CPSC 340: Machine Learning and Data Mining

More PCA

Fall 2017

Admin

- Assignment 4:
  - Due Friday of next week.

- No class Monday due to holiday.
  - There will be tutorials next week on MAP/PCA (except Monday).

1. Decision trees
2. Naïve Bayes classification
3. Ordinary least squares regression
4. Logistic regression
5. Support vector machines
6. Ensemble methods
7. Clustering algorithms
8. Principal component analysis
9. Singular value decomposition
10. Independent component analysis (bonus)

http://www.kdnuggets.com/2016/08/10-algorithms-machine-learning-engineers.html

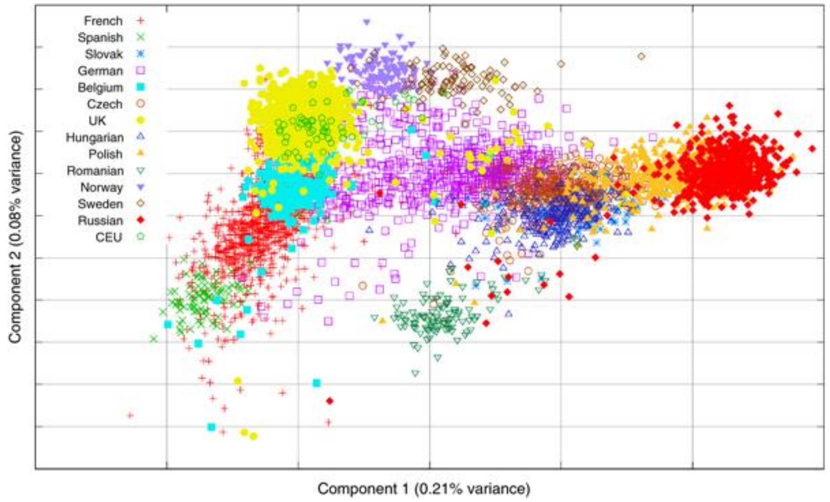
Last Time: Latent-Factor Models

- Latent-factor models take input data ‘X’ and output a basis ‘Z’:

X → Latent-factor model → Z

  – Usually, ‘Z’ has fewer features than ‘X’.

- Uses: dimensionality reduction, visualization, factor discovery.



Trait	Description
Openness	Being curious, original, intellectual, creative, and open to new ideas.
Conscientiousness	Being organized, systematic, punctual, achievement-oriented, and dependable.
Extraversion	Being outgoing, talkative, sociable, and enjoying social situations.
Agreeableness	Being affable, tolerant, sensitive, trusting, kind, and warm.
Neuroticism	Being anxious, irritable, temperamental, and moody.

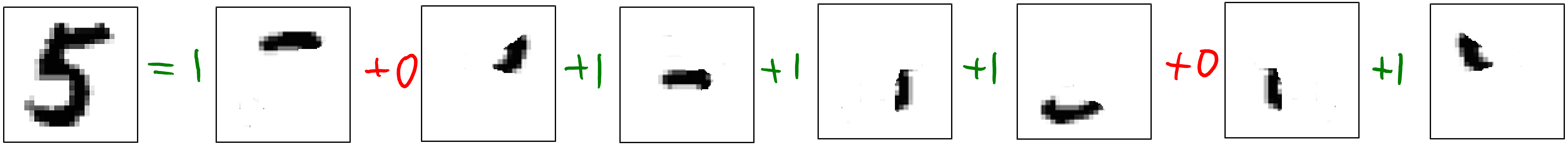
http://infoproc.blogspot.ca/2008/11/european-genetic-substructure.html
https://new.edu/resources/big-5-personality-traits

Last Time: Principal Component Analysis

- Principal component analysis (PCA) is a linear latent-factor model:
  - These models “factorize” matrix X into matrices Z and W:

$$\underset{n\times d}{X} \approx \underset{n\times k}{Z}\,\underset{k\times d}{W} \qquad x_i \approx W^T z_i \qquad x_{ij} \approx (w^j)^T z_i$$

  - We can think of rows w_c of W as ‘k’ fixed “part” (used in all examples).
  - z_i is the “part weights” for example x_i: “how much of each part w_c to use”.

