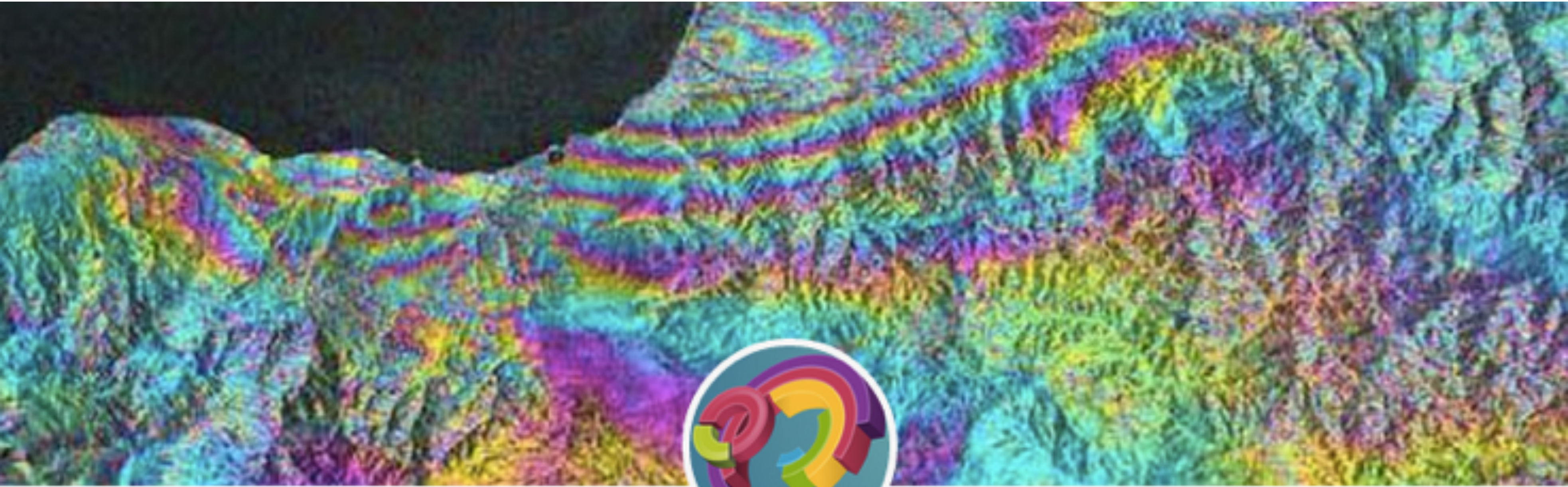
GENERAL ASSEMBLY

INTRO TO DATA SCIENCE
LECTURE 6: DIMENSIONALITY REDUCTION

Francesco Mosconi
DAT10 SF // October 22, 2014

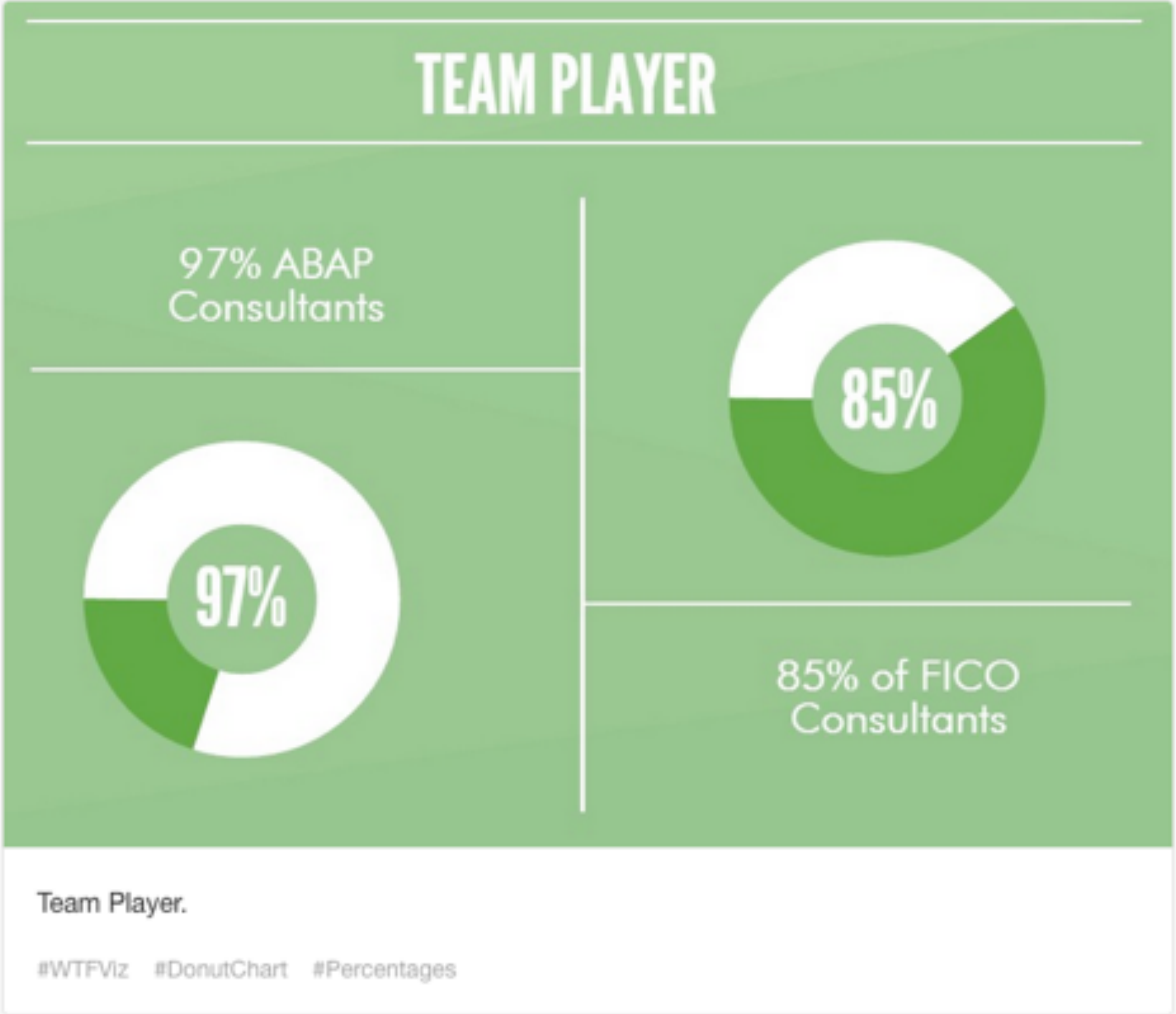
DATA SCIENCE IN THE NEWS



WTF Visualizations

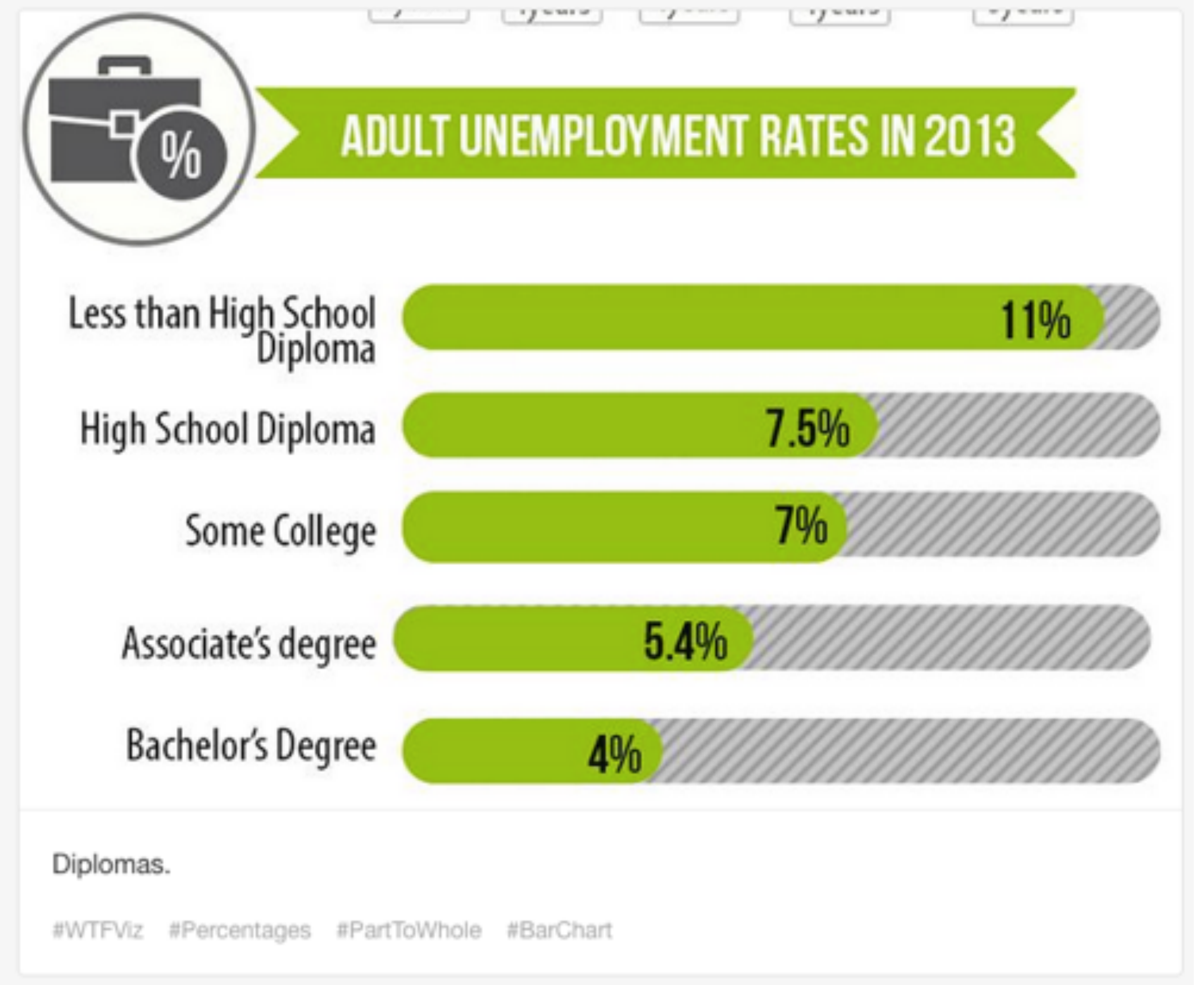
Visualizations that make no sense.
For a discussion of what is wrong with a particular visualization, tweet at us @WTFViz.
