Machine Learning No Supervisados

Componentes Principales

Tomás Fontecilla

24 de agosto de 2023

Machine Learning No Supervisado

Hasta ahora hemos hablado de lo que compone el curso. En general es bien conocido el método supervisado de análisis, donde disponemos de una variable que nos permite revisar si acaso lo que hacemos "es correcto". Así, regularmente en el caso supervisado uno evalúa si la salida del modelo está cerca (caso valores numéricos) o igual (caso valores categóricos) al real.

En el caso de los modelos no supervisados no es tan simple. Usuamente, se incorpora como parte del análisis exploratorio de datos.

Machine Learning No Supervisado

En general,

- si el problema es de baja dimensionalidad ($p \leq 3$) existen varios métodos no paramétricos que estiman las propiedades del conjunto de datos y son representables gráficamente.
- Estos modelos no paramétricos fallan rápidamente en más dimensiones debido a la maldición de la dimensionalidad.
- Debido a esto, uno debe conformarse con realizar modelos globales algo crudos, como mezclas de gaussianas o estadísticas descriptivos que puedan caracterizar los datos.

Machine Learning No Supervisado

Para estas estadísticas descriptivas

- Componentes principales (escalamiento multidimensional, mapas auto organizados entre otros) intenta identificar colectores de baja dimensión dentro del espacio X que representan una alta densidad de los datos. Esto provee información sobre las asociaciones que permite considerar o no si son funciones de una variable menor "latente".
- K-means (y otros métodos de clustering) intenta encontrar regiones en el espacio X que contengan distintas modas de X, para determinar si X puede ser representado como una mezcla de densidades simples representadas por los tipos de clases de observaciones.
- Los modelos de mezclas tienen un objetivo similar al anterior. A través de reglas de asociación intenta construir descripciones simplesque describan regiones de alta densidad en caso especial de datos evaluados binariamente de muy alta dimensionalidad.

Análisis de Componentes Principales - Conceptos Clave

- Matriz de Covarianza Σ
- Valores propios λ y vectores Propios ν
- Varianza Explicada $\frac{\lambda_i}{\sum_{i=1}^{p} \lambda_i}$
- Componentes Principales $(PC_1, PC_2, PC_3, ..., PC_p)$
- Ortogonalidad de los Componentes Principales $(\nu_i^T \nu_j = 0,\ i \neq j)$
- Ordenar valores propios por varianza $(\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_p)$

Recordemos que la descomposición propia $C = V \Lambda V^T$ está dada por $\Lambda = diag(\lambda_1, ..., \lambda_n)$ donde $(\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_p \geq 0)$ son valores propios y V es ortogonal y la m-ésima columna de V es el vector propio de C.

Machine Learning - Motivación

Componentes Principales

- Para poder eliminar la *maldición de dimensionalidad* de los datos, podemos tomar el enfoque de proyectar p variables en q rangos de tal manera que $q \leq p$. Con componentes principales intentamos perder la menor cantidad de información posible.
- Para no perder mucha información haremos algo que normalmente intentaríamos evitar: realizaremos las proyecciones *maximizando la varianza*. Dicho de otra forma, *minimizaremos la distancia* entre los datos originales y las proyecciones.
- Para simplificar, asumiremos que los datos están "centrados", lo que hace que cada variable tenga media 0. Así,

$$x^T x = n\Sigma$$

donde Σ es la matriz de covarianzas de los datos y n es el número de datos.

Componentes Principales - Minimizando los residuales de la proyección

Empecemos con una proyección unidimensional.

