (Análisis de componentes principales)
- Linear Discriminant Analysis
- ...

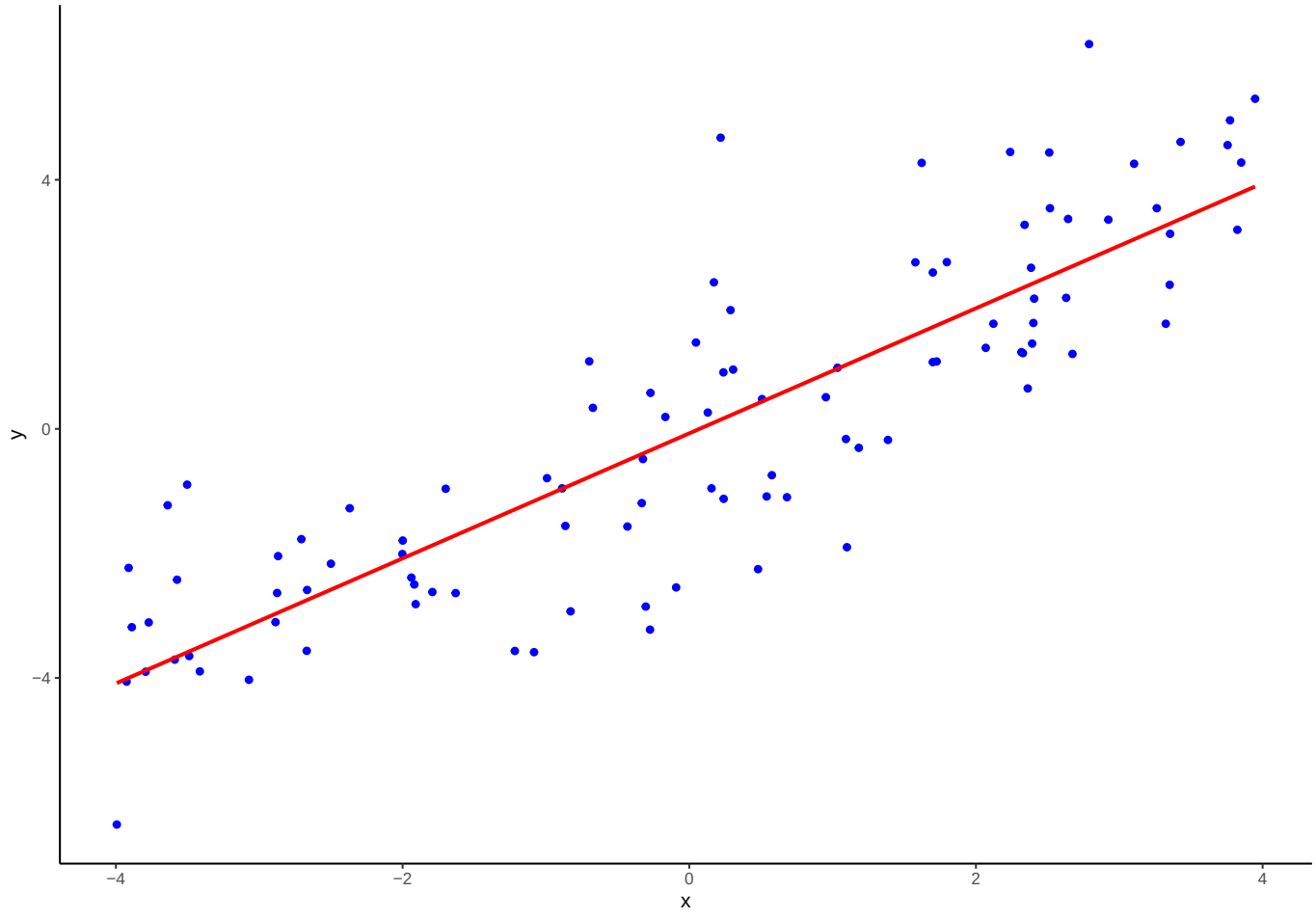
Análisis de componentes principales (PCA)

- en la intersección entre aprendizaje no supervisado y procesamiento de datos
- va a ser útil para reducir el número de features
- PCA nos permite reducir la cantidad de variables intentando considerando tanto de la varianza de las features como sea posible
- intuición: si tienes muchas variables muchas de ellas probablemente van a estar correlacionadas