Check out our friends Thumbs Up Viz and accidental aRt, or submit.

DATA SCIENCE IN THE NEWS



source: http://wtfviz.net/

DATA SCIENCE IN THE NEWS



source: http://wtfviz.net/

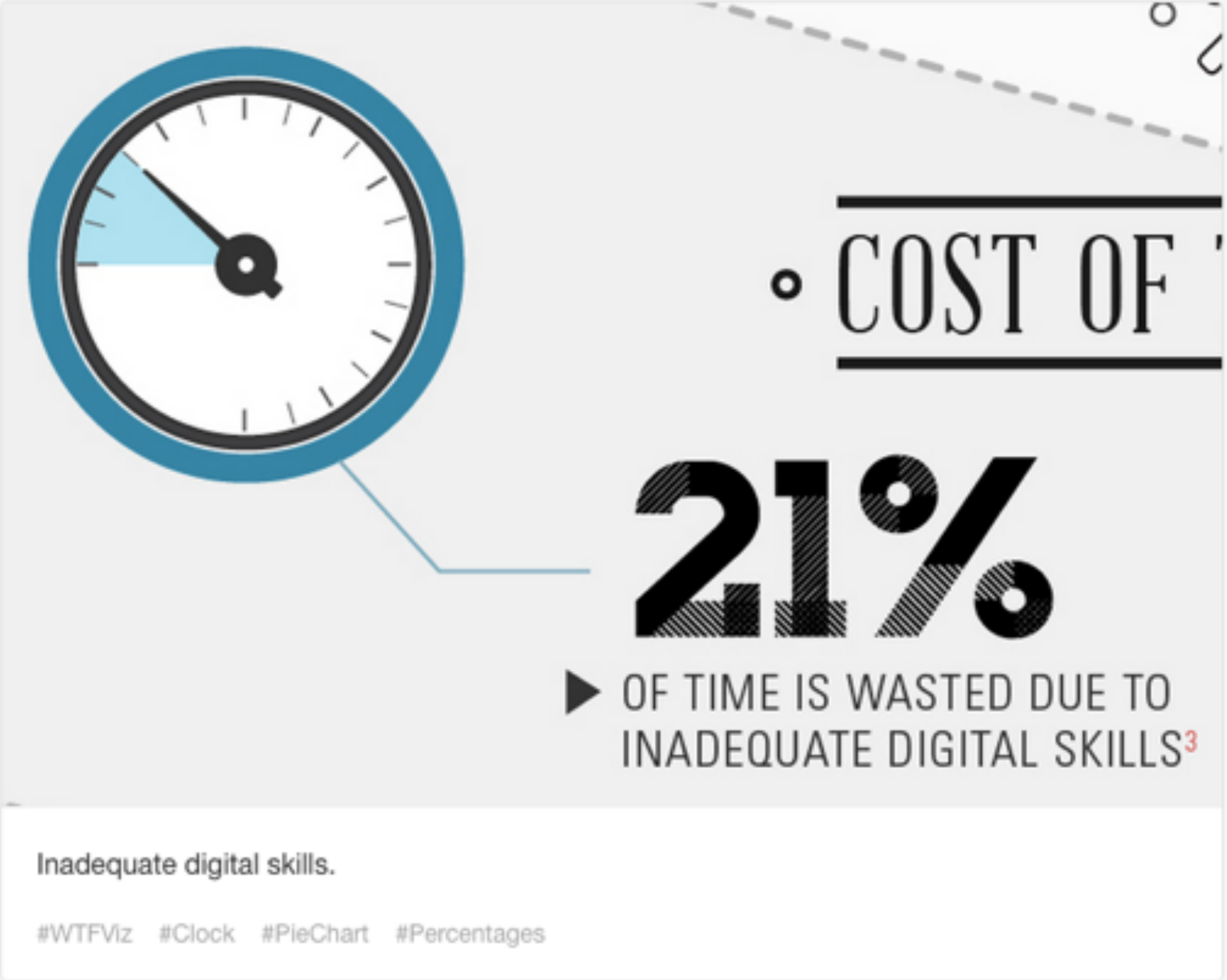
DATA SCIENCE IN THE NEWS

Core skills.

#WTFViz #Radial #BarChart #LabelLines

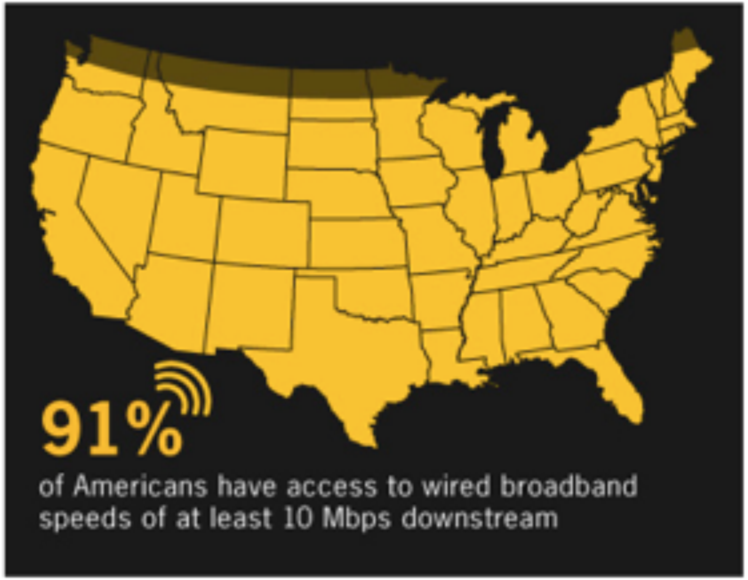
source: http://wtfviz.net/

DATA SCIENCE IN THE NEWS



source: http://wtfviz.net/

DATA SCIENCE IN THE NEWS



Northern regions.

