Data Mining

Data dimensionality

Santiago Alonso-Díaz

Tecnólogico de Monterrey
EGADE, Business School

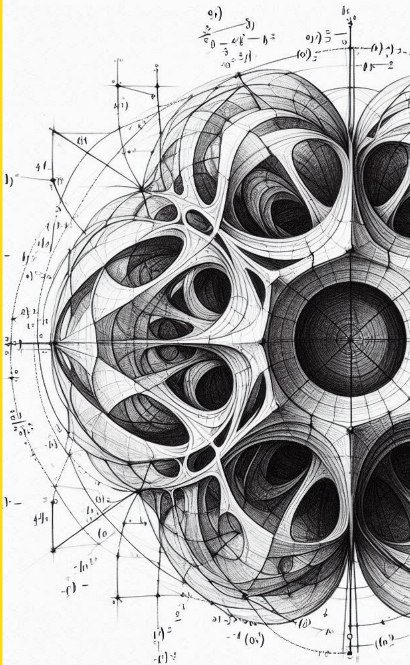


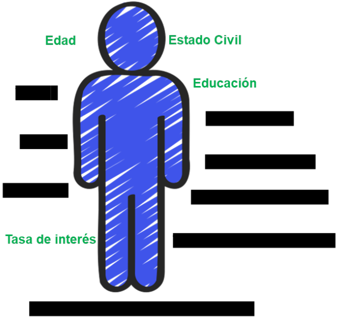
Photo: Dalle2

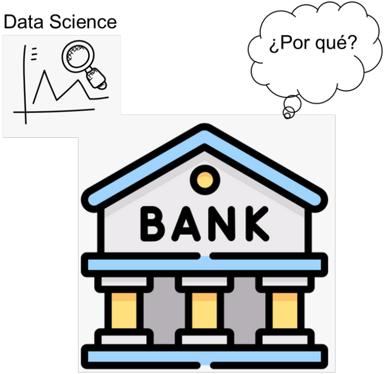
Dimensions are variables

Clients have many dimensions

Dimensions are variables

Option 1: drop variables (e.g. Causality, Lasso)





Dimensions are variables

Option 2: compress/reduce variables (e.g. PCA, clustering)

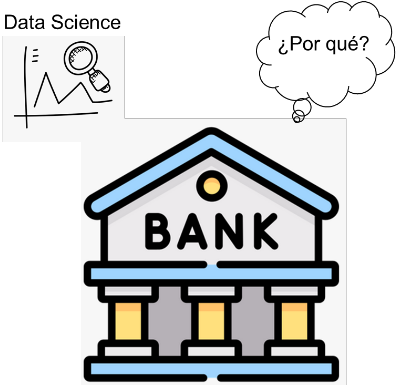


Table of contents

Compressing Variables (Unsupervised)

What is compressing?

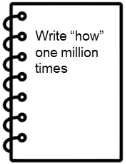
Some times we want to keep all the information but in a more compact format.

Imagine you have a weird book that has the word "how" one million times. You could compress it to a single line and someone else could still write/read it.

The “how” book

A thick book with “how” written many times

The “how” book compressed



How to write the “how” book

Figure: Some things can be compressed

What is compressing?

In business compressing is very useful.

You could have 45 variables from a client.

However, we cannot process in parallel such many variables. We need a more compact representation.

Let's see some unsupervised techniques.

Clustering

Our unclustered data may look like this. Many points/clients:

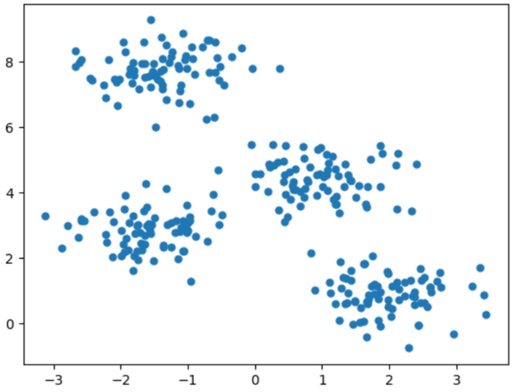


Figure: We want to assign each point to a cluster.

