Autoencoders

Matias Vera - Juan Zuloaga

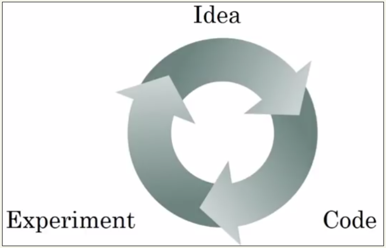
Centro de Simulación Computacional para Aplicaciones Tecnológicas

Agenda

1. ¿Que es un Autoencoder?
2. ¿Para que sirve un Autoencoder?
3. ¿Cuando puede servir autoencoder?
4. Principal Components Analysis (PCA)

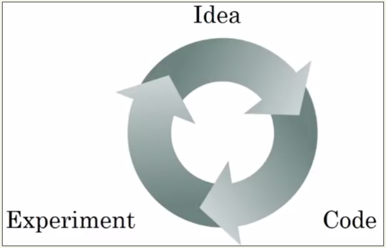
Aprendizaje Estadístico

- No se conoce la verdadera estadística.
- Se aprende por medio de datos.
- El buen desempeño no debe limitarse a los datos conocidos.



Aprendizaje Estadístico

- No se conoce la verdadera estadística.
- Se aprende por medio de datos.
- El buen desempeño no debe limitarse a los datos conocidos.

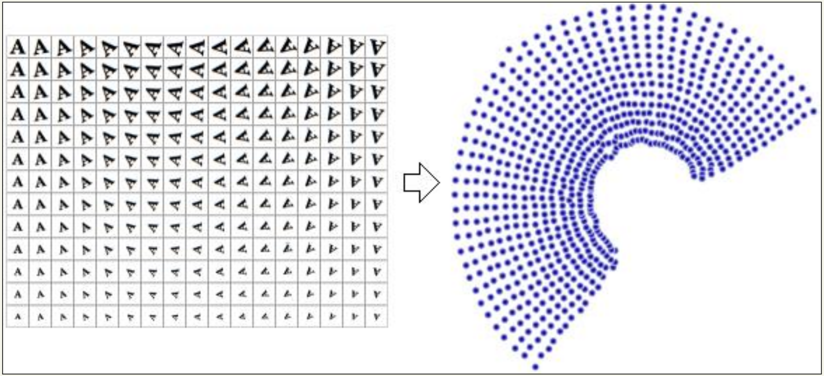


TIPOS DE APRENDIZAJES

- Aprendizaje supervisado: Cuento con pares de datos $\{(\mathrm{x}^{(i)}, y^{(i)})\}_{i=1}^{n}$.
- Aprendizaje no supervisado: Cuento solamente con datos $\{\mathrm{x}^{(i)}\}_{i=1}^{n}$.
- Aprendizaje semi-supervisado: Cuento con muchos datos no supervisados y unos pocos supervisados.

Manifold

¿Cuál es la dimensión efectiva de los datos?



Manifold

¿Cuál es la dimensión efectiva de los datos?

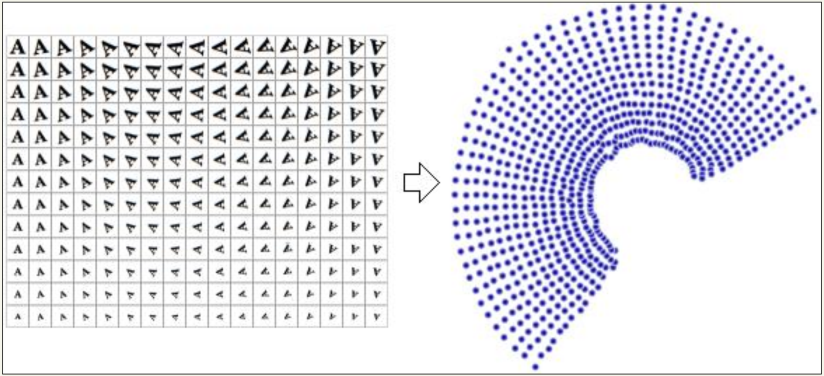


Diagrama en bloques de un Autoencoder

$$x \longrightarrow \boxed{\text{Encoder}} \xrightarrow{u} \boxed{\text{Decoder}} \longrightarrow \hat{x}$$

Manifold

¿Cuál es la dimensión efectiva de los datos?