#WTFViz #Map #Percentages

source: http://wtfviz.net/

http://wtfviz.net/

LAST TIME:

- LINEAR REGRESSION (INCL. MULTIPLE REGRESSION)
- POLYNOMIAL REGRESSION
- REGULARIZATION

QUESTIONS?

I. DIMENSIONALITY REDUCTION
II. PRINCIPAL COMPONENTS ANALYSIS

EXERCISE:
III. REGRESSION EXERCISE

I. DIMENSIONALITY REDUCTION

Q: What is dimensionality reduction?

Q: What is dimensionality reduction?

A: A set of techniques for reducing the size (in terms of features, records, and/or bytes) of the dataset under examination.

Q: What is dimensionality reduction?
A: A set of techniques for reducing the size (in terms of features, records, and/or bytes) of the dataset under examination.

In general, the idea is to regard the dataset as a matrix and to decompose the matrix into simpler, meaningful pieces.

Q: What is dimensionality reduction?
A: A set of techniques for reducing the size (in terms of features, records, and/or bytes) of the dataset under examination.

In general, the idea is to regard the dataset as a matrix and to decompose the matrix into simpler, meaningful pieces.

Dimensionality reduction is frequently performed as a pre-processing step before another learning algorithm is applied.

	Continuous	Categorical
Supervised	???	???
Unsupervised	???	???

	Continuous	Categorical
Supervised	regression	classification
Unsupervised	dimension reduction	clustering

Q: What are the motivations for dimensionality reduction?

Q: What are the motivations for dimensionality reduction?

The number of features in our dataset can be difficult to manage, or even misleading (e.g., if the relationships are actually simpler than they appear).

For example, suppose we have a dataset with some features that are related to each other.

For example, suppose we have a dataset with some features that are related to each other.

Ideally, we would like to eliminate this redundancy and consolidate the number of variables we're looking at.

To say this more intuitively, we want to go from a more complex representation of our data to a less complex one (while retaining as much of the signal in our data as possible).

We can do this by looking at our data “from another angle”.

In doing this, we tease out the “principal components” of our data.

For example, suppose we have a dataset with some features that are related to each other.

Ideally, we would like to eliminate this redundancy and consolidate the number of variables we're looking at.
If these relationships are *linear*, then we can use well-established techniques like PCA/SVD.

EXAMPLE: 1D HARMONIC OSCILLATOR

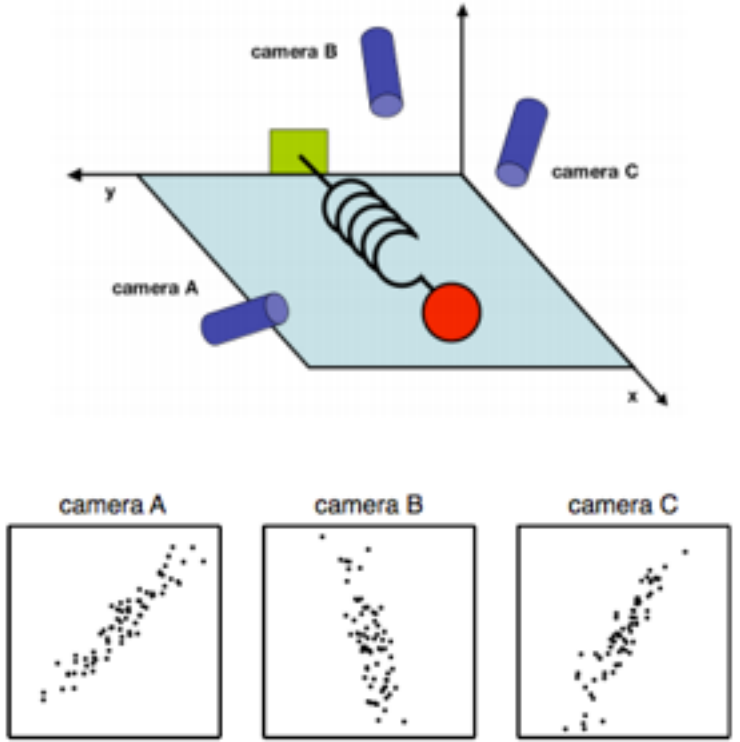


FIG. 1 A toy example. The position of a ball attached to an oscillating spring is recorded using three cameras A, B and C. The position of the ball tracked by each camera is depicted in each panel below.

source: http://www.snl.salk.edu/~shlens/pca.pdf

ASIDE: CURSE OF DIMENSIONALITY

The complexity that comes with a large number of features is due in part to the curse of dimensionality.

The complexity that comes with a large number of features is due in part to the curse of dimensionality.

Namely, the sample size needed to accurately estimate a random variable taking values in a d-dimensional feature space grows exponentially with d (almost).

Another way of characterizing this is to say that high-dimensional spaces are inherently sparse.

Another way of characterizing this is to say that high-dimensional spaces are inherently sparse.

ex: A high-dimensional orange contains most of its volume in the rind!

ex: A high-dimensional hypercube contains most of its volume in the corners!

In either case, most of the points in the space are “far” from the center.

In either case, most of the points in the space are “far” from the center.

This illustrates the fact that local methods will break down in these circumstances (eg, in order to collect enough neighbors for a given point, you need to expand the radius of the neighborhood so far that locality is not preserved).

In either case, most of the points in the space are “far” from the center.

This illustrates the fact that local methods will break down in these circumstances (eg, in order to collect enough neighbors for a given point, you need to expand the radius of the neighborhood so far that locality is not preserved).

The bottom line is that high-dimensional spaces can be problematic.

Q: What is the goal of dimensionality reduction?

Q: What is the goal of dimensionality reduction?

We'd like to analyze the data using the most meaningful basis (or coordinates) possible.

Q: What is the goal of dimensionality reduction?

We'd like to analyze the data using the most meaningful basis (or coordinates) possible.

More precisely: given an n x d matrix A (encoding n observations of a d-dimensional random variable), we want to find a k-dimensional representation of A ($k < d$) that captures the information in the original data, according to some criterion.

Q: What is the goal of dimensionality reduction?

- reduce computational expense
- reduce susceptibility to overfitting
- reduce noise in the dataset
- enhance our intuition

Q: How is dimensionality reduction performed?

Q: How is dimensionality reduction performed?
A: There are two approaches: feature selection and feature extraction.

Q: How is dimensionality reduction performed?
A: There are two approaches: feature selection and feature extraction.

feature selection – selecting a subset of features using an external criterion (*filter*) or the learning algo accuracy itself (*wrapper*)

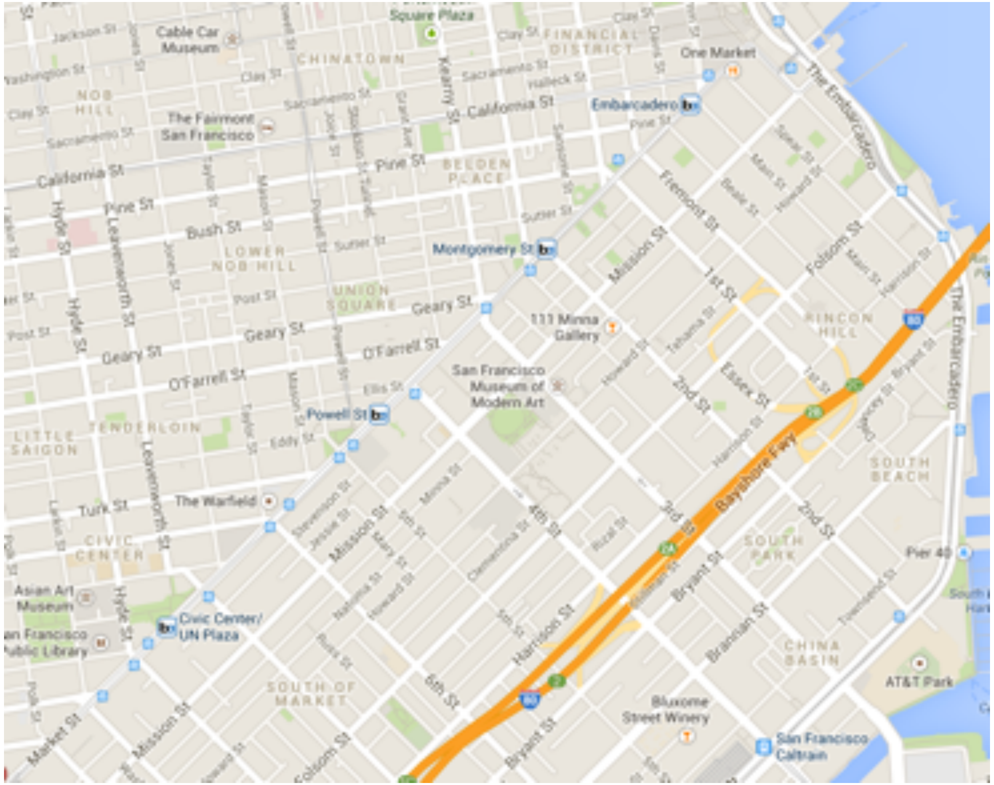
feature extraction – mapping the features to a lower dimensional space

