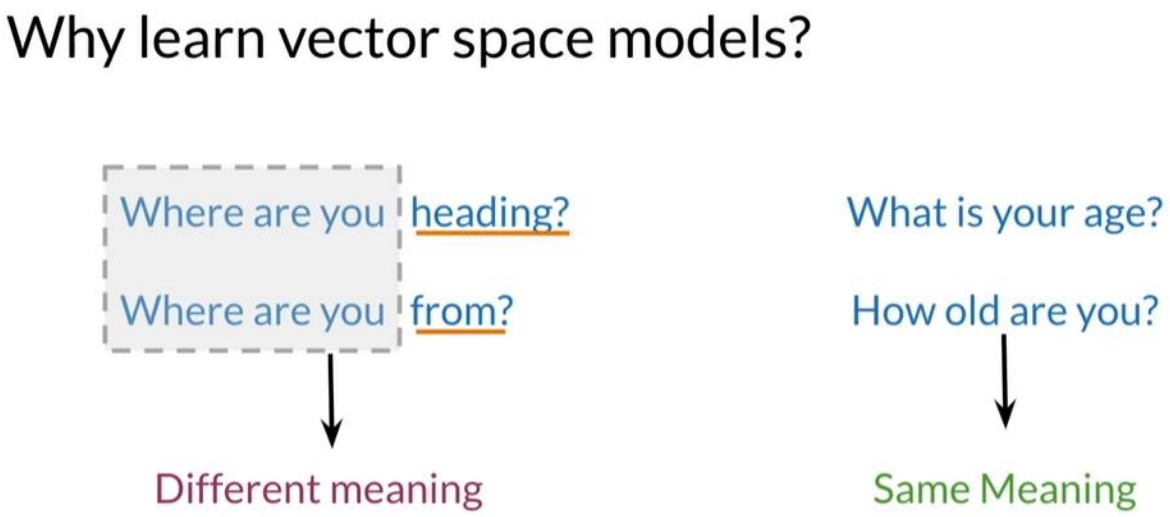
Vector Space Models :

Represent Kata dan document sebagai sebuah vector

So kenapa Vector space models ?

- Ya karena dia bisa **Identifikasi** Kesamaan antar kalimat sebagai contoh :



Kedua contoh diatas bisa dihitung “Similarty” antar Kalimat sehingga kita bisa menentukan whether kalimat tersebut “Similar” atau ndak dalam suatu konteks, bisa saja Kesimpulan katanya sama Cuma kumpulan katanya berbeda, kek contoh ae .

- Dapat mengindentifikasi Dependensi antar kata.

Aplikasinya :

Information Extraction

Machine Translation

Chat bot

Dll

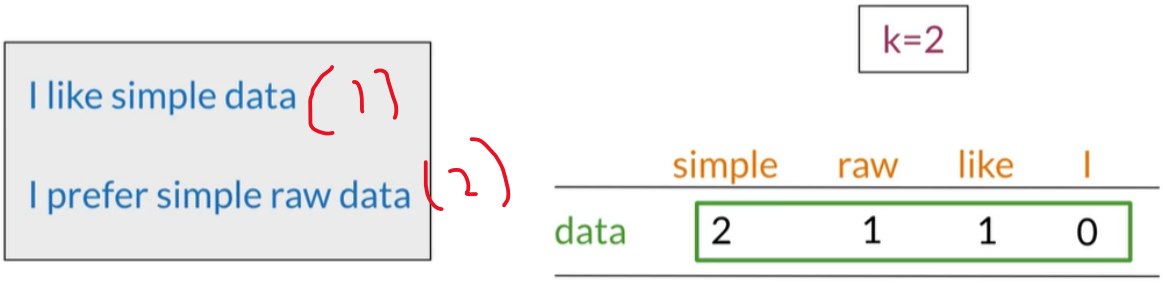
Word By Word and Word by Doc

Co-occurrence → Vector Representation

Co-occurrence Matrix → Matrix Representation

Word by Word Design → Banyaknya kedua kata tersebut muncul dalam jarak **K.**

Contoh



Oke gimana ini bacanya ?

So coba liat kata **data** ,jarak kata **data** dengan kata **simple** itu paling besar **2** dari pada setiap kalimat yang ada, kalimat 1 jarakne Cuma 1, sedangkan kalimat 2 jarakny adalah 2,"Simple raw data". Untuk yang jaraknya lebih dari 2, si **I** ya jarak akhirnya adalah 0, karena dia beneran jauh, ini btuh bantuan pak Adit biar lebih paham konsepnya ada beberapa pertanyaaan yang perlu di tanya ,ehe..

