

SINGULAR VALUE DECOMPOSITION

SERGIO PERNICE

UCEMA

PERSPECTIVA

- Regresiones
- $\mathbf{x}_i \in \mathbb{R}^{d \times 1}$ es una muestra de las “variables explicativas”, contamos con n de ellas.
- $y_i \in \mathbb{R}$ es un valor de la variable que queremos explicar.
- **Matriz de datos** $D \in \mathbb{R}^{n \times d}$, n = número de muestras (samples), d = dimensionalidad de la muestra (features) (\mathbf{x}_i^T es la fila i -ésima) .
- **Objetivo:** que la máquina “aprenda”, a partir de los datos, la función $y(\mathbf{x})$.
- **Método:** “entrenamos” a la máquina con los datos $y_i, i = 0, \dots, n - 1$, y buscamos los parámetros que minimicen la función error, o costo, que definimos como el promedio del cuadrado de los errores.
- Las regresiones son un ejemplo de “**aprendizaje supervisado**”: nosotros, los entrenadores, le damos las respuestas correctas en muchos ejemplos para que la máquina aprenda la función.
- Esto imita cierto tipo de aprendizaje humano. Como cuando le señalamos a un niño un perro y le decimos “perro”. El niño, de alguna manera, termina aprendiendo una función que hace que cuando vea perros sabe que es un perro y cuando no vea perros sabe que lo que esta viendo no es un perro.
- **Ejemplos:** miles de ejemplos en economía.

PERSPECTIVA

- PCA
- $\mathbf{x}_i \in \mathbb{R}^{d \times 1}$ es una muestra de los datos que queremos estudiar, contamos con n de ellas.
- No hay variables que queramos explicar. Simplemente queremos ver si podemos descubrir estructura en los datos. Idealmente de baja dimensionalidad.
- **Matriz de datos** $D \in \mathbb{R}^{n \times d}$, n = número de muestras (samples), d = dimensionalidad de la muestra (features) (\mathbf{x}_i^T es la fila i -ésima) .
- **Objetivo:** que la máquina descubra la mejor aproximación de baja dimensionalidad de los datos para que los podamos entender mejor.
- **Método:** buscamos las “mejores proyecciones” de menor dimensión: $j < d$, idealmente $j \ll d$.
- PCA es un ejemplo de “**aprendizaje NO supervisado**”: no le damos las respuestas correctas para que la máquina aprenda una función. La maquina tiene que encontrar estructura de manera objetiva (a veces un poco la ayudamos con el pre procesamiento de datos).
- **Ejemplos:** ADN: $n \sim 2000, d = 200.000, j = 2, 3$. En economía: varias variables que caracterizan a cada país (PDI, población, Gini, etc.), encontrar las d dimensiones que mejor aproximan los datos, que tan bien los aproximan, nos ayuda a entenderlos?.

PERSPECTIVA

- Tanto en regresiones como en PCA trabajamos con una matriz de datos D que tiene una interpretación muy diferente de filas y columnas.
- Cada fila representa una muestra de los datos. Cada uno de esos datos tienen d “features”, o dimensiones.
- Las columnas representan, para cada feature, los valores de las distintas muestras.
- En ambas metodologías, la interpretación geométrica convive con una interpretación estadística: la matriz $D^T D$ representa a la matriz varianza-covarianza de los datos. Son como dos caras de los datos.

PERSPECTIVA

- En muchos casos ocurre que no hay una diferenciación entre filas y columnas, y es arbitrario considerar a D o a D^T como matriz original.
- Por ejemplo, imaginemos en una economía una matriz en la que tanto las filas como las columnas representan personas humanas y jurídicas, y el elemento ij de la matriz representa alguna medida de cuánto comercian entre sí la persona i con la persona j .
- En esta matriz obviamente no hay diferenciación entre filas y columnas.
- Esta matriz es gigante, complejísima, ininterpretable en crudo.
- ¿Podemos encontrar una aproximación a dicha matriz, que por un lado sea lo más objetiva posible, libre de nuestros preconcepciones respecto de lo que impulsa a las transacciones económicas, y por el otro sea de baja dimensionalidad, como para que sea más fácilmente interpretable?

PERSPECTIVA

- PCA es básicamente libre de nuestras preconcepciones, los datos hablan solos, PCA simplemente intenta descubrir estructura en ellos, y cuando esa estructura es de relativamente baja dimensionalidad, como ocurre de manera sorprendentemente frecuente, nos ayuda a entender mejor los datos.
- Pero no funciona para nuestro objetivo, porque en la esencia de PCA está la diferenciación entre “puntos” (las distintas filas) y las dimensiones en las que dichos puntos viven (las distintas columnas). La idea de PCA es capturar cuándo esos puntos se distribuyen aproximadamente en un subespacio de menor dimensionalidad.
- La cosa viene confusa porque, por ahora por lo menos, no está ni siquiera claro qué significa encontrar **una aproximación de la matriz de menor dimensionalidad de una matriz**.

MATRICES 2X2

- Queremos tratar de generar una aproximación de una matriz dada por una más simple.
- Además queremos mantener igual el “estatus” de las filas y el de las columnas.
- Una propiedad de las matrices que es igual para filas que para columnas es su “rango”. Como vimos, el rango fila de una matriz (el número de filas linealmente independientes) es siempre igual a su rango columna (el número de columnas linealmente independientes). Pasara entonces por aproximar a una matriz por otra de menor rango??
- Empecemos una exploración empírica con las matrices más sencillas: de 2x2.
- Ver JN SVD2x2.ipynb y sección 2 del paper de SVD.