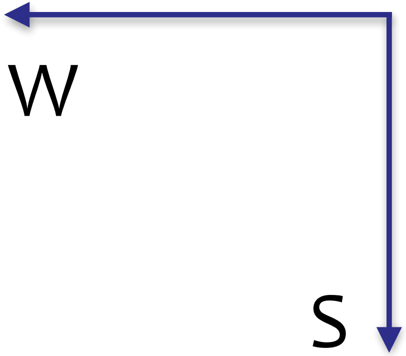
Feature selection is important, but typically when people say dimensionality reduction, they are referring to *feature extraction*.

Feature selection is important, but typically when people say dimensionality reduction, they are referring to *feature extraction*.

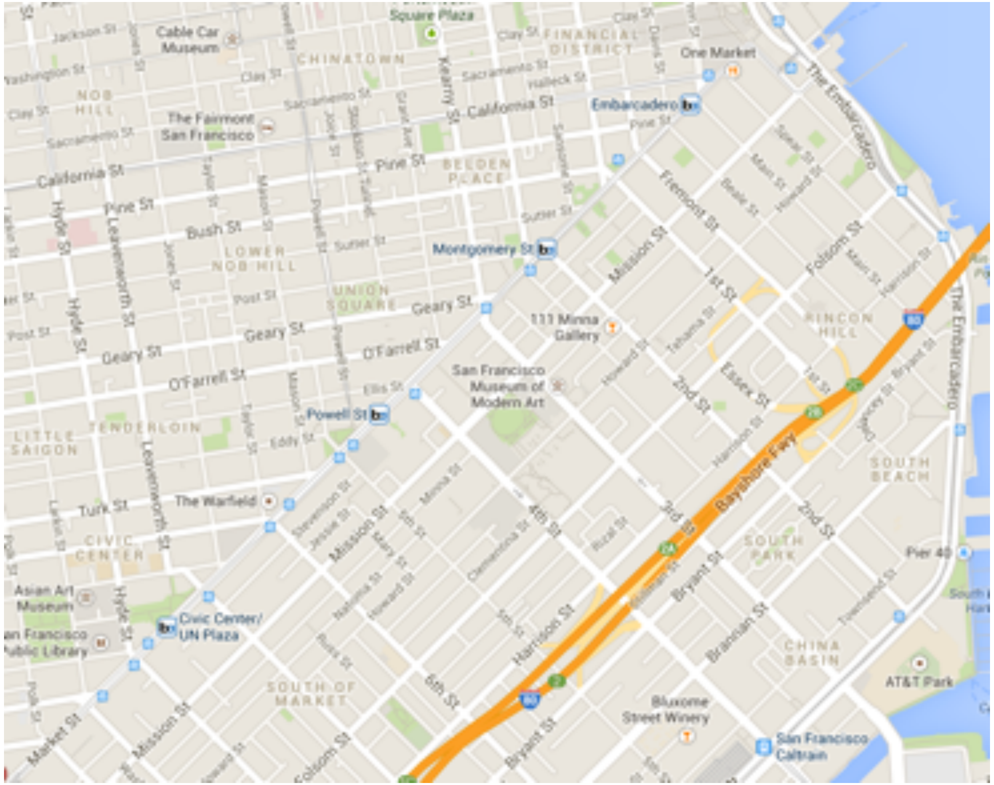
The goal of feature extraction is to create a new set of coordinates that *simplify the representation* of the data.