Objetivo

Hay que entender que el objetivo no es simplemente reconstruir los datos. Sino que es reconstruir los datos a partir de una representación relevante para explicar algún fenómeno o resolver otra tarea. Si no se reconocen patrones en la naturaleza de los datos no hay aprendizaje.

Mathematical Snippets - "An unexpected bijection between the real plane and the real line" `https://www.youtube.com/watch?v=XcMZsF4vDbo`

Manifold

¿Cuál es la dimensión efectiva de los datos?

Objetivo

Hay que entender que el objetivo no es simplemente reconstruir los datos. Sino que es reconstruir los datos a partir de una representación relevante para explicar algún fenómeno o resolver otra tarea. Si no se reconocen patrones en la naturaleza de los datos no hay aprendizaje.

Cuidado!

Existen transformaciones $\mathcal{T} : \mathbb{R}^{d_x} \to \mathbb{R}$ biyectivas (googlear por ejemplo Teorema de Cantor-Schröder-Bernstein). Pero las representaciones reducidas obtenidas de esta manera pueden no ser interesantes. Hay que tener en cuenta la precisión del computo y, sobre todo, la aplicación en la que se va a utilizar.

Mathematical Snippets - "An unexpected bijection between the real plane and the real line" `https://www.youtube.com/watch?v=XcMZsF4vDbo`

Manifold

Regularización de autoencoders

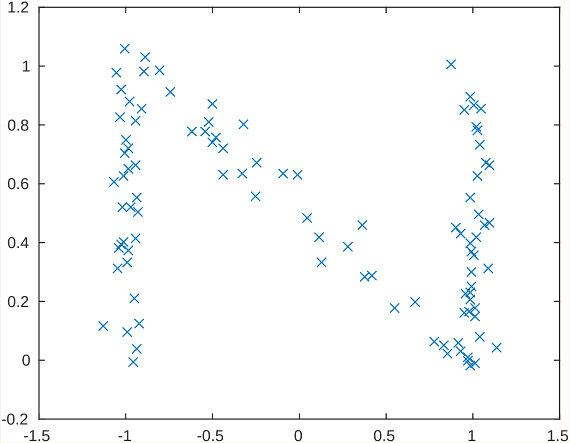
Bajo ECM para cualquier tipo de entrada	Bajo ECM para los sets de entrenamiento y testeo	Bajo ECM solamente en el set de entrenamiento
✗	✓	✗

Objetivo

No quiero memorizar el conjunto de datos ni aprender una transformación biyectiva: Busco aprender el manifold. La regularización en un autoencoder busca balancear estos conceptos.

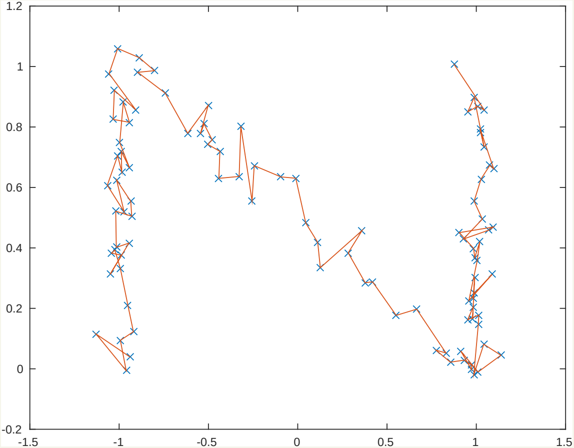
Manifold

Regularización de autoencoders



Manifold

Regularización de autoencoders



OVERFITTING

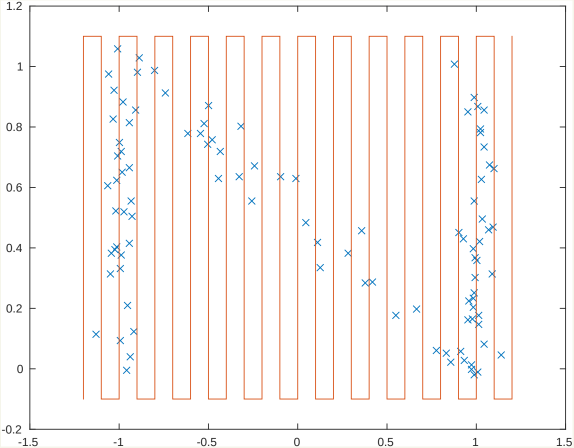
No hay aprendizaje, se están memorizando las muestras.