- La proyección de un vector de datos $\vec{x_i}$ sobre el vector unitario $\vec{\omega}$ (que es una linea que pasa por el origen) como $\vec{x_i} \cdot \vec{\omega}$ lo que lo hace escalar.
- Esta es la distancia de la proyección al origen.
- la coordenada realmente es $(\vec{x_i}\vec{\omega})\vec{\omega}$. La proyección media será cero, porque el promedio de los vectores $\vec{x_i}$ es cero:

$$\frac{1}{n}\sum_{i=1}^{n}(\vec{x_i}\vec{\omega})\vec{\omega} = ((\frac{1}{n}\sum_{i=1}^{n}(\vec{x_i})\vec{\omega})\vec{\omega})$$

- si intentamos usar los vectores proyectados o imagen en lugar de los vectores originales habrá errores, porque las imágenes no coincidirán con el vector original. Esta diferencia se llama el **residual** de la proyección.
- Su tamaño está dado por:
$\|\vec{x_i} - (\vec{x_i} \cdot \vec{\omega})\vec{\omega}\|^2 = \vec{x_i} \cdot \vec{x_i} - (\vec{x_i} \cdot \vec{\omega})^2$ dado que $\vec{\omega} \cdot \vec{\omega} = \|\vec{\omega}\|^2 = 1$

Componentes Principales - MSE

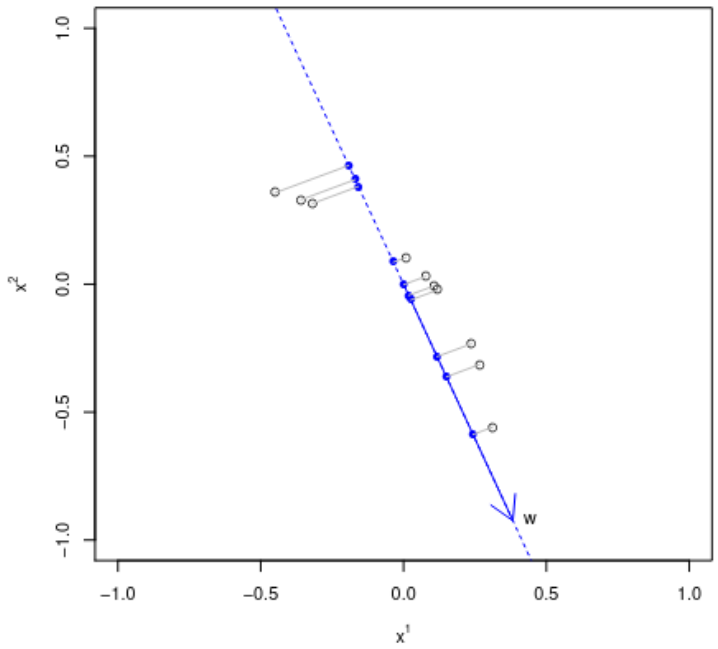
Al sumar los residuos, se obtiene:

$$\begin{aligned} MSE(\vec{\omega}) &= \frac{1}{n}\sum_{i=1}^{n}\|\vec{x_i}\|^2 - (\vec{\omega}\vec{x_i})^2 \\ &= \frac{1}{n}\left(\sum_{i=1}^{n}\|\vec{x_i}\|^2 - \sum_{i=1}^{n}(\vec{\omega}\vec{x_i})^2\right) \end{aligned}$$

La suma no depende de $\vec{\omega}$ asi que no importa para minimizar la media cuadrática de los residuos.

Componentes Principales - Resumiendo la historia... y la matemática detrás

Todo esto es conocido!



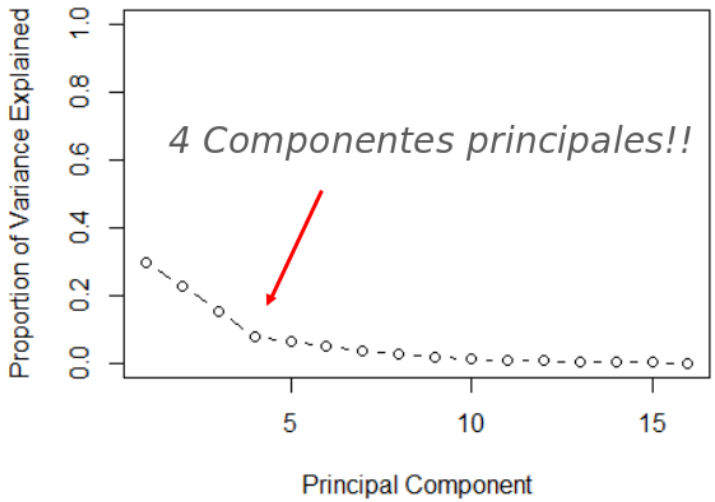
- Queremos maximizar varianza
- queremos proyectar con respecto a ejes
- queremos minimizar los errores