Word by document design → Total banyaknya sebuah **kata** muncul di sebuah kategori

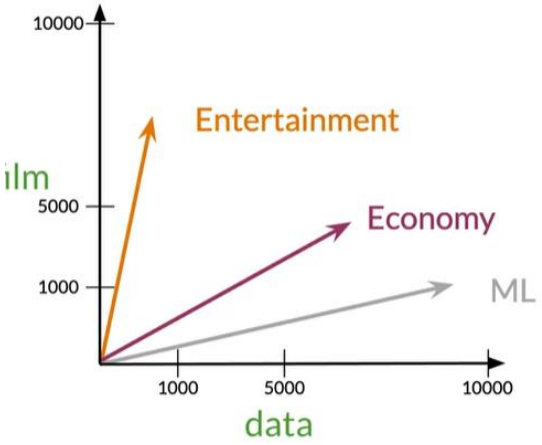
Word by Document Design

Number of times a word *occurs within a certain category*

	Entertainment	Economy	Machine Learning
data	500	6620	9320
film	7000	4000	1000

Setelah kita membuat co-occurrence matrix, selanjutnya adalah merepresentasikannya sebagai vector space yang dapat menentukan kesamaan antar kata/dokumen.

Vector Space



	Entertainment	Economy	ML
data	500	6620	9320
film	7000	4000	1000

Measures of "similarity:"
Angle
Distance

Nah selanjutnya kita bakalan compute " Similarty" Angle dan distance dari design yang kita buat, **word by word, word by doc**

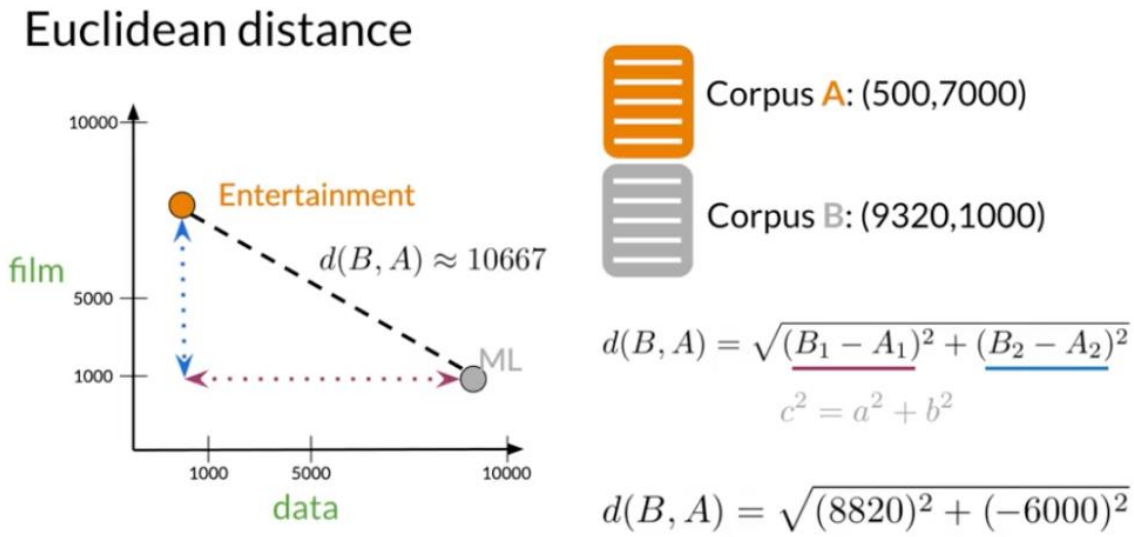
Euclidian Distance → Ngitung jarak antara 2 titik, ini rumus ne