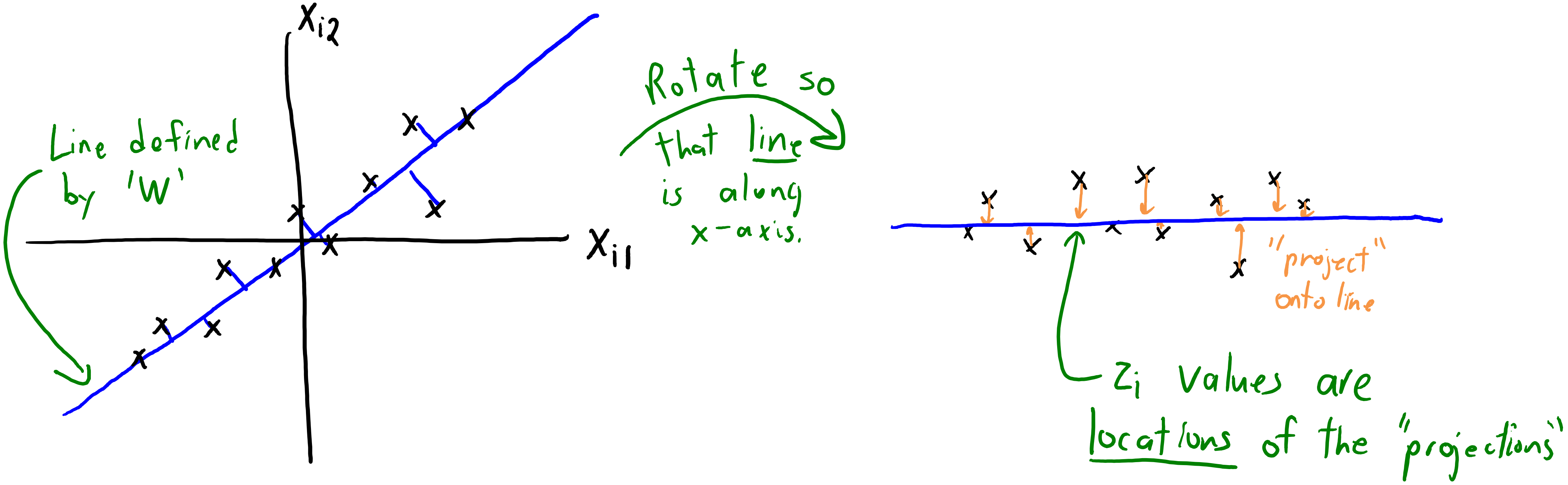


$$\hat{x}_i = z_{i1} w_1 + z_{i2} w_2 + z_{i3} w_3 + z_{i4} w_4 + z_{i5} w_5 + z_{i6} w_6 + z_{i7} w_7$$

“parts” are called “principal components” or “factors”