Clustering

Our clustered data now has 4 centers or "clients":

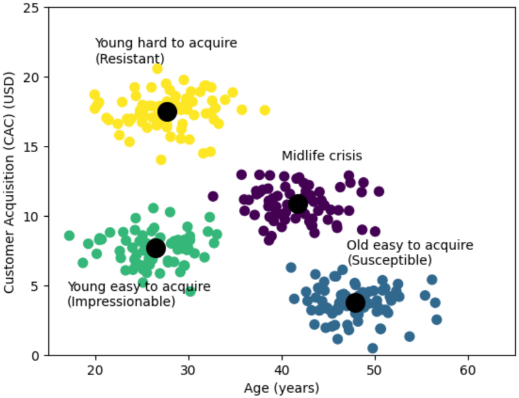


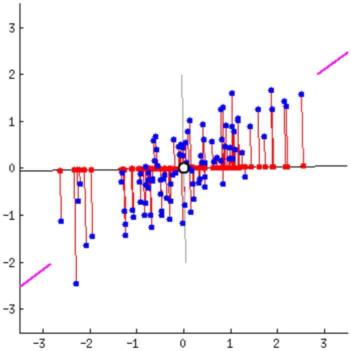
Figure: Clustered and labelled.

Let’s go to python

Clustering.ipynb

PCA intuition

PCA finds the perpendicular axes that point in the directions of maximum spread



PCA

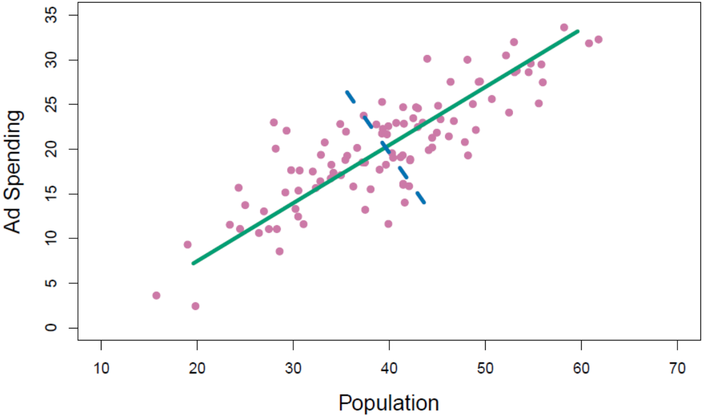


Figure: Example with two dims. We could decide to compress the data with the first component (green): it is loaded with both dimensions. (James et al., 2023)

Dimension reduction via PCA

Each component is orthogonal, captures the largest possible variance (spread of points), and minimizes the mean squared error.

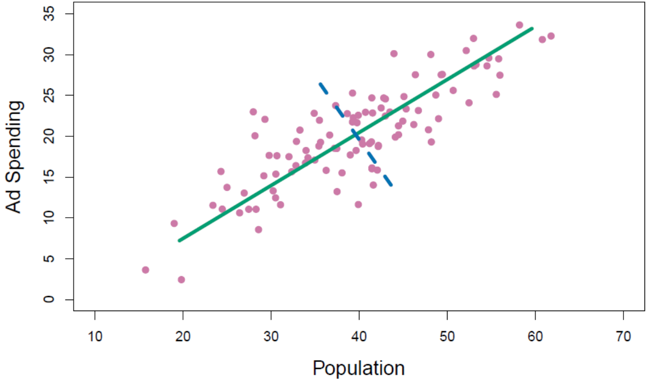


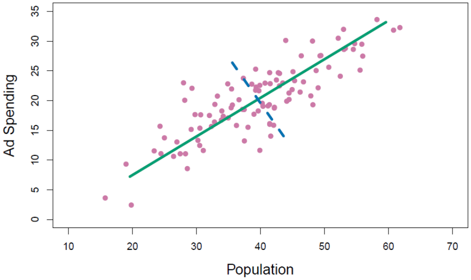
Figure: Example with two dims (James et al., 2023)

Dimension reduction via PCA

In formula, the first component Z_1 is:

$$Z_1 = 0.839 \times (pop_i - \bar{pop}) + 0.544(ad_i - \bar{ad})$$

We refer to 0.839 and 0.544 as loadings. Note that we compress two dimensions to one Z_1. Important to center the variables (e.g. z-scores).



1st component

The first component is interesting because it has an additional interpretation (additional to capturing the most variance):

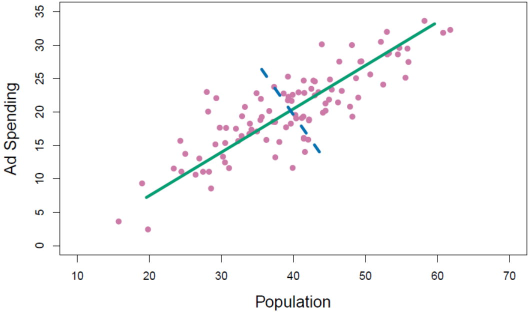
It is the closest to the data

For instance, if $Z_1 < 0$, that combination of population and ad expenses is below average

More components

We can do PCA with n variables in the data, and have up to n components. With two variables, we can estimate a second component (and no more):

$$Z_2 = 0.544 \times (pop_i - \bar{pop}) - 0.839(ad_i - \bar{ad})$$



Bonus: procedure

How we calculate the weights for the linear combinations for each component?
Linear algebra!