✗

Necesito regularización

Manifold

Regularización de autoencoders



IDENTIDAD

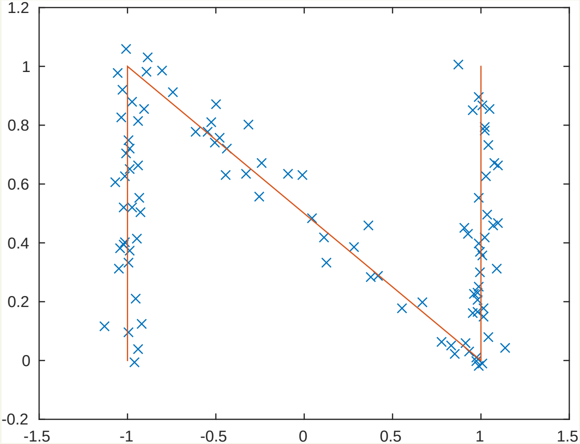
Se está aprendiendo la función identidad y no la naturaleza de los datos.

✗

Necesito regularización

Manifold

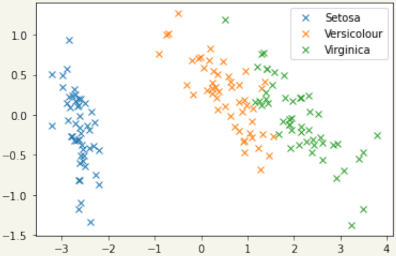
Regularización de autoencoders



Algunas Aplicaciones

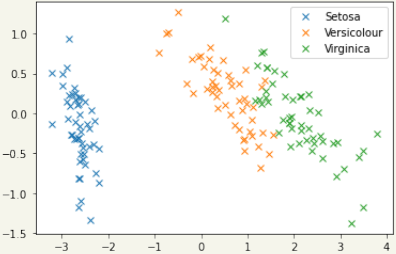
- Para efectuar una inferencia más precisa
- Para pre-procesar los datos
- Para generar datos sintéticos
- Para detectar anomalías

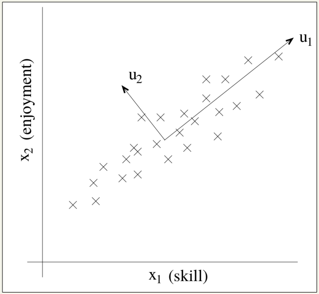
Inferencia



Visualizar en un gráfico 2d o 3d para explicar algunos fenómenos (iris dataset)

Inferencia





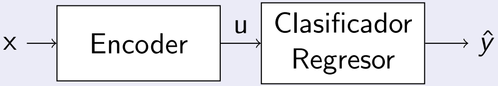
Visualizar en un gráfico 2d o 3d para explicar algunos fenómenos (iris dataset)

Generar alguna métrica que combine variables muy distintas entre si (radio-controlled helicopters)

Pre-processing

Preprocesing: Opción 1

Entrenar el autoencoder y luego usar las muestras en el espacio latente para entrenar el clasificador/regresor.



Pre-processing

Preprocesing: Opción 1

Entrenar el autoencoder y luego usar las muestras en el espacio latente para entrenar el clasificador/regresor.

$$x \longrightarrow \boxed{\text{Encoder}} \xrightarrow{u} \boxed{\begin{array}{c}\text{Clasificador}\\ \text{Regresor}\end{array}} \longrightarrow \hat{y}$$

Preprocesing: Opción 2

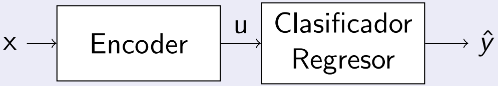
Entrenar el autoencoder y luego usar las reconstrucciones para entrenar el clasificador.

$$x \longrightarrow \boxed{\text{Encoder}} \xrightarrow{u} \boxed{\text{Decoder}} \xrightarrow{\hat{x}} \boxed{\begin{array}{c}\text{Clasificador}\\ \text{Regresor}\end{array}} \longrightarrow \hat{y}$$

Pre-processing

Preprocesing: Opción 1

Entrenar el autoencoder y luego usar las muestras en el espacio latente para entrenar el clasificador/regresor.



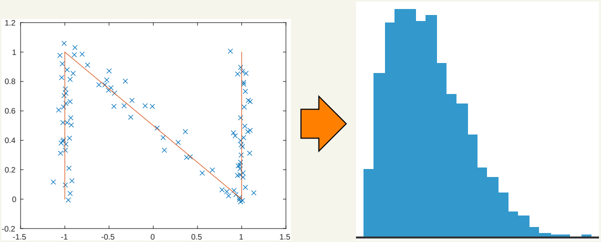
Preprocesing: Opción 2

Entrenar el autoencoder y luego usar las reconstrucciones para entrenar el clasificador.