$$d(B, A) = \sqrt{((B_1 - A_1)^2 + (B_2 - A_2)^2)}$$

Itu B_1 .. B_n Total Feature ya, disini Cuma ada 2 fitur aja

Kalau n Feature,rumusnya kek gini

$$d(\vec{v}, \vec{w}) = \sqrt{\sum_{i=1}^{n}(v_i - w_i)^2}$$



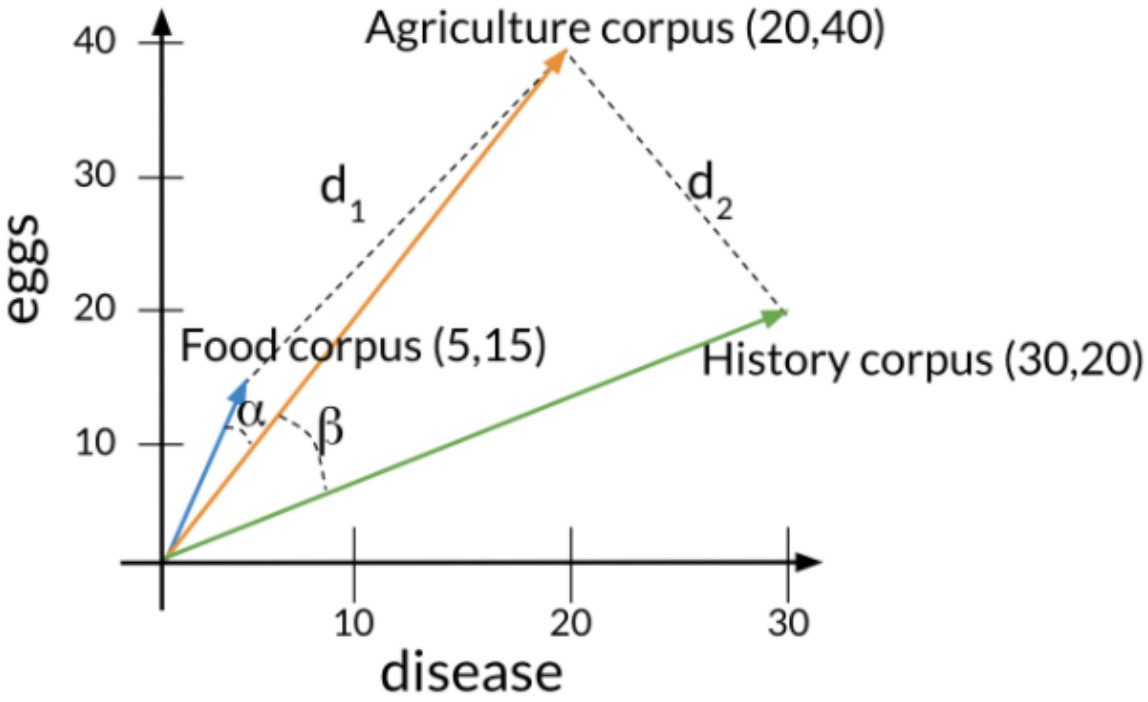
Intinya kalkulasi aja, jadi disini kita bisa simpulkan bahwasanya Terdapat **perbedaan** signifikan pada corpus A dengan Corpus B. Dengan artian Corpus A membahas suatu hal yang berbeda di banding corpus B, ez

	data	$\vec{w}$ boba	$\vec{v}$ ice-cream
AI	6	0	1
drinks	0	4	6
food	0	6	8

$$= \sqrt{(1-0)^2 + (6-4)^2 + (8-6)^2}$$
$$= \sqrt{1+4+4} = \sqrt{9} = 3$$

Gambar diatas contoh aja ngab, itu contoh seberapa similar boba dengan ice cream.

Tetapi Euclidean Distance punya permaslahan, yaitu **belum akurat dalam menghitung kesamaan antar 2 vector**