- Standardize the variables (e.g. z-scores)
- Calculate covariance matrix (CM)
- Rotate the data by finding eigenvectors (EVc) and eigenvalues (EVa) of the covariance matrix (CM).

$$CM \times EVc = EVa \times EVc$$

- It turns out that the eigenvectors provides the direction and the eigenvalues the spread of the data.

Issues with PCA

- PCA is not feature selection. Each component is a linear sum of ALL features.
- Identification of different geometries (see figure)

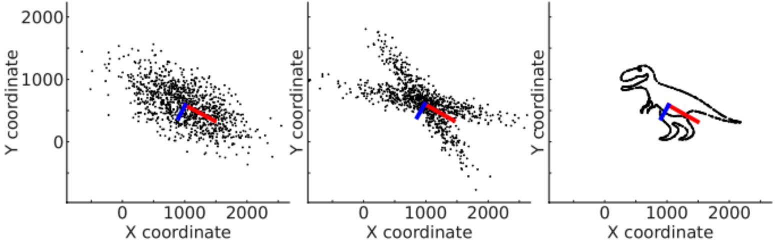


Figure: Similar PCA, different geometries (Dyer & Kording, 2023)

Removing variables (Supervised)

Regularization: general intuition

Figure: You want a party (model) with people (parameters) but not too many, there is a sweet spot.

Regularization: general intuition

Penalize number of parameters in the cost function.
The original RSS (to minimize):

$$RSS = \sum_{i=1}^{n} \left(y_i - \left(\beta_0 + \sum_{j=1}^{p} \beta_j x_{ij} \right) \right)^2$$

The intuition of regularization is this:

$$RSS_{regularized} = RSS + \text{penalty per each } \beta$$

Note that each additional β hinders the minimization i.e. it would be good that some β go to zero.

Lasso regression

The penalty in lasso is the absolute value:

$$\text{RSS}_{\text{Lasso}} = RSS + \lambda \sum_{j=1}^{p} |\beta_j|$$

Note that we use ℓ_1 i.e. absolute value. λ is a free parameter obtainable via cross-validation.

βs to zero

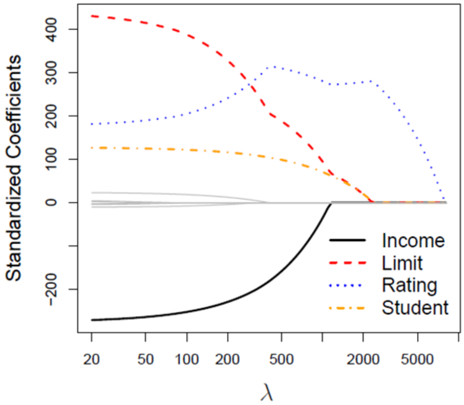


Figure: Credit Data. DV: default. With a large λ, coeff. disappear i.e. feature selection (James et al., 2023)

In python

Let's go to Python

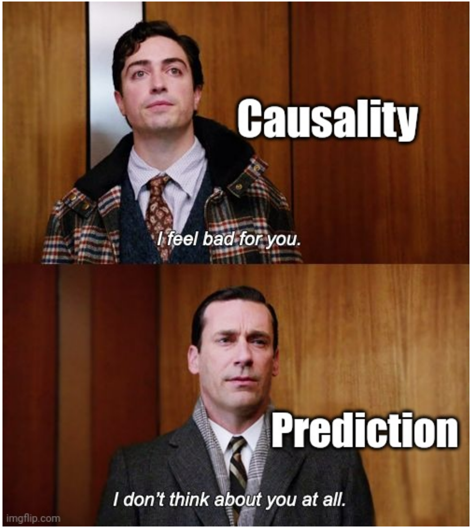
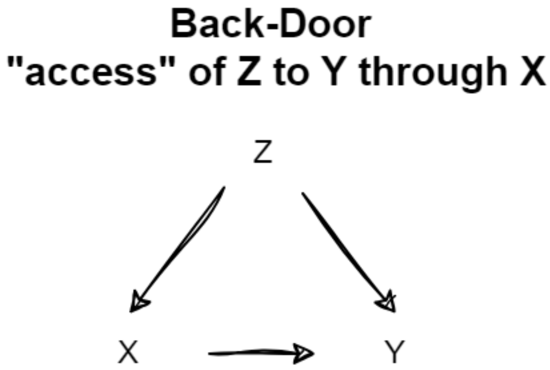