Semi-supervise learning

Puedo usar las muestras no supervisadas para entrenar el autoencoder y las supervisadas para el clasificador o el regresor final.

Generación de datos



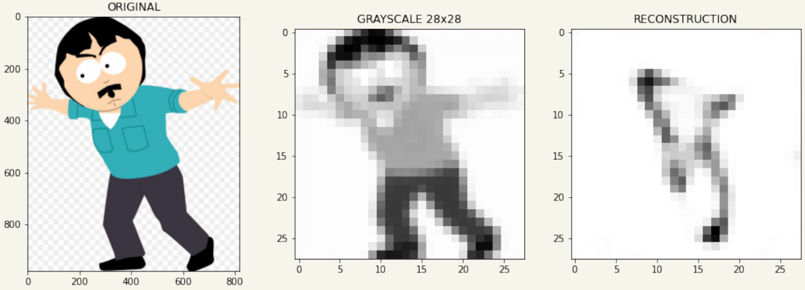
1. Aprendo el manifold
2. Genero el histograma en el espacio reducido
3. Modelo y estimo la distribución
4. Sampleo nuevas muestras en el espacio reducido
5. Reconstruyo

Detección de anomalías

Paradigma

Durante el entrenamiento un autoencoder aprende patrones en los datos para reconstruirlos con cierta facilidad. Entonces es de esperar que una muestra que no cumpla los patrones aprendidos sea más dificil de reconstruir.

EJEMPLO AUTOENCODER ENTRENADO CON MNIST:



¿Cuando usar un autoencoder?

Clasificación de las aplicaciones

Las aplicaciones de los autoencoders se dividen en dos grupos:

- Las que son relevantes por si mismas.
- Las que son un paso intermedio hacia una tarea de clasificación o regresión. $\leftarrow$ **¿Siempre servirá?**

¿Cuando usar un autoencoder?

Clasificación de las aplicaciones

Las aplicaciones de los autoencoders se dividen en dos grupos:

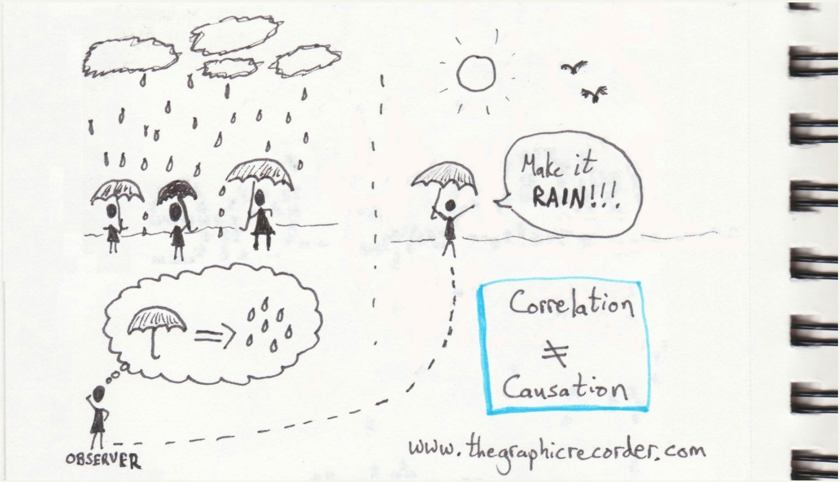
- Las que son relevantes por si mismas.
- Las que son un paso intermedio hacia una tarea de clasificación o regresión. $\leftarrow$ **¿Siempre servirá?**

¿Que distribución aprende durante el entrenamiento?

Desde un punto de vista probabilístico, el entrenamiento de un algoritmo busca aprender la distribución estadística (total o parcial) de los datos:

- **Aprendizaje supervisado**: Para cada entrada x, se desea aprender parte de la información contenida en la distribución de una variable objetivo $Y|X = x$.
- **Aprendizaje no supervisado**: Toda la información aprendida estará contenida en distribución de los datos X.

Hablemos de causalidad



Causalidad: ¿Quién causa a quién?