Kita bisa liat kalau misalnya d2 < d1 , nih kalkulasinya

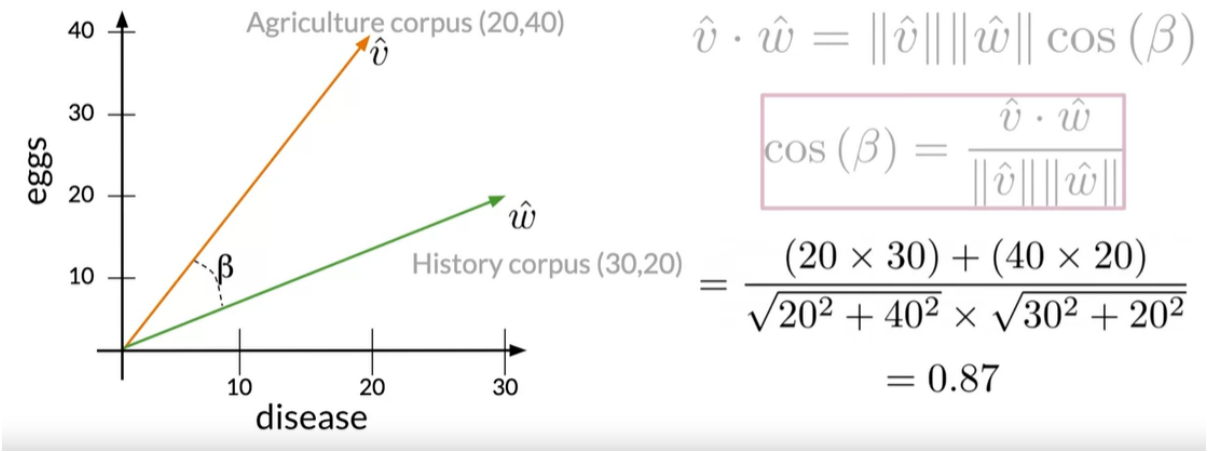
D1(ac,fc) = 40.6201

D2(ac,hc) = 30

Thus Agriculture corpus more similar with history corpus, LOL,this is absolute problem, instead the answer could be food corpus and agriculture having more similarity. Nah solusinya denga cosine similarity, jadi dia ngitung angle diantara 2 titik, semakin menyempit , maka semakin mirip. Coba liat angel beta dengan alpha , hasilnya

: $\beta > \alpha$ **,** yang mana harusnya agriculture corpus lebih mirip dengan food corpus dibandingkan dengan history. **Dalam kasus ini sebenarnya data yang dimiliki oleh food corpus berbeda jauh dengan food, ada baiknya Ketika memiliki data set seperti ini gunakan cosine similarity.** Karena apabila membandikan large document dengan smaller ones, **Euclidean distance tidak akan akurat.**

Ini contoh ngitungnya



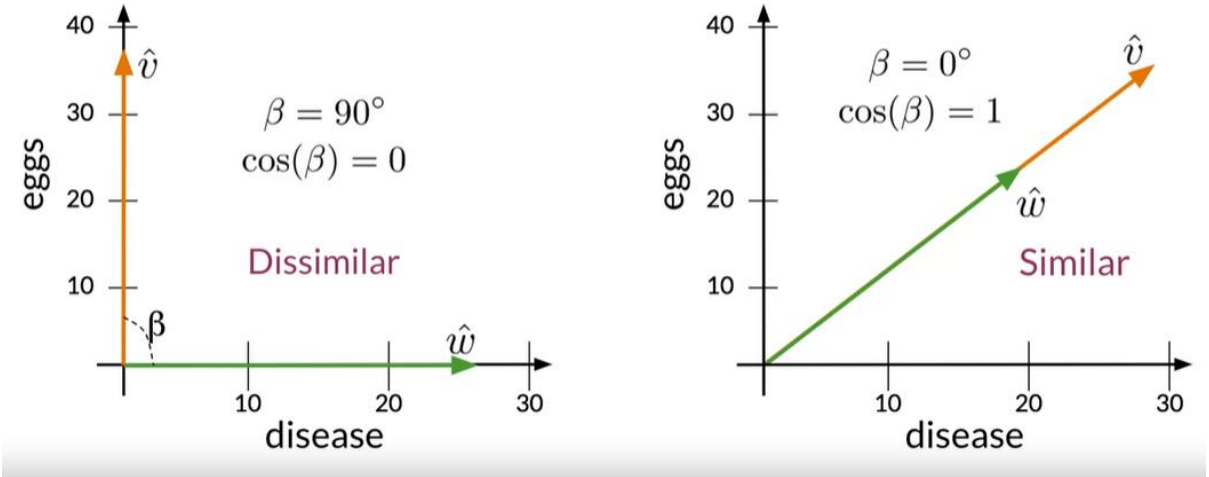
$$\hat{v} \cdot \hat{w} = \|\hat{v}\| \|\hat{w}\| \cos(\beta)$$