Componentes Principales - ¿Cuántos Componentes?

utilizando la varianza maximizada y la proporción de esta explicada por cada componente, podemos realizar un gráfico scree

Sea λ la varianza explicada por el componente principal. Luego, la proporción está dada por:

$$\frac{\lambda_p}{\sum_{i=1} \lambda_i} = \frac{(u^p)^T S u^p}{\sum_{i=1} (u^i)^T S u^i}$$



Componentes Principales - Recordatorio

Debido a que este algoritmo es no supervisado, no tenemos una forma real para saber cómo validar directamente estos modelos, por lo tanto no existe una forma pura de determinar si están bien ajustados y suele hacerse combinaciones de modelos no supervisados e interpretaciones gráficas para determinar si hay cambios o no en los resultados.
En el caso de PCA, es la distribución de la varianza la que manda, y la acumulación de ella en los componentes para determinar si es suficiente o mejor seguir con otro método.

Algoritmo por matriz de varianza covarianza

- El primer paso para hacer componentes principales es estandarizar los datos.
  La estandarización, o conversión la haremos con:

$$Z = \frac{X_{ij} - \bar{X}_j}{\sigma}$$

- Calcular la matriz de varianza-covarianza

$$\Sigma = \frac{Z^T Z}{n-1}$$

- Calcular valores propios (λ) y vectores propios (ν)

$$\Sigma \nu_i = \lambda_i \nu_i \qquad \text{ó} \qquad C = V \Lambda V^T$$

- Seleccionar componentes principales, escogiendo basado en el valor de varianza explicada.
- Proyectar los datos en los componentes principales a Y, con Y = ZP (P es la matriz de componentes)

Algoritmo por SVD

- El método de varianza covarianza puede volverse lento, por lo que habría que explorar alternativas. Una de ellas es el método por descomposicón de valor singular (SVD).
- SVD permite calcular los Componentes Principles que explican mayor varianza, lo que hace más rápido que calcularlos *todos*.
- El SVD de $X \in \mathbb{R}^{n \times p}$ es $X = USV^T$ donde
  - $U \in \mathbb{R}^{n \times n}$ es ortogonal
  - $V \in \mathbb{R}^{p \times p}$ es ortogonal
  - $S \in \mathbb{R}^{n \times p}$ es cero excepto para $s_{11} \geq s_{22} \geq ... \geq 0$, llamados valores singulares

Algoritmo SVD

- Descomponer la matriz de datos $X = U\Lambda V^T$
  - U: Matriz de vectores singulares izquierdos, que representa la relació entre las filas de X
  - Λ: Matirz diagonal de valores singulares, que contiene información de la importancia de cada componente
  - V^T la transpuesta de la matriz de vectores singulares derechos, representa la relación entre columnas de X.
- La conexión entre métodos es:

$$V\Sigma V^T = C = \frac{1}{1-n}X^T X = V(\frac{1}{1-n}\Lambda^T \Lambda)V^T$$

Consideraciones

- Hay que balancear la reducción de dimensiones con la pérdida de información.

Aplicaciones en el mundo real

- Genética
- Procesamiento de imágenes
- Finanzas
- Ciencias sociales

Limitaciones

- Presunción de linearidad
- Interpretabilidad de los componentes
- No es apropiado para todo tipo de datos