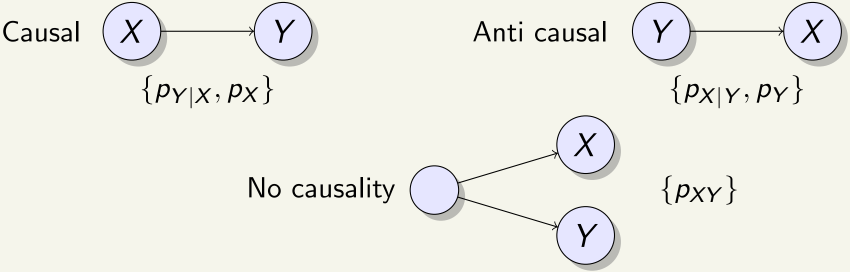
Independent Causal Mechanisms (ICM) Principle

The causal generative process of a system's variables is composed of autonomous modules that do not inform or influence each other. In the probabilistic case, this means that the conditional distribution of each variable given its causes (i.e., its mechanism) does not inform or influence the other mechanisms.

Causalidad: ¿Quién causa a quién?

Independent Causal Mechanisms (ICM) Principle

The causal generative process of a system's variables is composed of autonomous modules that do not inform or influence each other. In the probabilistic case, this means that the conditional distribution of each variable given its causes (i.e., its mechanism) does not inform or influence the other mechanisms.



Causalidad:¿Quién causa a quién?

$$Y = g(X, U) \quad \text{con} \quad X \perp U \qquad \text{o} \qquad X = g(Y, U) \quad \text{con} \quad Y \perp U$$

Causalidad:¿Quién causa a quién?

$$Y = g(X, U) \quad \text{con} \quad X \perp U \qquad \text{o} \qquad X = g(Y, U) \quad \text{con} \quad Y \perp U$$

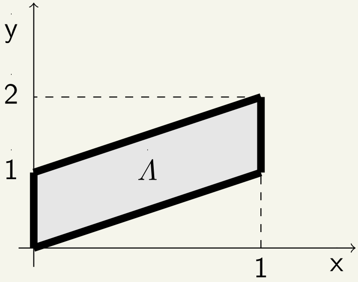
La estadística no basta!

Para toda conjunta p_{XY} siempre existe $U \perp X$ y $g(\cdot, \cdot)$ tal que $Y = g(X, U)$

Causalidad:¿Quién causa a quién?

La estadística no basta!

Para toda conjunta p_{XY} siempre existe $U \perp X$ y $g(\cdot,\cdot)$ tal que $Y = g(X, U)$



$$Y|X = x \sim \mathcal{U}(x, x+1) \equiv x + \mathcal{U}(0,1)$$

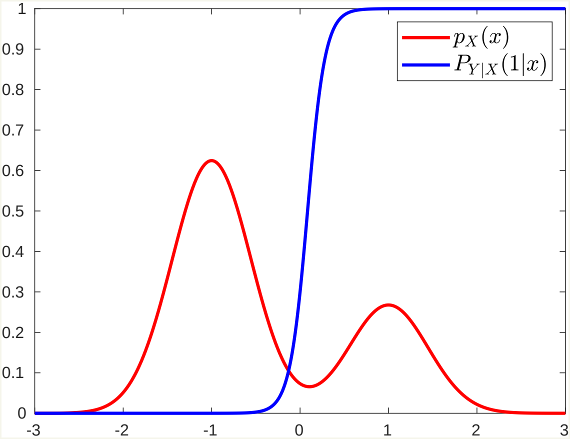
$$X|Y = y \sim \begin{cases} \mathcal{U}(0, y) & 0 < y < 1 \\ \mathcal{U}(y-1, 1) & 1 < y < 2 \end{cases}$$

$$(X, Y) \sim \mathcal{U}(\Lambda)$$

Causalidad:¿Quién causa a quién?

La estadística no basta!

Para toda conjunta p_{XY} siempre existe $U \perp X$ y $g(\cdot,\cdot)$ tal que $Y = g(X, U)$



$$Y \sim \mathrm{Cat}\{-1, 1\}$$
$$X|Y = y \sim \mathcal{N}(y, \sigma^2)$$

$$X \sim p_X$$
$$Y|X = x \sim P_{Y|X}(y|x)$$

Causalidad:¿Quién causa a quién?

La estadística no basta!