$$\cos(\beta) = \frac{\hat{v} \cdot \hat{w}}{\|\hat{v}\| \|\hat{w}\|}$$

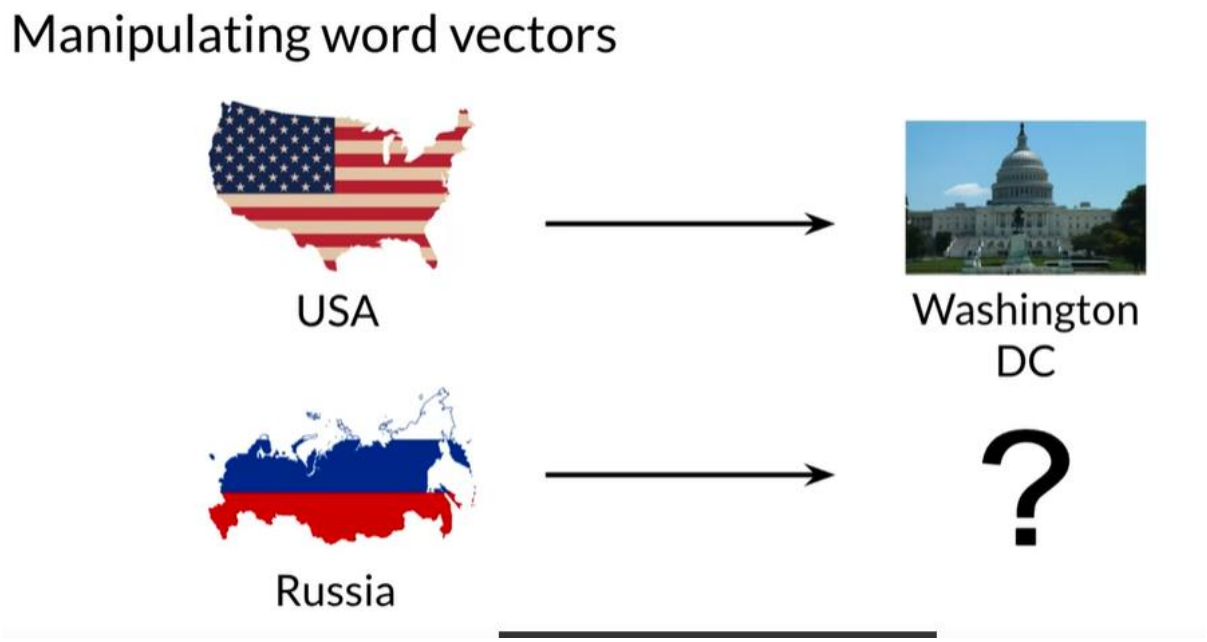
$$= \frac{(20 \times 30) + (40 \times 20)}{\sqrt{20^2 + 40^2} \times \sqrt{30^2 + 20^2}}$$

$$= 0.87$$

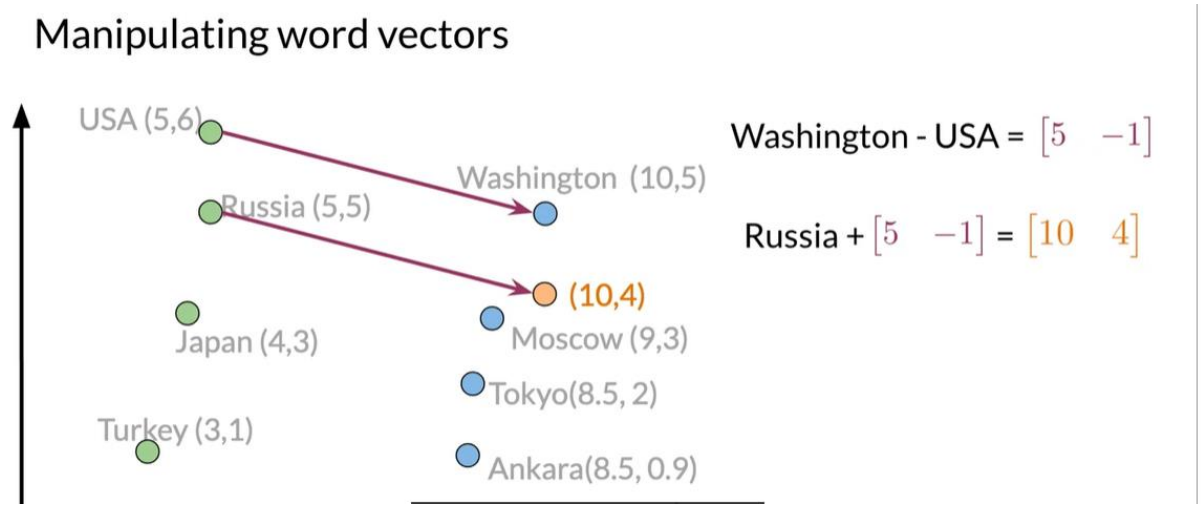
Diharapkan dah pernah kalkulasi Norm dkk, biar tau maksud kalkulasinya. Gw belum tau teknikal matematiknya gimans, harus tau asal muasal rumusnya, ehe pelajarin diri aja. So mungkin ada yang bertanya, kok angle bisa ngitung similarity? Nih jawabannya



Sekarang kita ke manipulasi kata di vector spaces. Well Ini sebenarnya berkaitan dengan perhitungan aja sih dari beberapa fitur. Sebagai contoh kita pengen tau ibu kota Rusia cuman kita punya observasi data negara amerika dan ibu kotanya.