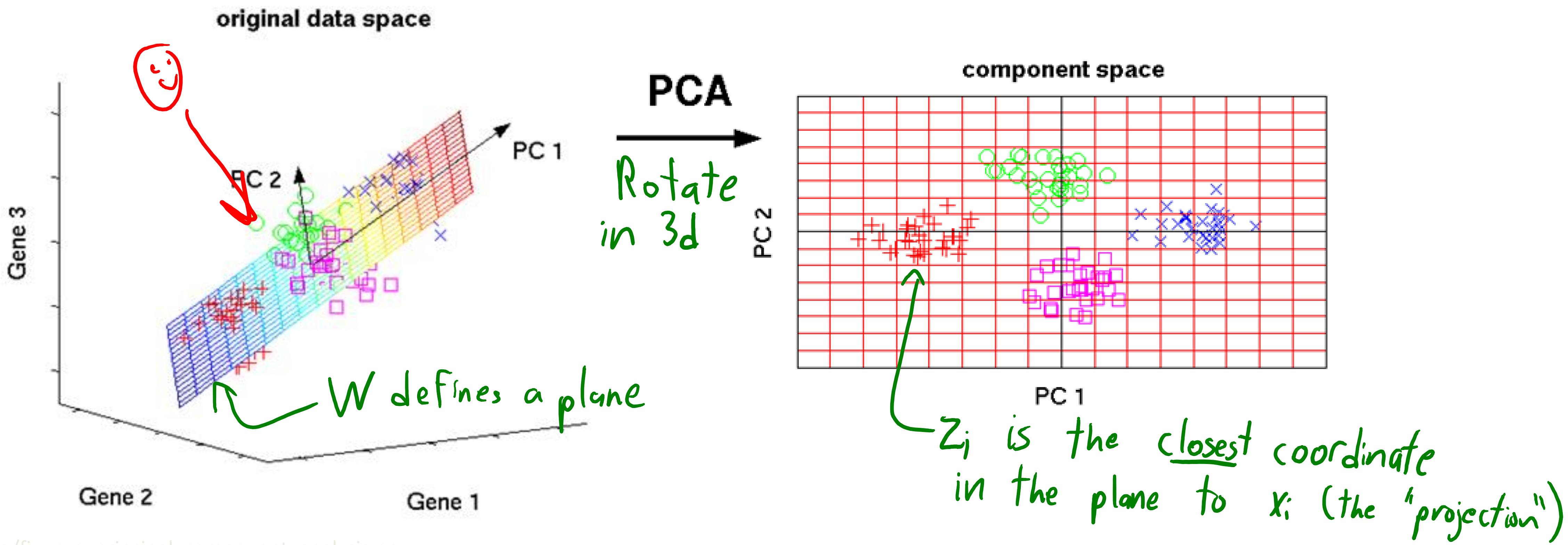
Last Time: PCA Geometry

- When $k=1$, the W matrix defines a line:
  - We choose 'W' as the line minimizing squared distance to the data.
  - Given 'W', the z_i are the coordinates of the x_i "projected" onto the line.



Last Time: PCA Geometry

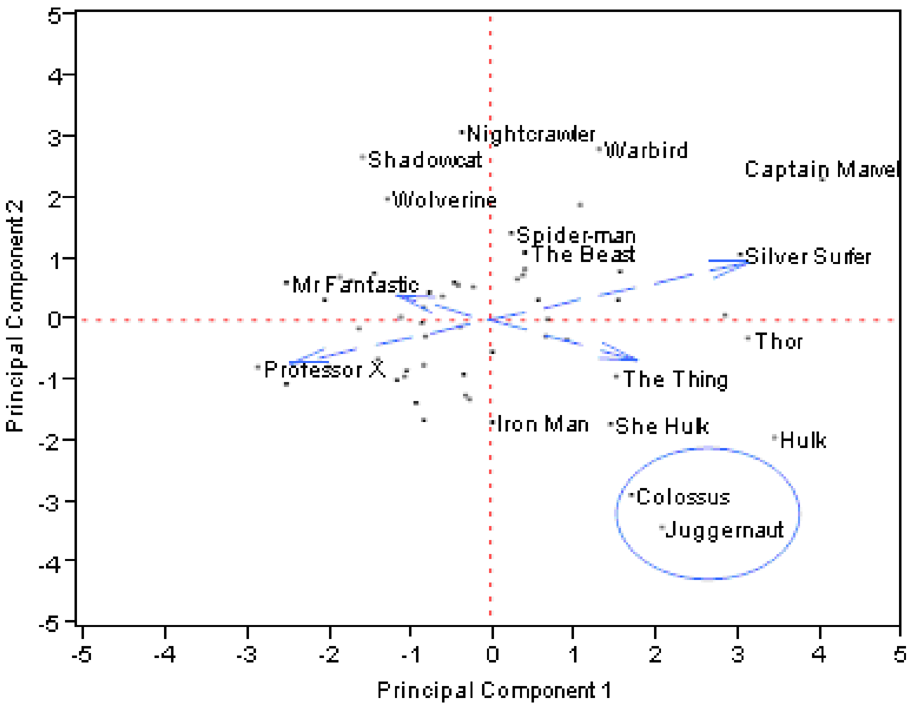
- When $k=2$, the W matrix defines a plane:
  - We choose 'W' as the plane minimizing squared distance to the data.
  - Given 'W', the z_i are the coordinates of the x_i "projected" onto the plane.