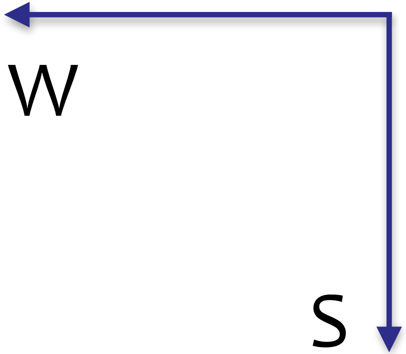
INTUITIVE EXAMPLE - BIKING DOWN MARKET STREET



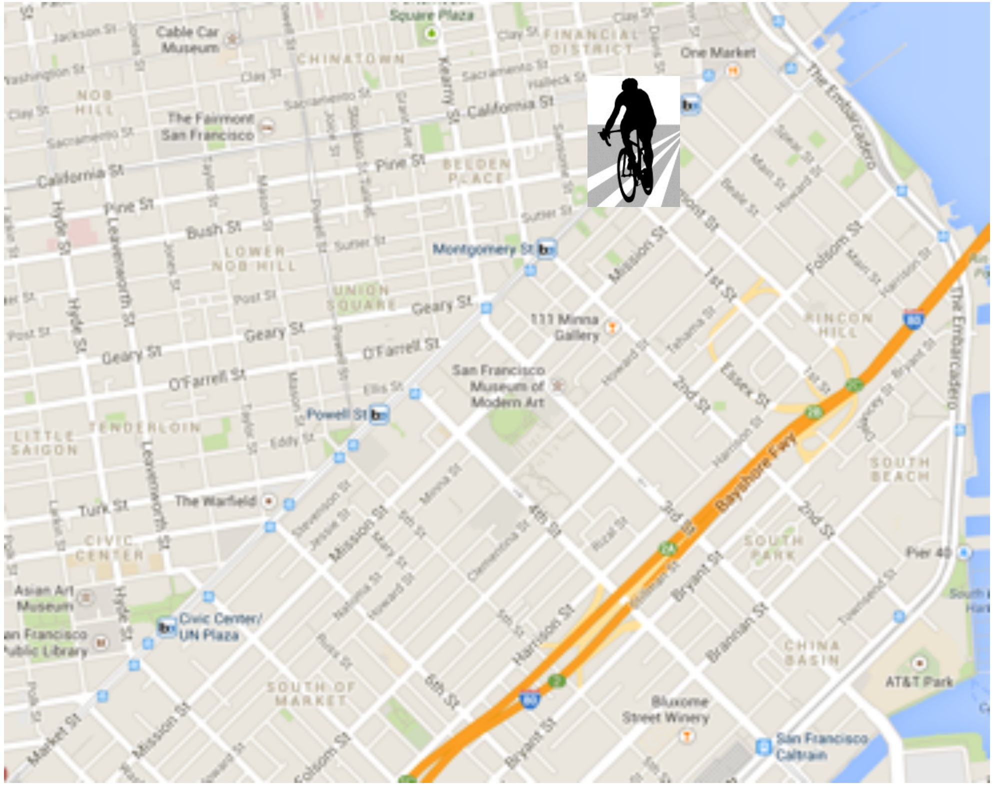


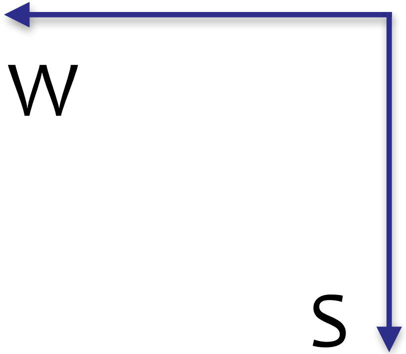
INTUITIVE EXAMPLE - BIKING DOWN MARKET STREET



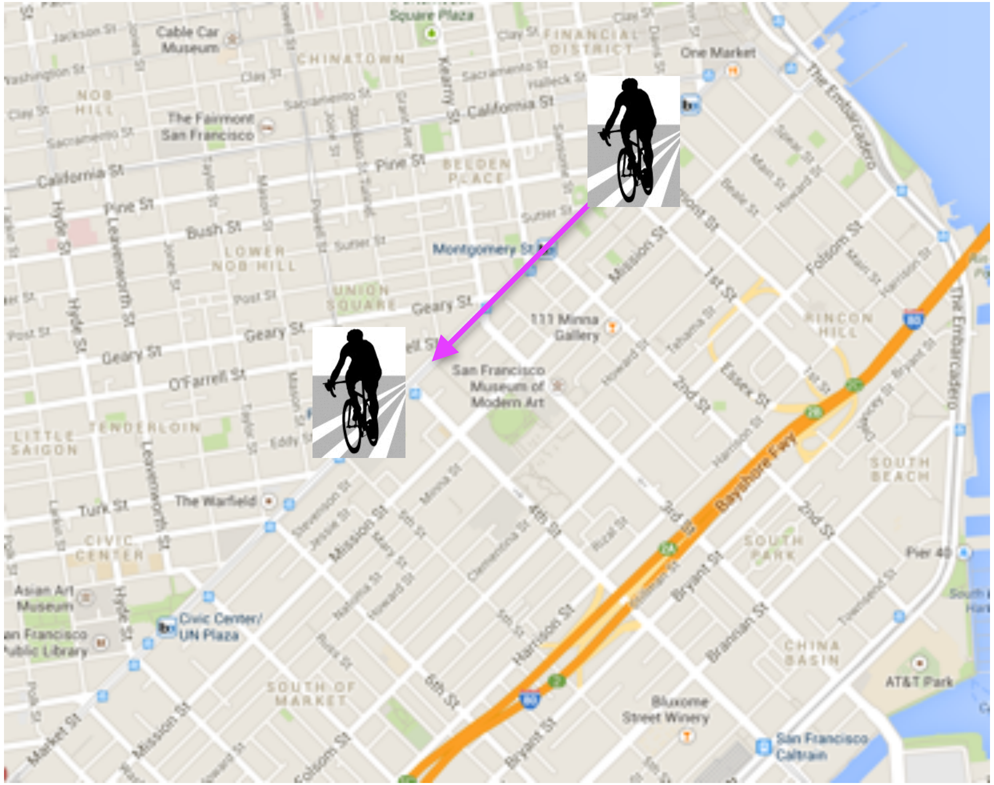


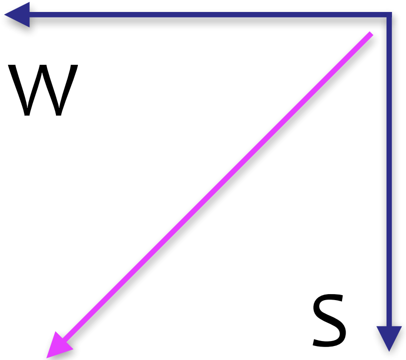
INTUITIVE EXAMPLE - BIKING DOWN MARKET STREET



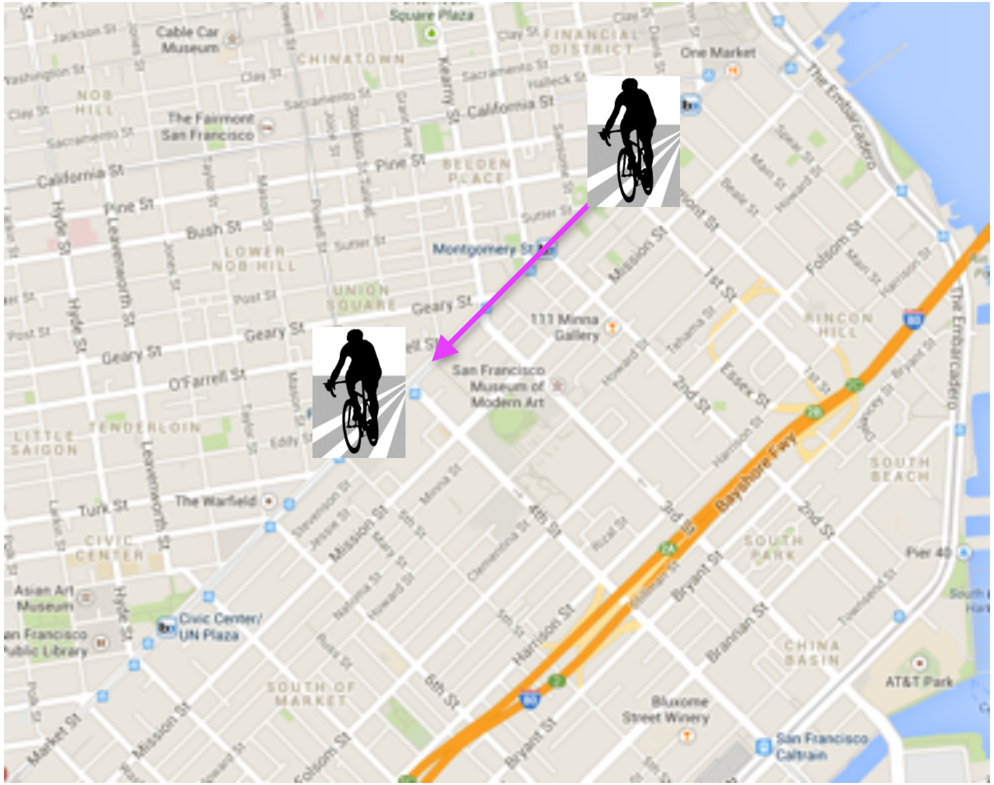


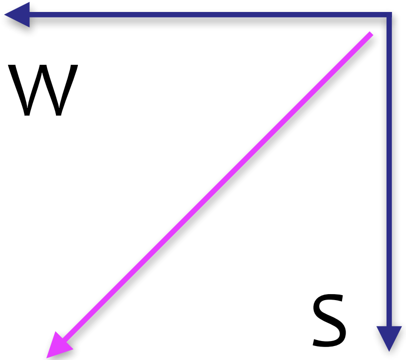
INTUITIVE EXAMPLE - BIKING DOWN MARKET STREET



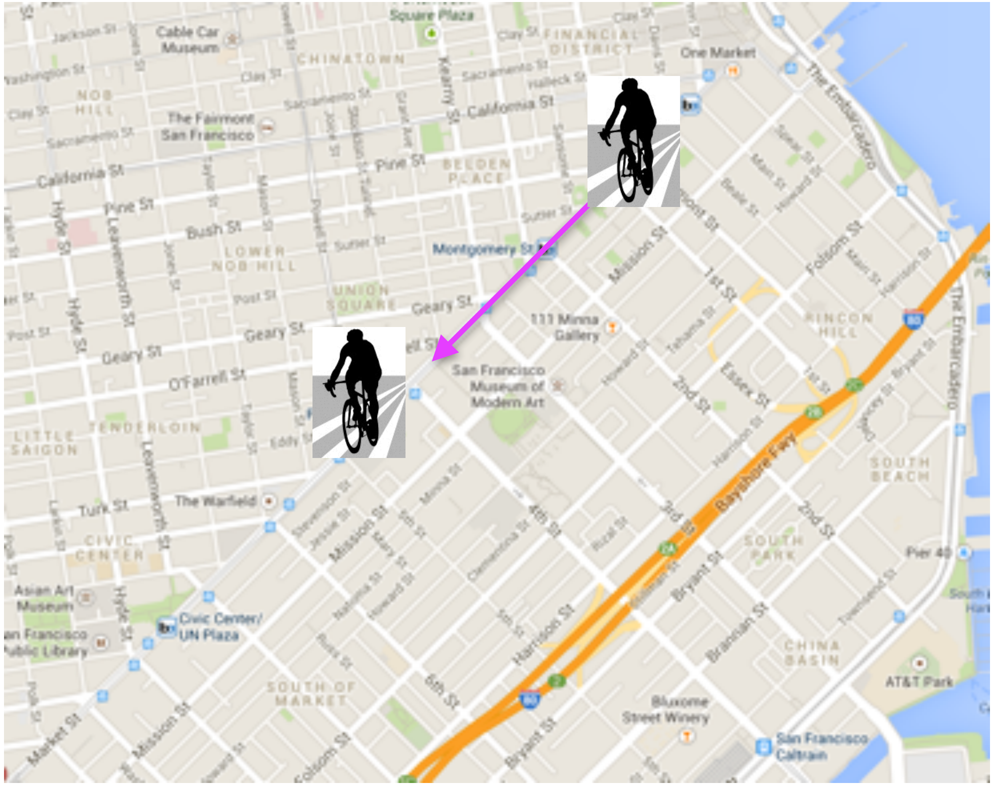