Pertama kita coba hitung dahulu vector diff antara observasi amerika dengan ibukotanya, lah kok kek gitu? Yakarena dengan kita ngitung hal teresebut kita bisa tau **Dimensi** ibu kota itu kurang lebih terletak dimana. Setelah kalkulasi Vector diff tadi, selanjutnya jumlah vector representation dari Russia dengan vector diff tadi



Dapet nih dia kemungkinan berlokasi di dimensi (10,4) Yowes tinggal hitung aja dengan **Cosine similarity atau Euclidean distance** untuk

dapetin Nilai yang similar, **INI BISA AJA SALAH ya, tergantung hasil dari pada representasi vector klean.** Oh ya Tujuan dari Manipulating words ini untuk **identify patterns ya,kalau untuk prediksi seperitnya kurang ,YA harus liat datanya :P**

Oke gw jelasin,itu gambar dikiri kenapa disebut dissimilar, yakarena Ketika **Beta** = 90 derajat ,cos(90 derajat) = 0, dan kalau liat dariapada plot terlihat orthogonal kan, indikasi kedua korpora tersebut **dissimilar**. Sedangkan jika **beta = 0** derajat,cos (0) = 1, Sehingga kalau liat plot ya corpora tersebut akan dikatakan mirip, karena anglenya udah pasti 0 derajat, dalam artian di corpora A maupun corpora B memiliki total kata eggs yang sama namun kata disease di salah satu corpora lebih banyak, KALAU MIKIR ADA MINUS< TANPOL > TOTAL KOK MINUS.

Buka lab nya dlu sebelum lanjut,

Intinya sekarang adalah reduksi dimensi, Oke kan tadi liat tuh conto represntasi datanya, untuk kata china sendiri , featuresnya ada sekitaran 300 kan, nah ini tuh kata kata yang telah dijadikan feature kek yang gambar gambar diatas,but kita ga bisa dong plotting **Relationship Sperti** apa dari seluruh feature ini, Thus kita butuh suatu algoritma untuk **Reduksi Dimensi. PCA, Maksud reduksi dimensi adalah mengurangi total fitur yang sebelumnya menjadi lebih kecil dengan tetap mempertahanakan informasi yang esensial.**

So PCA bakalan ngelakuin hal berikut :

	$d > 2$		
oil	0.20	...	0.10
gas	2.10	...	3.40
city	9.30	...	52.1
town	6.20	...	34.3

PCA →

	$d = 2$	
oil	2.30	21.2
gas	1.56	19.3
city	13.4	34.1
town	15.6	29.8

Thus, bakalan mendapatkan Representasi fitur yang lebih kecil BUT punya bobot informasi yang hampir mirip dengan data sebelumnya, WENAK nya coba buktiin PCA di Image aja.