Para toda conjunta p_{XY} siempre existe $U \perp X$ y $g(\cdot,\cdot)$ tal que $Y = g(X,U)$

$$
\begin{aligned}
p_{XY}(x,y) &= e^{-x} 1\{0 < y < x\} \\
&= \underbrace{xe^{-x} 1\{x > 0\}}_{p_X(x)} \underbrace{\frac{1}{x} 1\{0 < y < x\}}_{p_{Y|X}(y|x)} \\
&= \underbrace{e^{-(x-y)} 1\{x > y\}}_{p_{X|Y}(x|y)} \underbrace{e^{-y} 1\{y > 0\}}_{p_Y(y)}
\end{aligned}
$$

Causalidad:¿Quién causa a quién?

La estadística no basta!

Para toda conjunta p_{XY} siempre existe $U \perp X$ y $g(\cdot,\cdot)$ tal que $Y = g(X, U)$

$$\begin{aligned} p_{XY}(x,y) &= e^{-x}1\{0<y<x\} \\ &= \underbrace{xe^{-x}1\{x>0\}}_{p_X(x)}\underbrace{\frac{1}{x}1\{0<y<x\}}_{p_{Y|X}(y|x)} \\ &= \underbrace{e^{-(x-y)}1\{x>y\}}_{p_{X|Y}(x|y)}\underbrace{e^{-y}1\{y>0\}}_{p_Y(y)} \end{aligned}$$

$$X = Y + \mathcal{E}(1), \qquad Y = X \cdot \mathcal{U}(0,1)$$

Causal and Anticausal Learning

Causal Learning

Desde esta perspectiva, en una configuración causal $X \to Y$ no debería ayudarnos conocer p_X a inferir $p_{Y|X}$.

Solución Óptima

Las decisiones óptimas $\hat{P}_\theta(y|x) = P_{Y|X}(y|x)$ y $\varphi_\theta(x) = \mathbb{E}[Y|X = x]$ no dependen de la marginal. Es decir, la solución es la misma por más que cambie la marginal p_X.

Causal and Anticausal Learning

Causal Learning

Desde esta perspectiva, en una configuración causal $X \to Y$ no debería ayudarnos conocer p_X a inferir $p_{Y|X}$.

Solución Óptima

Las decisiones óptimas $\hat{P}_\theta(y|x) = P_{Y|X}(y|x)$ y $\varphi_\theta(x) = \mathbb{E}[Y|X=x]$ no dependen de la marginal. Es decir, la solución es la misma por más que cambie la marginal p_X.

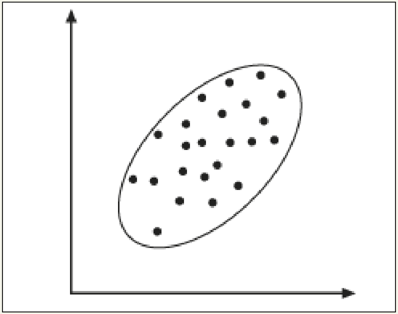
Igual un poquito ayuda

$$\arg\min_{\theta\in\Theta} \mathbb{E}[-\log \hat{P}_\theta(Y|X)] = \arg\min_{\theta\in\Theta} \mathbb{E}_{p_X}\left[D(P_{Y|X}(\cdot|X)\|\hat{P}_\theta(\cdot|X))\right]$$

$$\arg\min_{\theta\in\Theta} \mathbb{E}_P[(Y-\varphi_\theta(X))^2] = \arg\min_{\theta\in\Theta} \mathbb{E}_{p_X}[(\varphi_\theta(X)-\mathbb{E}[Y|X])^2]$$

Principal Components Analysis

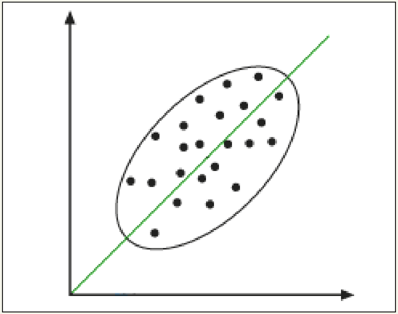
Reducción lineal



Lectura recomendada: *Andrew Ng* - "Lecture notes: Principal components analysis".

Principal Components Analysis

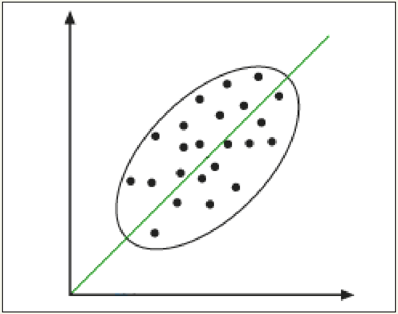
Reducción lineal



Lectura recomendada: *Andrew Ng* - "Lecture notes: Principal components analysis".