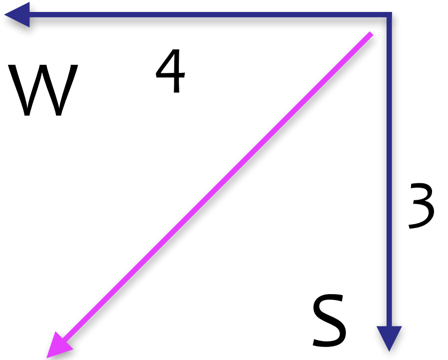
INTUITIVE EXAMPLE - BIKING DOWN MARKET STREET





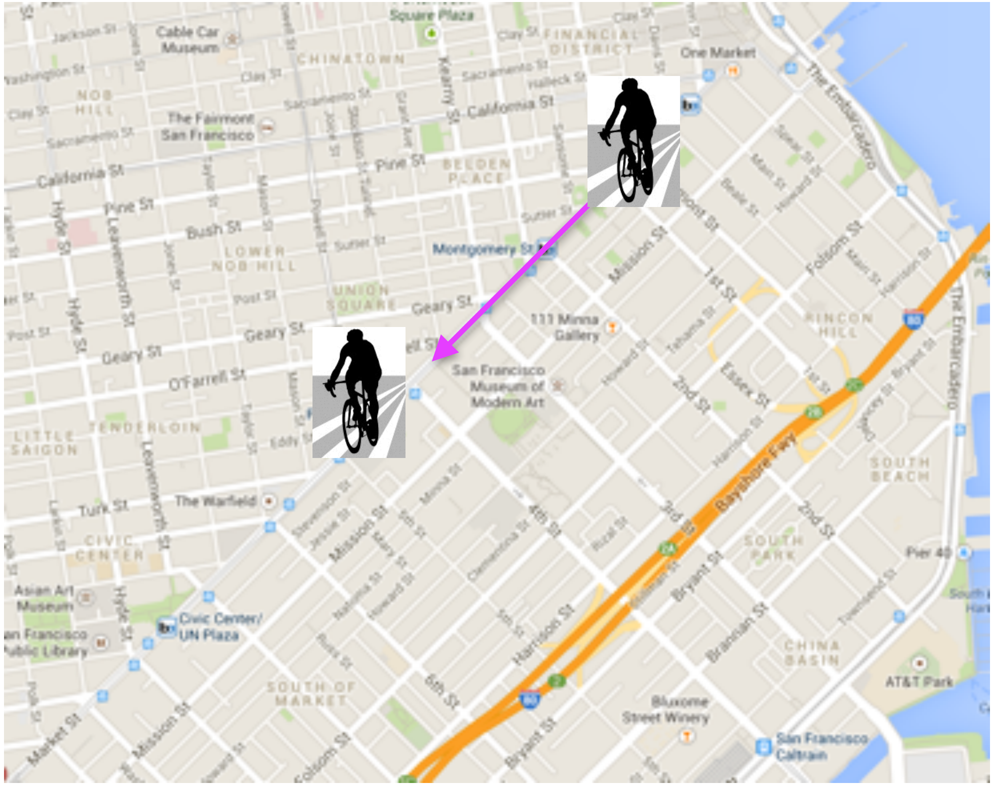
How many dimensions do we need to specify the position of this bike?

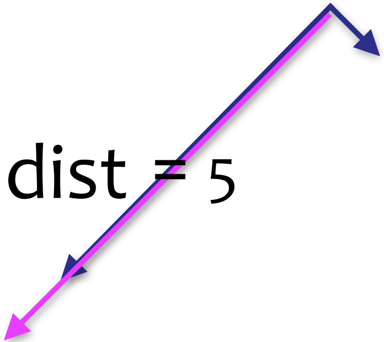




Yep, two. But could we represent the biker's position with fewer dimensions? How?

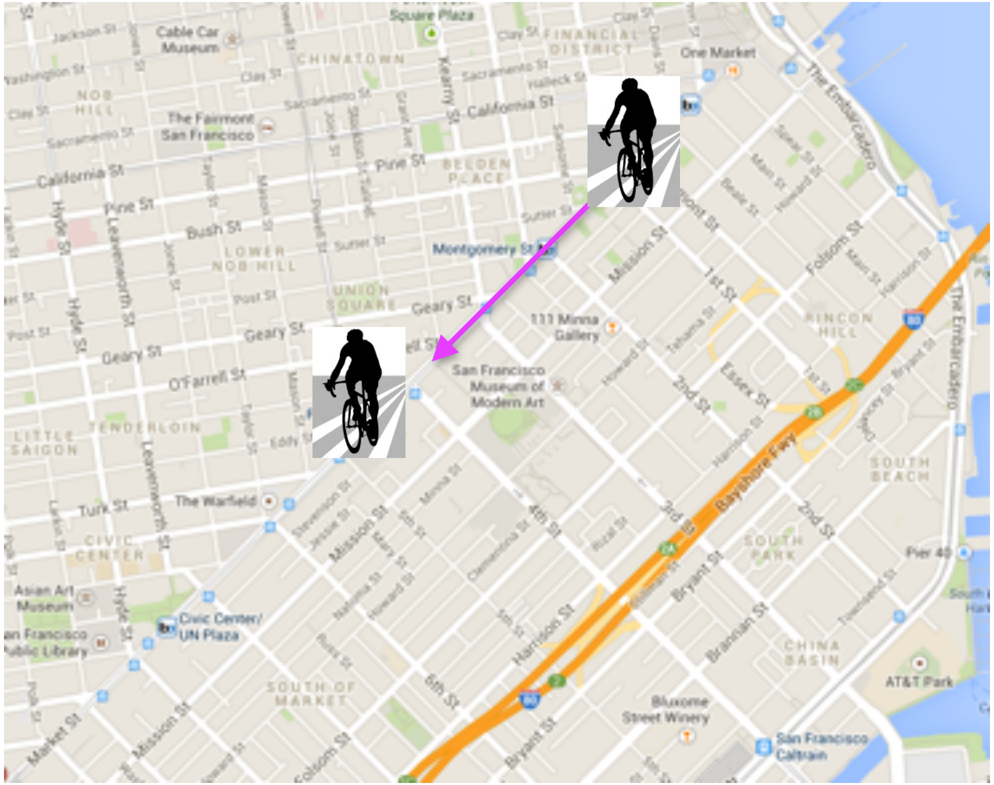
INTUITIVE EXAMPLE - BIKING DOWN MARKET STREET

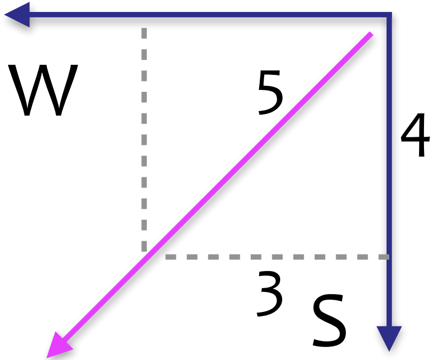




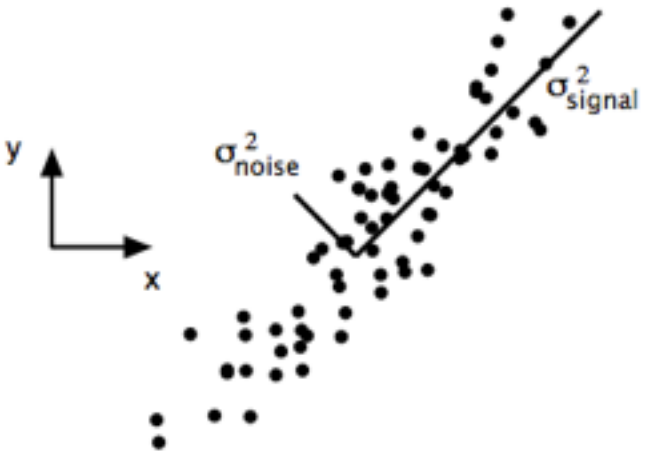
What if we just used distance down Market St.?