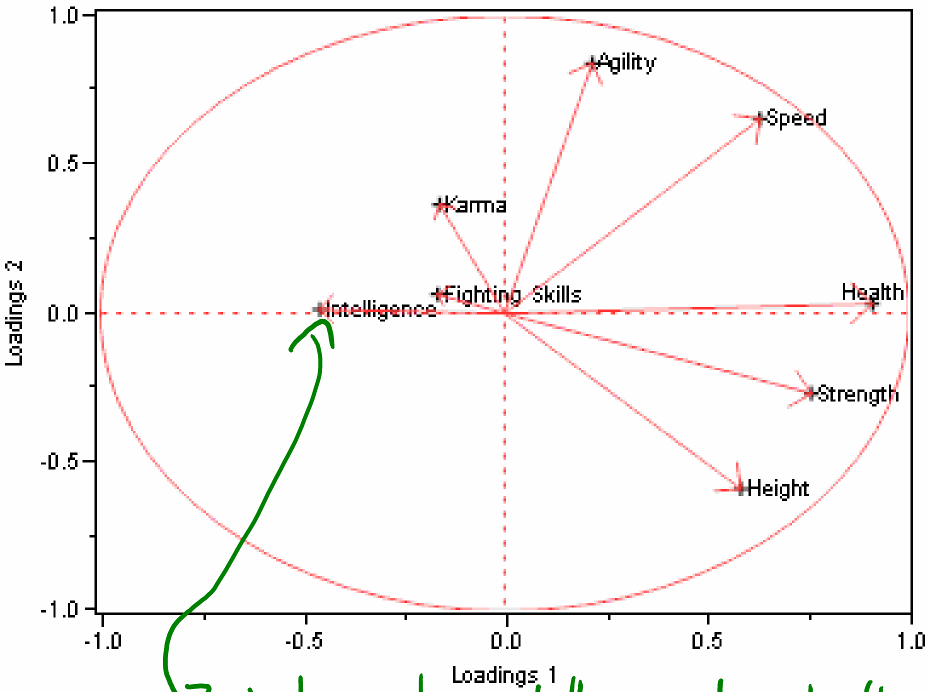


http://www.nlpca.org/fig_pca_principal_component_analysis.png

Last Time: PCA Geometry

- When $k=2$, the W matrix defines a plane:
  – Even if the original data is high-dimensional,
    we can visualize data “projected” onto this plane.





z_i value when intelligence$=1$ and other $x_{ij}=0$

http://www.prismtc.co.uk/superheroes-pca/

Digression: Data Centering (Important)

- In PCA, we assume that the data X is "centered".
  - Each column of X has a mean of zero.

- It's easy to center the data:

Set $\mu_j = \frac{1}{n}\sum_{i=1}^{n} x_{ij}$ (mean of column 'j')

Replace each x_{ij} with $(x_{ij} - \mu_j)$

- There are PCA variations that estimate "bias in each coordinate".
  - In basic model this is equivalent to centering the data.