Intuición

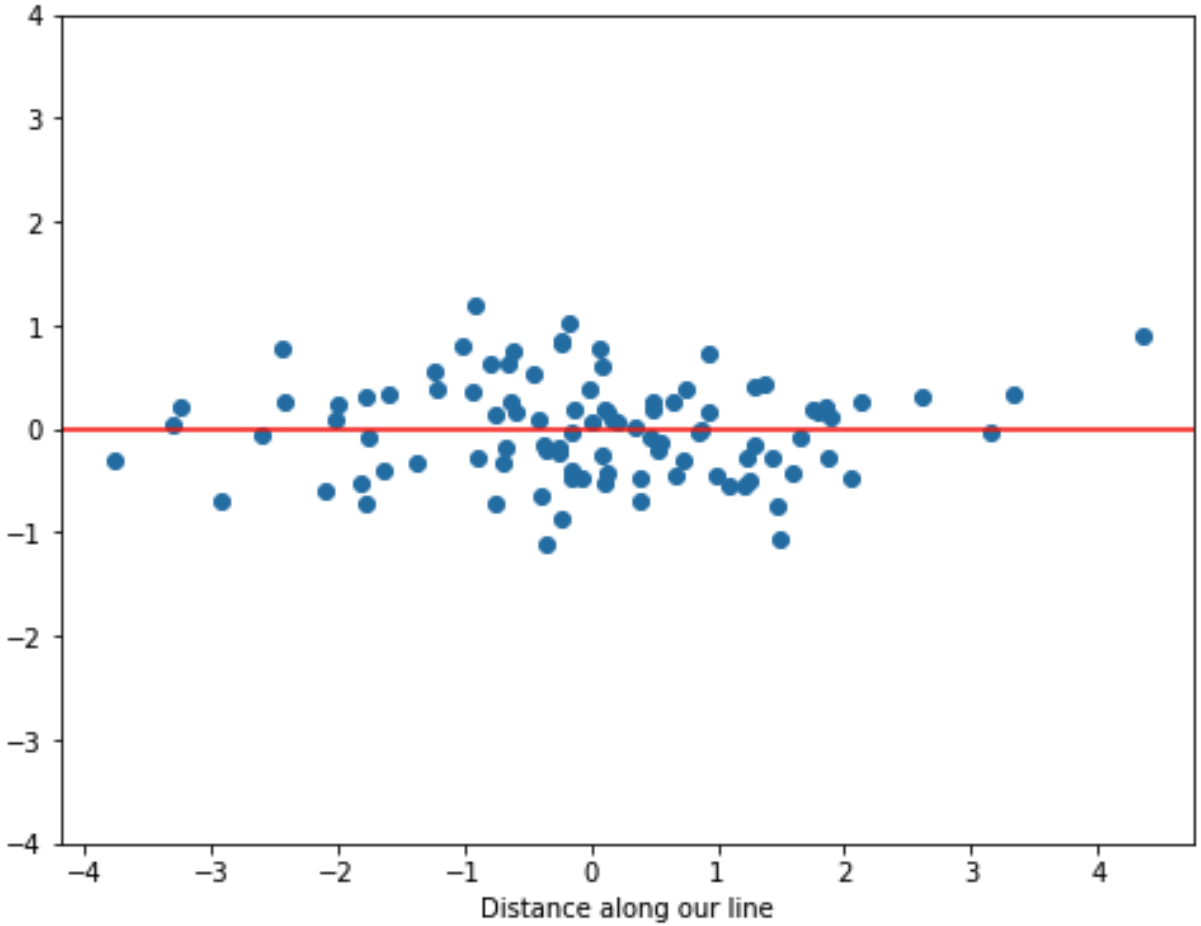
- variables posiblemente correlacionadas

```
FALSE `geom_smooth()` using formula 'y ~ x'
```



Intuición

- podemos usar la distancia entre los puntos y la linea como una nueva variable
- podemos describir los puntos como si se encuentran cerca o lejos de la linea