INTUITIVE EXAMPLE - BIKING DOWN MARKET STREET





Of course, we can always map back to the original coordinate system!



$$SNR = \frac{\sigma^2_{signal}}{\sigma^2_{noise}}.$$

FIG. 2 Simulated data of (x, y) for camera A. The signal and noise variances σ^2_{signal} and σ^2_{noise} are graphically represented by the two lines subtending the cloud of data. Note that the largest direction of variance does not lie along the basis of the recording (x_A, y_A) but rather along the best-fit line.

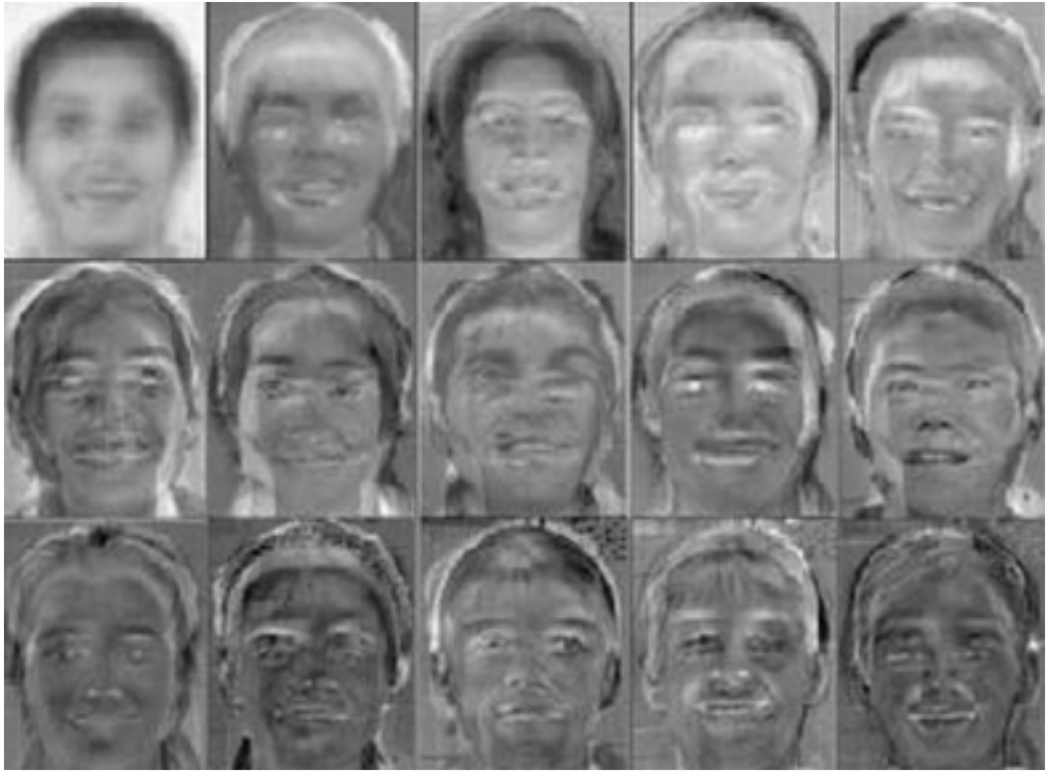
source: http://www.snl.salk.edu/~shlens/pca.pdf

Q: What are some applications of dimensionality reduction?

Q: What are some applications of dimensionality reduction?

- topic models (document clustering)
- image recognition/computer vision
- bioinformatics (microarray analysis)
- speech recognition
- astronomy (spectral data analysis)
- recommender systems

source: http://glowingpython.blogspot.it/2011/07/pca-and-image-compression-with-numpy.html



II. PRINCIPAL COMPONENT ANALYSIS

Principal component analysis is a dimension reduction technique that can be used on a matrix of any dimensions.

Principal component analysis is a dimension reduction technique that can be used on a matrix of any dimensions.

This procedure produces a new basis (a new coordinate system), each of whose components retain as much variance from the original data as possible.

Principal component analysis is a dimension reduction technique that can be used on a matrix of any dimensions.

This procedure produces a new basis, each of whose components retain as much variance from the original data as possible.

The PCA of a matrix A boils down to the eigenvalue decomposition of the covariance matrix of A.

The covariance matrix C of a matrix A is always square:

$$C = \begin{bmatrix} \mathrm{E}[(X_1-\mu_1)(X_1-\mu_1)] & \mathrm{E}[(X_1-\mu_1)(X_2-\mu_2)] & \cdots & \mathrm{E}[(X_1-\mu_1)(X_n-\mu_n)] \\ \mathrm{E}[(X_2-\mu_2)(X_1-\mu_1)] & \mathrm{E}[(X_2-\mu_2)(X_2-\mu_2)] & \cdots & \mathrm{E}[(X_2-\mu_2)(X_n-\mu_n)] \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{E}[(X_n-\mu_n)(X_1-\mu_1)] & \mathrm{E}[(X_n-\mu_n)(X_2-\mu_2)] & \cdots & \mathrm{E}[(X_n-\mu_n)(X_n-\mu_n)] \end{bmatrix}.$$

off-diagonal elements C_{ij} give the *covariance* between X_i, X_j $(i \neq j)$

diagonal elements C_{ii} give the *variance* of X_i

Wait a minute, what's a covariance matrix?

$$C = \begin{bmatrix} E[(X_1-\mu_1)(X_1-\mu_1)] & E[(X_1-\mu_1)(X_2-\mu_2)] & \cdots & E[(X_1-\mu_1)(X_n-\mu_n)] \\ E[(X_2-\mu_2)(X_1-\mu_1)] & E[(X_2-\mu_2)(X_2-\mu_2)] & \cdots & E[(X_2-\mu_2)(X_n-\mu_n)] \\ \vdots & \vdots & \ddots & \vdots \\ E[(X_n-\mu_n)(X_1-\mu_1)] & E[(X_n-\mu_n)(X_2-\mu_2)] & \cdots & E[(X_n-\mu_n)(X_n-\mu_n)] \end{bmatrix}.$$

For that matter, what is covariance?

Remember variance?

Remember variance?

$$s^2 = \frac{\sum_{i=1}^{n}(X_i - \bar{X})^2}{(n-1)}$$

Variance is the average distance from the mean of a data set to a point in that data set.

In other words, it is a measure of the *spread* of the data.

Recall that standard deviation is the square root of variance.

Standard deviation and variance only operate on 1 dimension, so that you could only calculate the standard deviation for each dimension of the data set *independently* of the other dimensions. However, it is useful to have a similar measure to find out how much the dimensions vary from the mean *with respect to each other*.

This is called covariance.

Variance:

$$s^2 = \frac{\sum_{i=1}^{n}(X_i - \bar{X})^2}{(n-1)} \qquad var(X) = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(X_i - \bar{X})}{(n-1)}$$

Covariance:

$$cov(X,Y) = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{(n-1)}$$

Covariance is always measured between two dimensions. If you calculate the covariance between a dimension and itself, you get the variance.

The covariance matrix *C* of a matrix *A* is always square:

$$C = \begin{pmatrix} cov(x,x) & cov(x,y) & cov(x,z) \\ cov(y,x) & cov(y,y) & cov(y,z) \\ cov(z,x) & cov(z,y) & cov(z,z) \end{pmatrix}$$

off-diagonal elements C_{ij} give the *covariance* between X_i, X_j $(i \neq j)$

diagonal elements C_{ii} give the *variance* of X_i

The covariance matrix C of a matrix A is always square:

$$C = \begin{bmatrix} \mathrm{E}[(X_1-\mu_1)(X_1-\mu_1)] & \mathrm{E}[(X_1-\mu_1)(X_2-\mu_2)] & \cdots & \mathrm{E}[(X_1-\mu_1)(X_n-\mu_n)] \\ \mathrm{E}[(X_2-\mu_2)(X_1-\mu_1)] & \mathrm{E}[(X_2-\mu_2)(X_2-\mu_2)] & \cdots & \mathrm{E}[(X_2-\mu_2)(X_n-\mu_n)] \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{E}[(X_n-\mu_n)(X_1-\mu_1)] & \mathrm{E}[(X_n-\mu_n)(X_2-\mu_2)] & \cdots & \mathrm{E}[(X_n-\mu_n)(X_n-\mu_n)] \end{bmatrix}.$$

off-diagonal elements C_{ij} give the *covariance* between X_i, X_j $(i \neq j)$

diagonal elements C_{ii} give the *variance* of X_i

ASIDE: EIGENVALUE DECOMPOSITION

The *eigenvalue decomposition* of a square matrix A is given by:

$$A = Q\Lambda Q^{-1}$$

The *eigenvalue decomposition* of a square matrix A is given by:

$$A = Q\Lambda Q^{-1}$$

The columns of Q are the eigenvectors of A, and the values in Λ are the associated eigenvalues of A.