Principal Components Analysis

Reducción lineal



PASO 1: Normalizar

$$\tilde{x}_j^{(i)} = \frac{x_j^{(i)} - \mu_j}{\sigma_j}$$

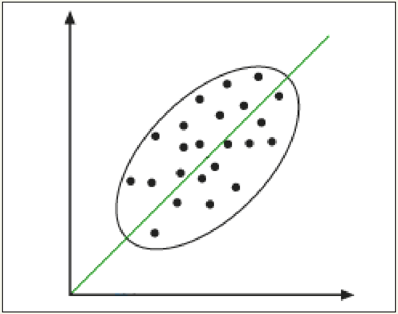
con

$$\mu_j = \frac{1}{n}\sum_{i=1}^{n} x_j^{(i)}$$

$$\sigma_j^2 = \frac{1}{n}\sum_{i=1}^{n}(x_j^{(i)} - \mu_j)^2$$

Lectura recomendada: *Andrew Ng* - "Lecture notes: Principal components analysis".

Principal Components Analysis



PASO 1: Normalizar

$$\tilde{x}_j^{(i)} = \frac{x_j^{(i)} - \mu_j}{\sigma_j}$$

con

$$\mu_j = \frac{1}{n} \sum_{i=1}^{n} x_j^{(i)}$$

$$\sigma_j^2 = \frac{1}{n} \sum_{i=1}^{n} (x_j^{(i)} - \mu_j)^2$$

PASO 2: Buscar el principal autoversor v_1

$$\min_{\substack{\mathrm{v}_1: \\ \|\mathrm{v}_1\|^2=1}} \sum_{i=1}^{n} \|\tilde{\mathrm{x}}^{(i)} - \alpha_i \mathrm{v}_1\|^2 \quad \text{con} \quad \langle \tilde{\mathrm{x}}^{(i)} - \alpha_i \mathrm{v}_1; \mathrm{v}_1 \rangle = 0$$

Lectura recomendada: *Andrew Ng* - "Lecture notes: Principal components analysis".

Principal Components Analysis

Algunas cuentas

Condicion de ortogonalidad:

$$\langle \tilde{x}^{(i)} - \alpha_i v_1 ; v_1 \rangle = 0 \quad \rightarrow \quad \langle \tilde{x}^{(i)} ; v_1 \rangle = \alpha_i \|v_1\|^2 = \alpha_i$$

Principal Components Analysis

Algunas cuentas

Condicion de ortogonalidad:

$$\langle \tilde{\mathsf{x}}^{(i)} - \alpha_i \mathsf{v}_1 ; \mathsf{v}_1 \rangle = 0 \quad \rightarrow \quad \langle \tilde{\mathsf{x}}^{(i)} ; \mathsf{v}_1 \rangle = \alpha_i \|\mathsf{v}_1\|^2 = \alpha_i$$

Optimización:

$$\min_{\substack{\mathsf{v}_1: \\ \|\mathsf{v}_1\|^2=1}} \sum_{i=1}^{n} \|\tilde{\mathsf{x}}^{(i)} - \alpha_i \mathsf{v}_1\|^2 = \min_{\substack{\mathsf{v}_1: \\ \|\mathsf{v}_1\|^2=1}} \sum_{i=1}^{n} \|\tilde{\mathsf{x}}^{(i)}\|^2 - \alpha_i^2$$

Principal Components Analysis

Algunas cuentas

Condicion de ortogonalidad:

$$\langle \tilde{\mathrm{x}}^{(i)} - \alpha_i \mathrm{v}_1; \mathrm{v}_1 \rangle = 0 \quad \rightarrow \quad \langle \tilde{\mathrm{x}}^{(i)}; \mathrm{v}_1 \rangle = \alpha_i \|\mathrm{v}_1\|^2 = \alpha_i$$

