# **The use of an algorithm to solve a specific business problem.**

**Definition of an algorithm**

An algorithm is a finite list (set) of instructions, most often used in solving problems or performing tasks basing in various niches. For this case in business world, a company that is built around algorithms that define its processes and deliver customer services is an algorithmic business. Unlike big data, algorithms define action and have real value.   
There are organizations that generate massive amounts of data, analyze them and then leave the decision making to humans. Then there are organizations that use algorithms that operate independent of human intelligence for many aspects of decision making, including pricing and customer service. There are several companies that use algorithms to solve their problem example the Ecommerce websites (Amazon and Alibaba) they use algorithmic business model mainly for pricing, online identity verification, inventory and seller matching. The stated companies use dynamic strategy that allows the retailers to automatically modify their prices according to change that the market experiences. They also take into account clients demands, competitors’ price and they also run large-scale pricing experiments and simulations to make better strategic business decisions.

**Company of focus**: *Amazon*

**Business problem to focus on**: *Need for online identity verification (Face recognition)*

**Problem definition**

When a visitor goes to an ecommerce website and signs up, the ecommerce website needs to somehow be sure that this is a legit person who wants to buy. This way, the company will avoid fraudulent accounts or bots which could result in revenue losses (especially with cash-on-delivery (COD) purchases). Identity verification is a necessary process that ensures a person’s identity matches the one that is supposed to be, that is; the one which was logged in during the first sign up procedure to the website. This process is also kwon as the KYC (Know your customer) process. Hence to curb this issue, a facial recognition algorithm must be embedded in the code this algorithm will be used to identify the person from their image which may be found on ID, Driver’s license or passport which was submitted during sign up process. Some features like eyes, lips, shape, etc. extract from the image or video source to identify the person’s identity.

## **Meaning of verifying identity**

Identity verification ensures that there is a **real person behind a process** and proves that the one **is who he or she claims to be**, preventing both a person from carrying out a process on our behalf without authorization, and creating false identities or commit fraud.

There are **various methods and systems for identity verification services**. The identity verification process can be carried out in different ways, depending on the channel and the way the verification is carried out.

In the **traditional face-to-face process**, an individual **personifies himself** in a company (store, branch, commercial office), institution (tax agency, employment service) or organization and delivers his **identity document** (ID, passport or equivalent) at the same time that the agent can **verify that the photograph corresponds to the person who is delivering the document**. A photocopy of the document is made, a form is filled out and checks are made, being ready to proceed with the pertinent steps by the identified user

**Algorithm of focus**

**Eigenface (PCA)**

Eigenfaces is a face recognition algorithm, which uses principal component analysis (PCA). PCA is a statistical approach that is used for dimensionality reduction. Eigenfaces reduce some less important features from the image and take only important and necessary features of the image. Eigenfaces reduce dimensionality with having important features of the image. When dimensionality reduced then quality and space occupied will also reduce because of losing information (less important).

**History of Eigenface**

The eigenface approach began with a search for a low-dimensional representation of face images. **Sirovich** and **Kirby (1987)** showed that [principal component analysis](https://en.wikipedia.org/wiki/Principal_component_analysis) could be used on a collection of face images to form a set of basis features.

In 1991, Turk and Pentland suggested an approach to face recognition that uses dimensionality reduction and linear algebra concepts to recognize faces. This approach is computationally less expensive and easy to implement and thus used in various applications at that time such as handwritten recognition, lip-reading, medical image analysis, etc.  
PCA (Principal Component Analysis) is a dimensionality reduction technique that was proposed by Pearson in 1901. It uses Eigenvalues and EigenVectors to reduce dimensionality and project a training sample/data on small feature space. Let’s look at the algorithm in more detail (in a face recognition perspective). These basis images, known as eigen pictures, could be linearly combined to

reconstruct images in the original training set. If the training set consists of *M* images, principal component analysis could form a basis set of *N* images, where *N < M*. The reconstruction error is reduced by increasing the number of eigen pictures; however, the number needed is always chosen less than *M*. For example, if you need to generate a number of *N* eigenfaces for a training set of *M* face images, you can say that each face image can be made up of "proportions" of all the *K* "features" or eigenfaces: Face image1 = (23% of E1) + (2% of E2) + (51% of E3) + ... + (1% En  )

**How it works**

First, an input image from the database is taken. a large dataset for training for getting a more accurate result. After taking the image, the image is classified using an image classifier thereafter, a single layer neural networks in our classifier. In this classifier, a 2D image to vector is made (If image size is pxq then we make a vector of pqx1). Then we are doing the process of feature extraction.

**Algorithm:**

In Eigenfaces we are using Principal Component Analysis . From 2D image we get to 1D vector it called feature vector.

PCA performs as given below

Let P={P1,P2,…,Pn} is random vector.