PCA Computation: Alternating Minimization

- With centered data, the PCA objective is:

$$f(W,Z) = \sum_{i=1}^{n} \sum_{j=1}^{d} \left((w^j)^T z_i - x_{ij} \right)^2$$

- In k-means we tried to optimize this with alternating minimization:
  – Fix "cluster assignments" Z and find the optimal "means" W.
  – Fix "means" W and find the optimal "cluster assignments" Z.

- Converges to a local optimum.
  – But may not find a global optimum (sensitive to initialization).

PCA Computation: Alternating Minimization

- With centered data, the PCA objective is:

$$f(W,Z) = \sum_{i=1}^{n} \sum_{j=1}^{d} ((w^j)^T z_i - x_{ij})^2$$

- In PCA we can also use alternating minimization:
  - Fix "part weights" Z and find the optimal "parts" W.
  - Fix "parts" W and find the optimal "part weights" Z.

- Converges to a local optimum.
  - Which will be a global optimum (if we randomly initialize W and Z).

PCA Computation: Alternating Minimization

- With centered data, the PCA objective is:

$$f(W,Z) = \sum_{i=1}^{n} \sum_{j=1}^{d} \left((w^j)^T z_i - x_{ij} \right)^2$$

- Alternating minimization steps:
  - If we fix Z, this is a quadratic function of W (least squares column-wise):

$$\nabla_W f(W,Z) = Z^T Z W - Z^T X \quad \text{so} \quad W = \underline{(Z^T Z)}^{-1}(Z^T X)$$

(writing gradient as a matrix)

  - If we fix W, this is a quadratic function of Z (transpose due to dimensions):

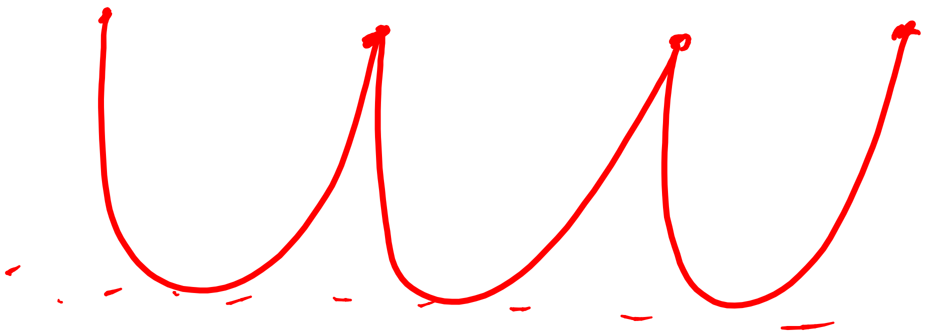
$$\nabla_Z f(W,Z) = Z W W^T - X W^T \quad \text{so} \quad Z = X W^T \underline{(W W^T)}^{-1}$$

These are usually invertible since $k \ll n$ and $k \ll d$

PCA Computation: Alternating Minimization

- With centered data, the PCA objective is:

$$f(W,Z) = \sum_{i=1}^{n}\sum_{j=1}^{d}\left((w^j)^T z_i - x_{ij}\right)^2$$



- This objective is not jointly convex in W and Z.
  – You will find different W and Z depending on the initialization.
    - For example, if you initialize with $W_1 = 0$, then they will stay at zero.

  – But it's possible to show that all "stable" local optima are global optima.
    - You will converge to a global optimum in practice if you initialize randomly.
      – Randomization means you don't start on one of the unstable non-global critical points.
    - E.g., sample each initial z_{ij} from a normal distribution.