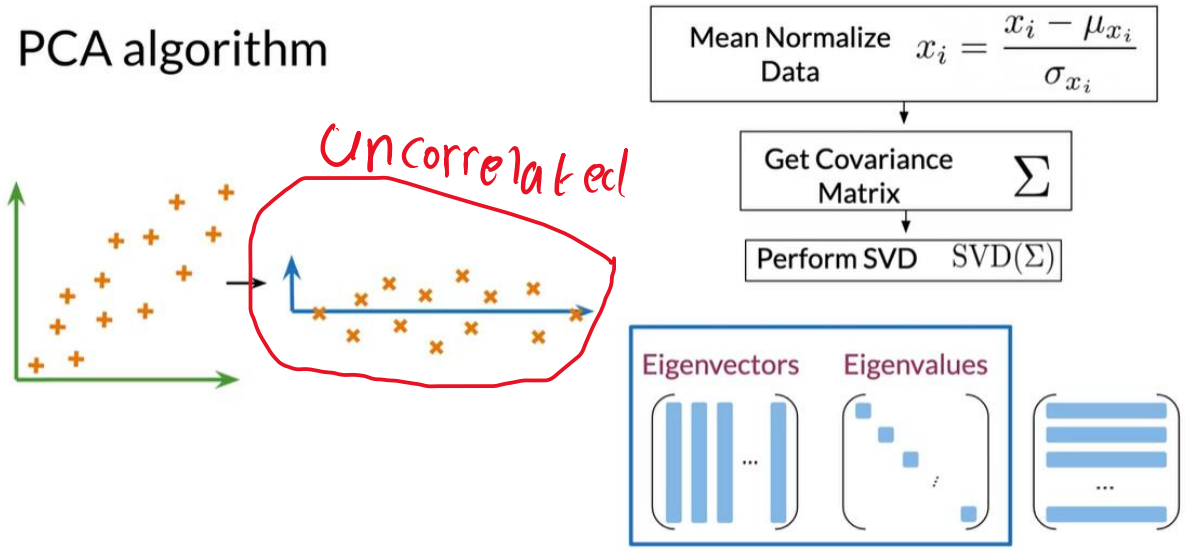
Ada Dua Komponen Penting di PCA

PCA algorithm

Eigenvector: Uncorrelated features for your data

Eigenvalue: the amount of information retained by each feature

Step Pertama Ngambil **Uncorrelated Feature**

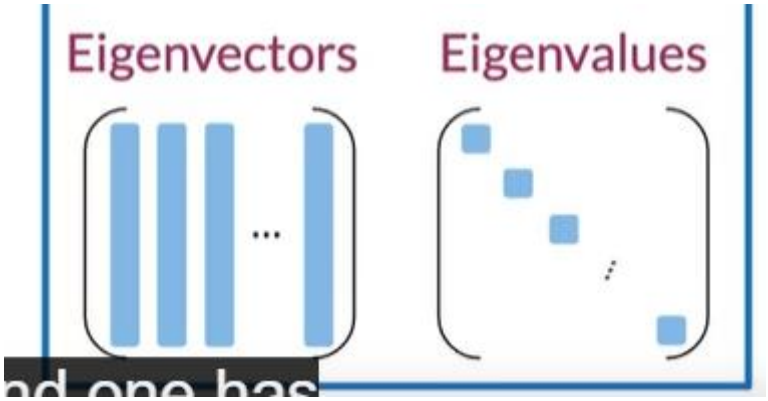


Covariance Matrix itu aps sih boy? And SVD itu apa ? Kita cukup tau apa itu covariance , covariance adalah mengukur sebarapa besar 2 variabel acak bervariasi bersamaan, Intina mah ini 2 variabel kalau dapet covariance nya bakalan diketahui **Arah Slope, Positif atau negatif atau tidak ada sama sekali** diantara 2 variabel tersebut. Untuk Value dari pada covariance sendiri **Sulit** di interpretasikan, dan harus melihat Contextnya, Tonton nih : https://www.youtube.com/watch?v=qtaqvPAeEJY

Dan

https://www.youtube.com/watch?v=xZ_z8KWkhXE

Biar makin paham, itu supplement aja. Sedangkan SVD itu adalah Langkah awal ntuk semua **data reduction** technique. So Ketika kita menghitung SVD kita akan mendapatkan 2 nilai penting untuk PCA, EigenVectors dan eigen values



Setelah mendapatkan kedua nilai ini, selanjutnya melakukan dot product antara Values dari Word embedding, itu lho yang fiturnya tadi sampe 299 , dengan **EigenVectors**, Lalu ambil hasil kalkulasinya , seluruh row dan hanya ambil 2 fitur aja . fungsi melakukan dot product ini ya untuk **projection** terhadap uncorrelated features.

Dot Product to Project Data $X' = XU[:, 0:2]$

Itu slicing nya kan bener, ehe.

Dan yang terakhir baru hitung

Percentage of Retained Variance $\frac{\sum_{i=0}^{1} S_{ii}}{\sum_{j=0}^{d} S_{jj}}$

Yang mana fungsinya untuk mendapatkan **Persenatasi variance yang di pertahanakan di vector spaces baru, i.e informasi yang telah direduksi**. Variance means to find the expected difference of deviation from actual value. Therefore, variance depends on the standard deviation of the given data set.

The more the value of variance, the data is more scattered from its mean and if the value of variance is low or minimum, then it is less scattered from mean. Therefore, it is called a measure of spread of data from mean. Males Translate,maap