Figure: Source: Khoa Vu Twitter

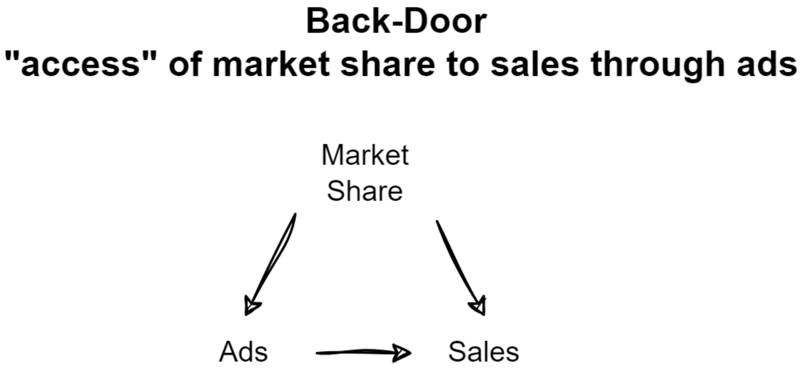
Back-door

Any path from X to Y that starts with an arrow pointing into X. Keep Z in the reg.



Back-door

Any path from X to Y that starts with an arrow pointing into X. Keep Mkt. Shr. in the reg.

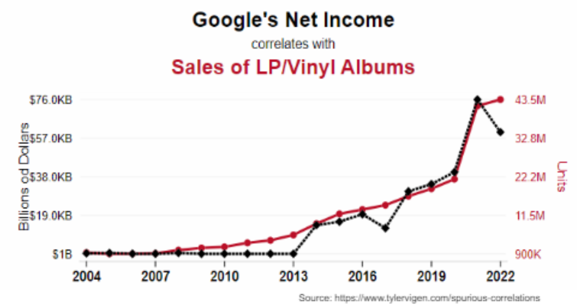


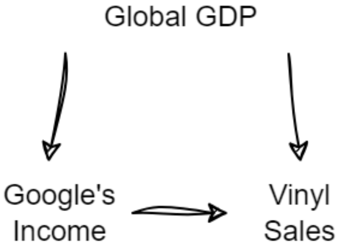
Back-door

Any path from X to Y that starts with an arrow pointing into X. You could drop Mkt. Shr. (unless you want the pure/direct effect of ads)

Problems with back-doors

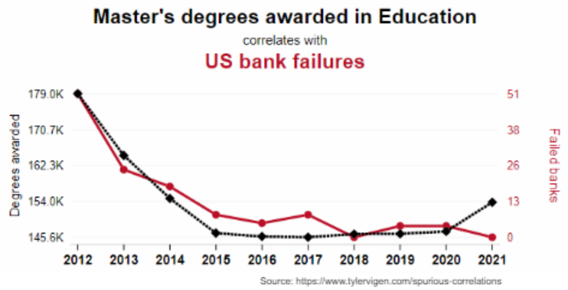
If not accounted, they enhance spurious correlations.

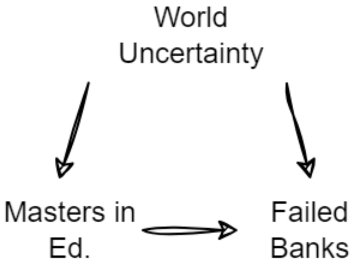




Problems with back-doors

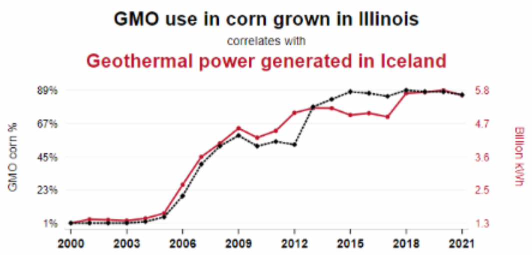
If not accounted, they enhance spurious correlations.



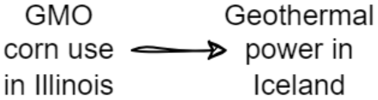


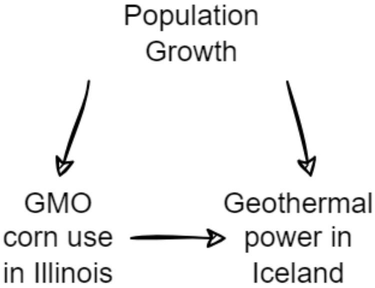
Problems with back-doors

If not accounted, they enhance spurious correlations.