PCA Computation: Prediction

- At the end of training, the "model" is the μ_j and the W matrix.
  - PCA is parametric.
- PCA prediction phase:
  - Given new data $\tilde{X}$, we can use μ_j and W this to form $\tilde{Z}$:

1. Center: replace each $\tilde{x}_{ij}$ with $(\tilde{x}_{ij} - \underline{\mu_j})$

means of training data

2. Find $\tilde{Z}$ minimizing squared error:

$$\tilde{Z} = \tilde{X}\underbrace{W^T(WW^T)^{-1}}_{\text{(could just store this } d \times k \text{ matrix)}}$$

PCA Computation: Prediction

- At the end of training, the "model" is the μ_j and the W matrix.
  - PCA is parametric.
- PCA prediction phase:
  - Given new data $\tilde{X}$, we can use μ_j and W this to form $\tilde{Z}$:
  - The "reconstruction error" is how close approximation is to $\tilde{X}$:

$$\|\underbrace{\tilde{Z}W}_{\hat{X}} - \tilde{X}\|_F^2$$

centered version

  - Our "error" from replacing the x_i with the z_i and W.

PCA Computation: Stochastic Gradient

- For big X matrices, you can also use stochastic gradient:

$$f(W,Z) = \sum_{i=1}^{n}\sum_{j=1}^{d}((w^j)^T z_i - x_{ij})^2 = \sum_{(i,j)} \underbrace{((w^j)^T z_i - x_{ij})^2}_{f(w_j, z_i, x_{ij})}$$

On each iteration, pick a random example 'i' and feature 'j':

→ Set $w_j^{t+1} = w_j^t - \alpha^t \nabla_{w_j} f(w_j, z_i, x_{ij})$

→ Set $z_i^{t+1} = z_i^t - \alpha^t \nabla_{z_i} f(w_j, z_i, x_{ij})$

- (Other variables stay the same.)

Choosing 'k' by "Variance Explained"

- "Variance" approach to choosing 'k':
  - Consider the variance of the x_{ij} values:

$$\underbrace{Var(x_{ij}) = E[(x_{ij} - \underbrace{\mu_{ij}}_{\text{assumed to be zero}})^2]}_{\text{definition of variance}} = \underbrace{E[x_{ij}^2] = \frac{1}{nd}}_{\text{definition of expectation}} \underbrace{\sum_{i=1}^{n}\sum_{j=1}^{d} x_{ij}^2 = \frac{1}{nd}\|X\|_F^2}_{\text{Frobenius norm}}$$

  - For a given 'k' we compute (variance of errors)/(variance of x_{ij}):

$$\frac{\|ZW - X\|_F^2}{\|X\|_F^2}$$

Centered version

  - Gives a number between 0 (k=n) and 1 (k=0), giving "variance remaining".
    - If you want to "explain 90% of variance", choose smallest 'k' where ratio is < 0.10.

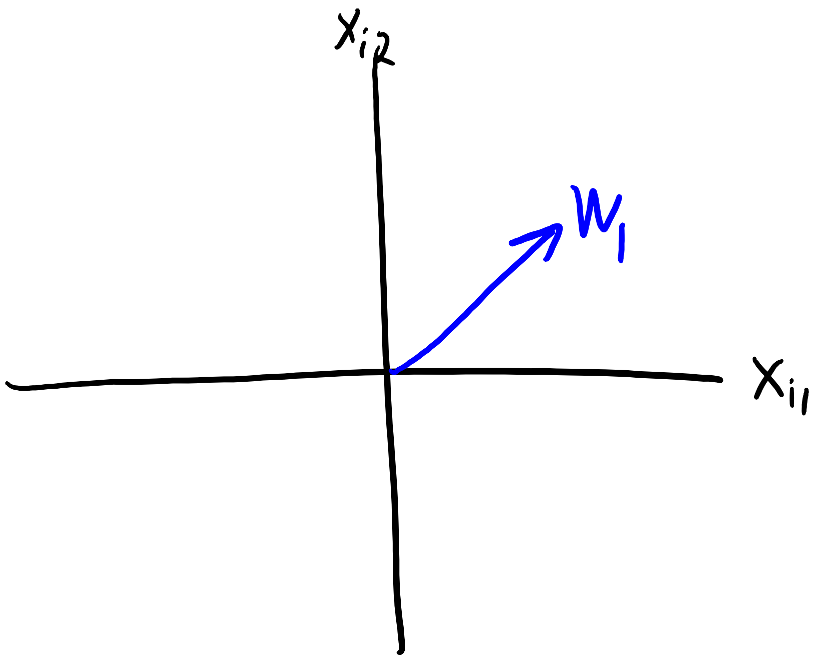
(pause)