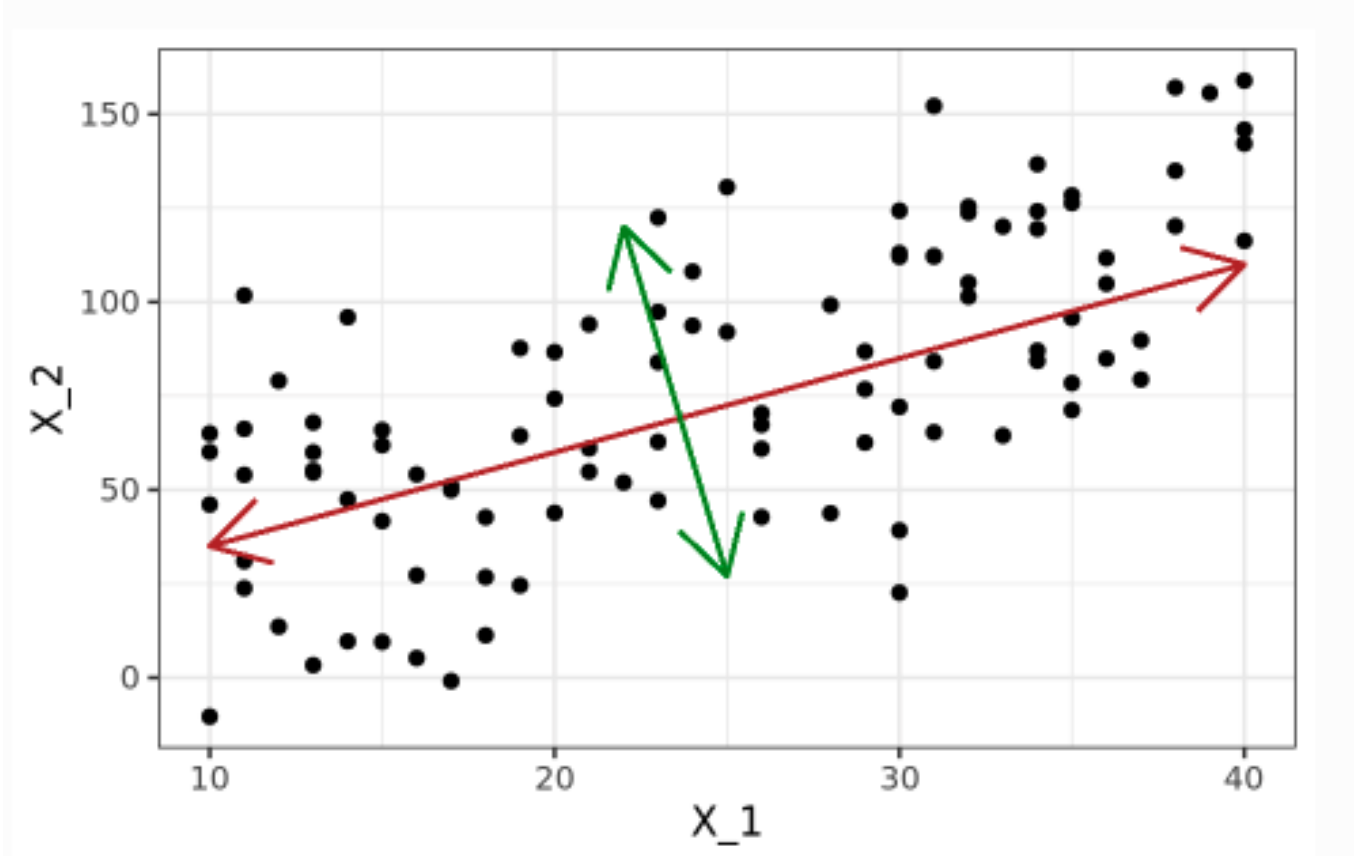


Intuición

- seguimos teniendo dos dimensiones pero:
  - la mayoría de la varianza ahora esta en una sola dimensión
  - ahora $y' \in [-1, 1]$
- PCA generaliza esto

PCA

- genera un nuevo conjunto de ejes
- los rota de tal manera que el primer eje caiga sobre la dirección que los datos muestrán mayor varianza
- estos nuevos ejes son los componentes principales



Problemas con PCA

- Perdida de varianza
- Perdida de grupos
- Perdida de patrones