Optimización:

$$\min_{\substack{\mathrm{v}_1: \\ \|\mathrm{v}_1\|^2=1}} \sum_{i=1}^{n} \|\tilde{\mathrm{x}}^{(i)} - \alpha_i \mathrm{v}_1\|^2 = \min_{\substack{\mathrm{v}_1: \\ \|\mathrm{v}_1\|^2=1}} \sum_{i=1}^{n} \|\tilde{\mathrm{x}}^{(i)}\|^2 - \alpha_i^2$$

$$\max_{\substack{\mathrm{v}_1: \\ \|\mathrm{v}_1\|^2=1}} \frac{1}{n} \sum_{i=1}^{n} \langle \tilde{\mathrm{x}}^{(i)}; \mathrm{v}_1 \rangle^2 = \max_{\substack{\mathrm{v}_1: \\ \|\mathrm{v}_1\|^2=1}} \mathrm{v}_1^T \underbrace{\left(\frac{1}{n} \sum_{i=1}^{n} \tilde{\mathrm{x}}^{(i)} \left(\tilde{\mathrm{x}}^{(i)}\right)^T \right)}_{\boldsymbol{\Sigma}} \mathrm{v}_1$$

Principal Components Analysis

Algunas cuentas

$$J(\mathsf{v}_1) = \mathsf{v}_1^T \boldsymbol{\Sigma} \mathsf{v}_1 - \lambda \left(\mathsf{v}_1^T \mathsf{v}_1 - 1 \right)$$

Lectura recomendada: *Petersen and Pedersen* - "Matrix Cookbook".

Principal Components Analysis

Algunas cuentas

$$J(\mathsf{v}_1) = \mathsf{v}_1^T \boldsymbol{\Sigma} \mathsf{v}_1 - \lambda \left(\mathsf{v}_1^T \mathsf{v}_1 - 1 \right)$$

$$\nabla J(\mathsf{v}_1) = 2 \left(\boldsymbol{\Sigma} - \mathsf{I}\lambda \right) \mathsf{v}_1 = 0$$

Lectura recomendada: *Petersen and Pedersen* - "Matrix Cookbook".

Principal Components Analysis

Algunas cuentas

$$J(\mathrm{v}_1) = \mathrm{v}_1^T \boldsymbol{\Sigma} \mathrm{v}_1 - \lambda \left(\mathrm{v}_1^T \mathrm{v}_1 - 1 \right)$$

$$\nabla J(\mathrm{v}_1) = 2 \left(\boldsymbol{\Sigma} - \mathrm{I}\lambda \right) \mathrm{v}_1 = 0$$

$$\boldsymbol{\Sigma} \mathrm{v}_1 = \lambda \mathrm{v}_1 \quad \rightarrow \quad \mathrm{v}_1 \text{ es AVE de } \boldsymbol{\Sigma} \text{ y } \lambda \text{ es AVA}$$

Lectura recomendada: *Petersen and Pedersen* - "Matrix Cookbook".

Principal Components Analysis

Algunas cuentas

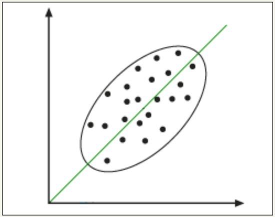
$$J(\mathrm{v}_1) = \mathrm{v}_1^T \boldsymbol{\Sigma} \mathrm{v}_1 - \lambda \left(\mathrm{v}_1^T \mathrm{v}_1 - 1 \right)$$

$$\nabla J(\mathrm{v}_1) = 2 \left(\boldsymbol{\Sigma} - \mathrm{I}\lambda \right) \mathrm{v}_1 = 0$$

$$\boldsymbol{\Sigma} \mathrm{v}_1 = \lambda \mathrm{v}_1 \quad \rightarrow \quad \mathrm{v}_1 \text{ es AVE de } \boldsymbol{\Sigma} \text{ y } \lambda \text{ es AVA}$$

El problema de optimización pasa a ser de la forma

$$\max_{\substack{\mathrm{v}_1: \\ \|\mathrm{v}_1\|^2 = 1}} \mathrm{v}_1^T \boldsymbol{\Sigma} \mathrm{v}_1 = \max_{\substack{\mathrm{v}_1: \\ \|\mathrm{v}_1\|^2 = 1}} \lambda(\mathrm{v}_1) \quad \rightarrow \text{ Máximo AVA}$$



Lectura recomendada: *Petersen and Pedersen* - "Matrix Cookbook".

Principal Components Analysis

Reducción y Reconstrucción

Sobre los autovalores

El porcentaje de energía perdida puede medirse por la proporción de autovalores despreciados.

- V: Matriz de autovectores más relevantes.
- x: Variable de entrada a procesar (ya normalizada).
- u: Representación reducida.
- $\hat{x}$: Reconstrucción

$$u = V \cdot x, \qquad \hat{x} = V^T \cdot u$$