Non-Uniqueness of PCA

- Alternating minimization and stochastic gradient find a global min.
  - But the actual W and Z are still sensitive to the initialization.
- This is because many different W and Z minimize f(W,Z).
  - The solution is not unique.
- To understand why, we'll need idea of "span" from linear algebra.
  - This also helps explain the geometry of PCA.
  - We'll also see that some global optima may be better than others.

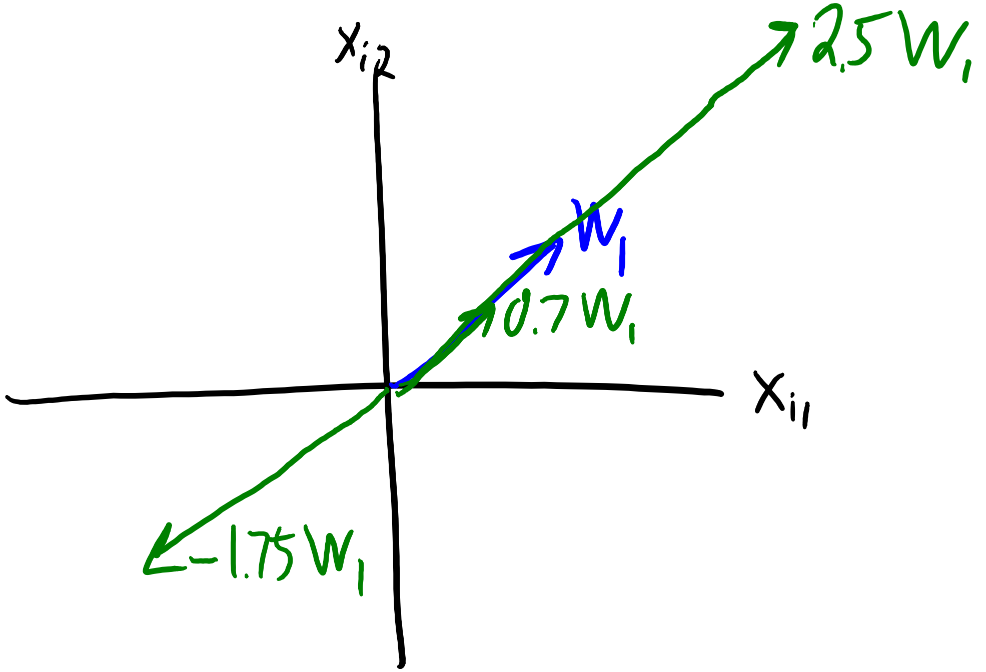
Span of 1 Vector

- Consider a single vector w_1 (k=1).