1. Find mean

Face Recognition Algorithms

1. Find Covariance Matrix

Find Covariance Matrix

1. Find eigenvector λi and eigenvalue bi of S : Sbi= λibi ,i=1,2,…,n
2. Sort all eigenvectors by their value and select the first k largest eigenvectors. So the new observed vector is given by

v = WT(P – m)

where W = {b1,b2,…,bk}

From using PCA we get eigenvector and dimensionality smaller that original dimensionality (pqx1) From classifier we get if the image is matching or not.

The potential application area of face recorgnition technology can be outlined as follows:

A automated surveillance, where the objective is to recognize and track people

Monitoring closed circuit television (CCTV), the facial recognition capability can be embedded into existing CCTV networks, to look for lost children or other missing persons or tracking known or suspected wanted criminals.

Image database investigations, searching image databases of licensed drivers, benefit recipients and finding people in large news photograph and video collections, as well as searching in the Facebook social networking web site.

Multimedia environments with adaptive human computer interfaces (part of ubiquitous or context aware systems, behavior monitoring at childcare or centers for old people, recognizing customers and assessing their needs)

Airplane‐boarding gate, the face recognition may be used in places of random checks merely to screen passengers for further investigation. Similarly, in casinos, where strategic design of betting floors that incorporates cameras at face height with good lighting could be used not only to scan faces for identification purposes, but possibly to afford the capture of images to build a comprehensive gallery for future watch‐list, identification and authentication tasks

**Pros and cons**

**Pros**

**Removes Correlated Features**

**With this kind of algorithm, it is easier and efficient to find correlation among the features since finding it manually in thousands of features is nearly impossible**

**Reduces Overfitting**

Overfitting mainly occurs when there are too many variables in the dataset. So, this kind of algorithm helps in overcoming the overfitting issue by reducing the number of features.

**Improves Visualization:**

It is very hard to visualize and understand the data in high dimensions. This type of algorithm transforms a high dimensional data to low dimensional data (2 dimension) so that it can be visualized easily.

We can use 2D Scree Plot to see which Principal Components result in high variance and have more impact as compared to other Principal Components.

Even the simplest IRIS dataset is 4 dimensional which is hard to visualize. We can apply this kind of algorithm to reduce it to 2-dimension for better visualization.

**Cons**

**Independent variables become less interpretable:**

After implementing, your original features will turn into Principal Components. Principal Components are the linear combination of your original features. Principal Components are not as readable and interpretable as original features

**Data standardization is must before Principal Component Analysis**

You must standardize your data before implementing PCA, otherwise PCA will not be able to find the optimal Principal Components.

For instance, if a feature set has data expressed in units of Kilograms, Light years, or Millions, the variance scale is huge in the training set. If PCA is applied on such a feature set, the resultant loadings for features with high variance will also be large. Hence, principal components will be biased towards features with high variance, leading to false results.

Also, for standardization, all the categorical features are required to be converted into numerical features before PCA can be applied.

**Information Loss:**

Although Principal Components try to cover maximum variance among the features in a dataset, if we don’t select the number of Principal Components with care, it may miss some information as compared to the original list of features

**Aspect of business addressed by this kind of Algorithm**

**Customer service**

With this kind of algorithm, the business is able to distinguish original clients from fake clients hence adjusting the way of treating customers/clients with given priorities.

**Record keeping**

With this kind of algorithm, record keeping which is the biggest aspect of any given business is achieved since records with verified identities are given much priorities hence making the business to run smoothly and also helps in record manipulation at any given scenario

**Alternative of PCA**

**The following is the other alternative of PCA**

**Factor analysis**

**Factor analysis** Is typically used to identify or confirm the latent factor structure for a group of measured variables. Latent factors are unobserved variables which typically can not be directly measured; but they are assumed to cause the scores we observe on the measured or indicator variables. FA is also used to reduce the number of variables which can reasonably measure or convey the latent factor structure. FA is a model-based technique. It is concerned with modelling the relationships between measured variables, latent factors, and error. Therefore, because of the recognition of error; FA is typically more consistent across samples (i.e., the results tend to be more generalizable and replicable than PCA). The ability of f actor analysis to recognize unique item variance (sometimes referred to as item error variance) is a key in the distinguishing it from PCA – which considers all variance equally and attempts to account for as much of it as possible without regard to types of variances. FA relies on assumptions of linearity, multivariate normality, and homoscedasticity

**Pros and cons of Factor analysis**

**Pros**

The advantages of factor analysis are as follows: Identification of groups of inter-related variables, to see how they are related to each other. Factor analysis can be used to identify the hidden dimensions or constructs which may or may not be apparent from direct analysis. Both objective and subjective attributes can be used. Reduction of number of variables, by combining two or more variables into a single factor. There is flexibility in naming using dimensions. It is not extremely difficult to do, inexpensive, and accurate.

**Cons**

Its usefulness depends on the researchers' ability to develop a complete and accurate set of product attributes - If important attributes are missed the value of the procedure is reduced accordingly. Also to point out that Naming of the **factors** can be difficult - multiple attributes can be highly correlated with no apparent reason

**Appendix**

**Reference:** Wikipedia, geek for geeks, stack exchange GitHub