The *eigenvalue decomposition* of a square matrix A is given by:

$$A = Q\Lambda Q^{-1}$$

The columns of Q are the eigenvectors of A, and the values in Λ are the associated eigenvalues of A.

For an eigenvector v of A and its eigenvalue λ, we have the important relation:

$$Av = \lambda v$$

ASIDE: EIGENVALUE DECOMPOSITION

The *eigenvalue decomposition* of a square matrix A is given by:

$$A = Q\Lambda Q^{-1}$$

The columns of Q are the eigenvectors of A, and the
are the associated eigenvalues of A.

NOTE

This relationship *defines* what it means to be an eigenvector of A.

For an eigenvector v of A and its eigenvalue λ, we have the important relation:

$$Av = \lambda v$$

The eigenvectors form a basis of the vector space on which A acts (e.g., they are orthogonal).

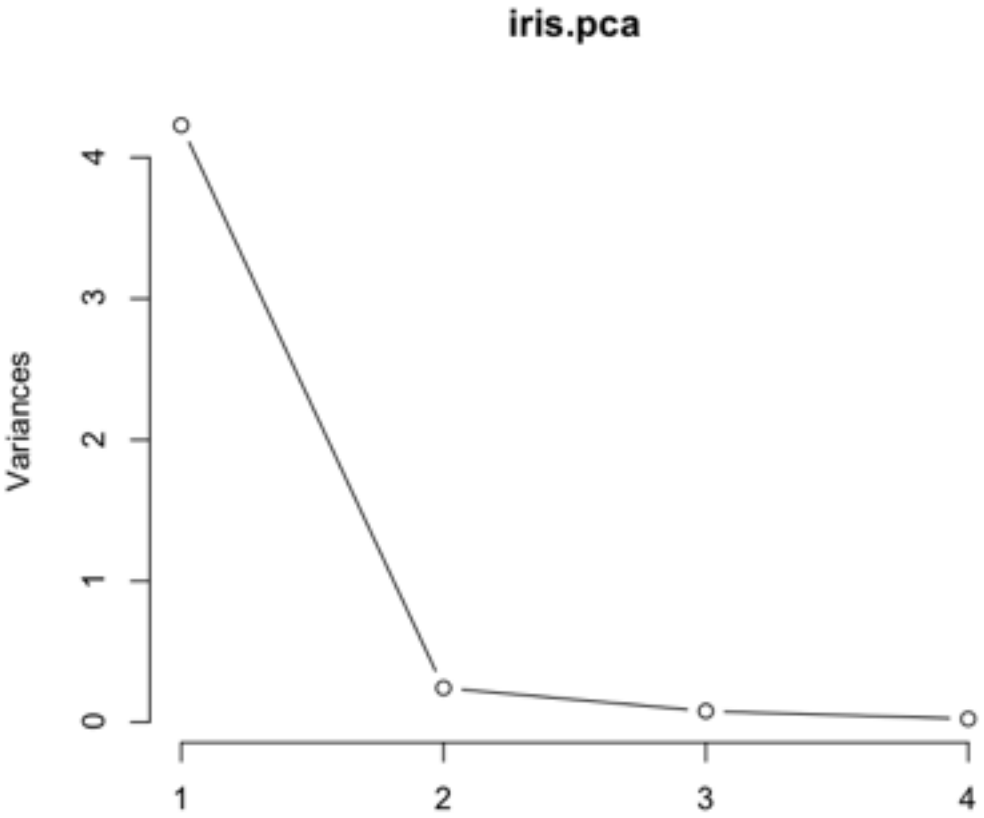
The eigenvectors form a basis of the vector space on which A acts (e.g., they are orthogonal).

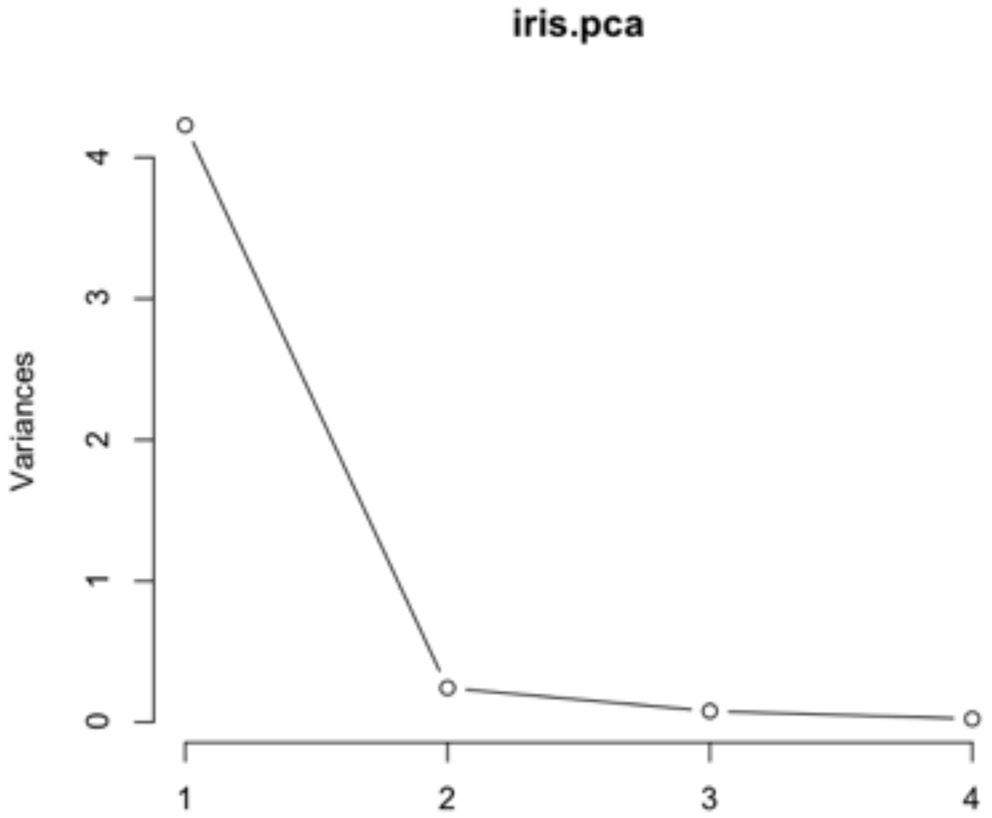
Furthermore the basis elements are ordered by their eigenvalues (from largest to smallest), and these eigenvalues represent the amount of variance explained by each basis element.

The eigenvectors form a basis of the vector space on which A acts (eg, they are orthogonal).

Furthermore the basis elements are ordered by their eigenvalues (from largest to smallest), and these eigenvalues represent the amount of variance explained by each basis element.

This can be visualized in a scree plot, which shows the amount of variance explained by each basis vector.


iris.pca
Variances
0
1
2
3
4
1
2
3
4



NOTE

Looking at this plot also gives you an idea of how many principal components to keep.

Apply the *elbow test*: keep only those pc's that appear to the left of the elbow in the graph.

1. **Linearity** – The change in basis is a linear projection

2. **Large variances have important structure** – e.g. large signal-to-noise ratio. In other words, we assume that principal components with larger associated variances are signal, while those with lower variances represent noise. NOTE: this is a strong (and not always correct) assumption!

3. **The principal components are orthogonal** – A simplification that makes PCA soluble with linear algebra matrix decomposition techniques