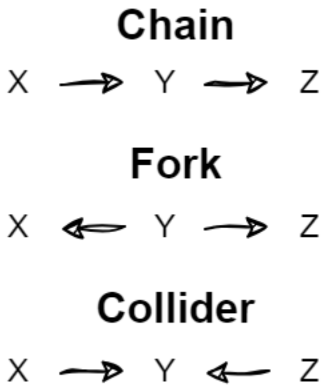


Source: https://www.tylervigen.com/spurious-correlations





Junctions



Junctions: Chain

A mechanism (need) mediates the relation with x (ad) and y (sales).

For instance, controlling-fixing for needs (mediator) cancels ad effects on sales.
Overcontrol: in this model, the only way to affect sales is through needs.

Junctions: Fork

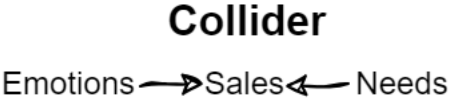
A common cause (need) explains two downstream variables (emotions and sales)

For instance, controlling-fixing for needs (confounder) cancels any spurious correlation between emotion & sales. In this model, emotions and sales are not connected. If I do not know needs, emotions and sales would look as if related due to the common source (e.g. needs go up, both emotions and sales change).

Junctions: Collider

Two variables (emotions and needs) affect a variable (sales).

For instance, controlling-fixing for sales (collider) connects emotions and needs artificially. For a fixed level of sales, we need to "open" emotions and sales because both cause sales. If I increase emotions, I need to modify needs to obtain that fixed level of sales. This creates an illusion that they are related, but just because we fixed sales.



Issues of controlling-fixing a collider

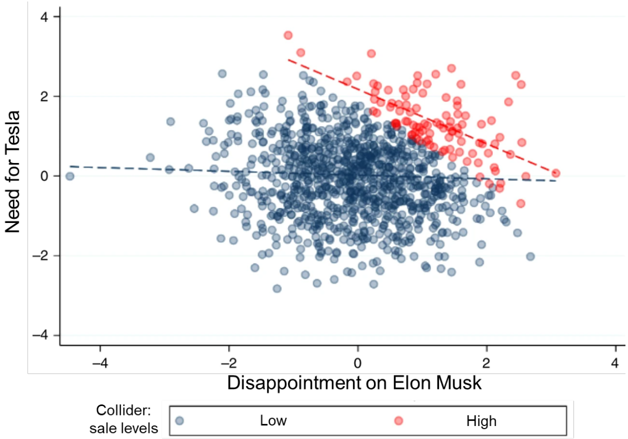


Figure: In loyal clients, needs and emotions are related (red dots). In the general population they are not (red + blue dots). Adapted from Griffith et al., 2020

Issues of controlling-fixing a collider

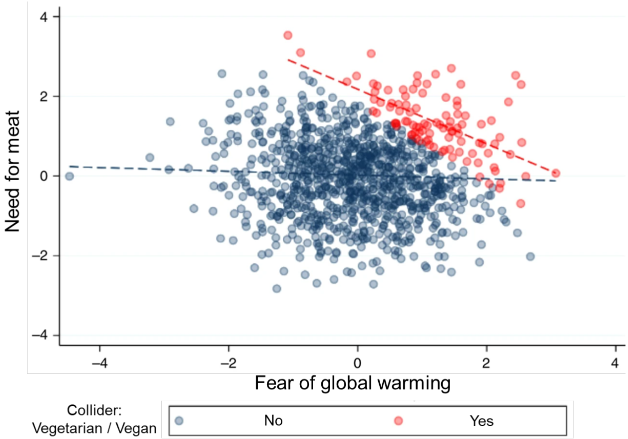


Figure: In vegans, needs and emotions are related (red dots). In the general population they are not (red + blue dots). Adapted from Griffith et al., 2020

When to keep or remove dimensions/variables

- Control-fix-condition on confounders to avoid omitted variable bias.
- Do not control for mediators. This could erase the path (overcontrol bias).
- Do not control for colliders due to overcontrol bias (colliders as mediators), spurious correlations, or could open a backdoor path.

References

Dyer, E. L., & Kording, K. (2023). Why the simplest explanation isn't always the best. *Proceedings of the National Academy of Sciences*, *120*(52), e2319169120.

Griffith, G. J., Morris, T. T., Tudball, M. J., Herbert, A., Mancano, G., Pike, L., Sharp, G. C., Sterne, J., Palmer, T. M., Davey Smith, G., et al. (2020). Collider bias undermines our understanding of covid-19 disease risk and severity. *Nature communications*, *11*(1), 5749.

James, G., Witten, D., Hastie, T., Tibshirani, R., & Taylor, J. (2023). *An introduction to statistical learning: With applications in python*. Springer Nature.