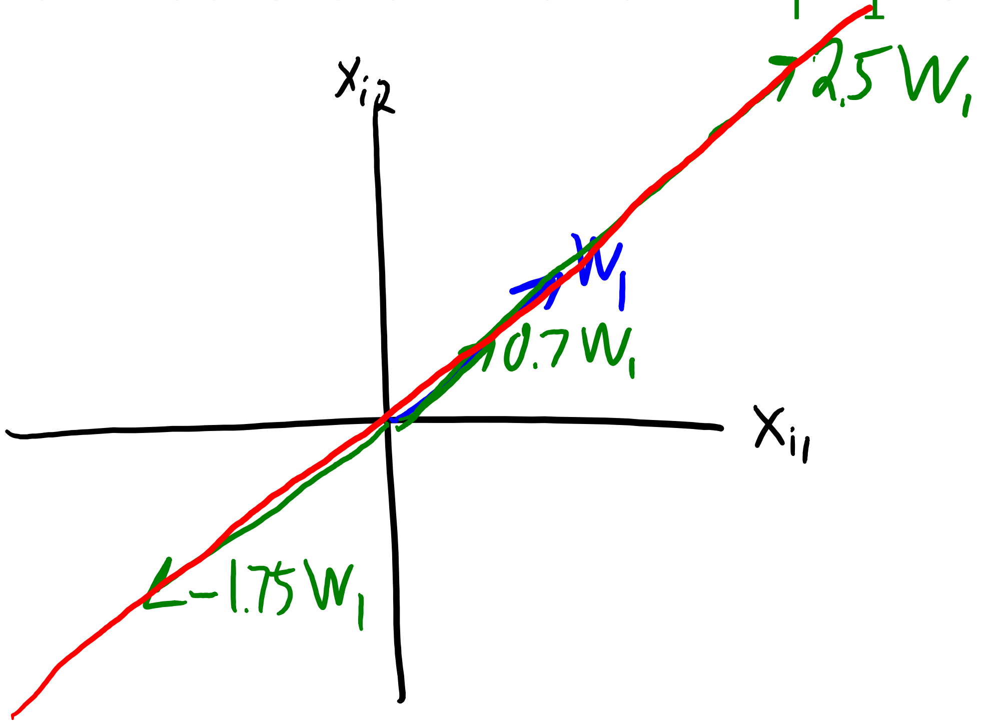
Span of 1 Vector

- Consider a single vector w_1 (k=1).
- The span(w_1) is all vectors of the form $z_i w_1$ for a scalar z_i.



Span of 1 Vector

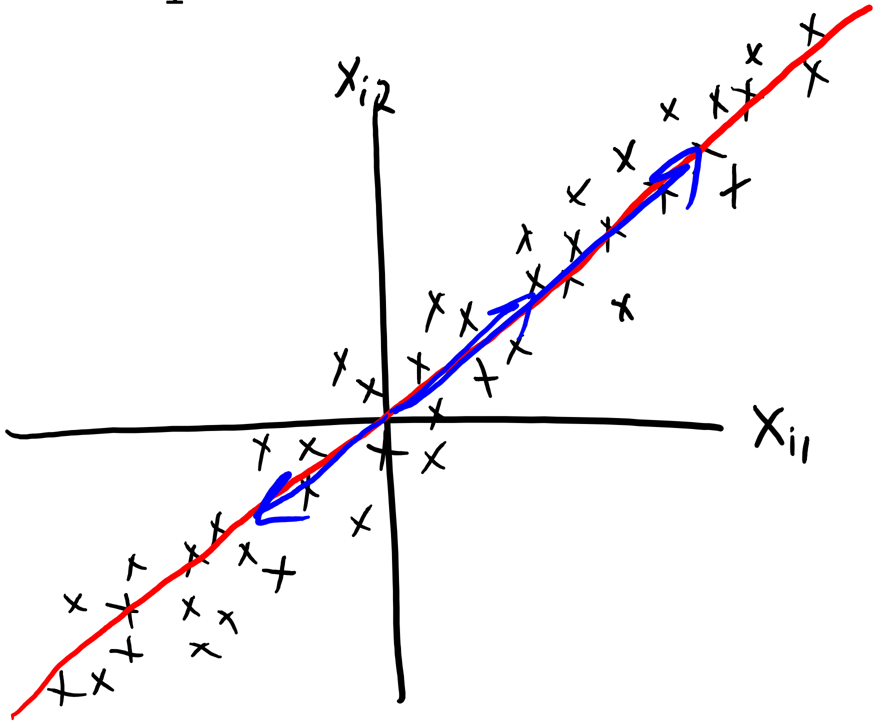
- Consider a single vector w_1 ($k=1$).
- The span(w_1) is all vectors of the form $z_i w_1$ for a scalar z_i.



- If $w_1 \neq 0$, this forms a line.

Span of 1 Vector

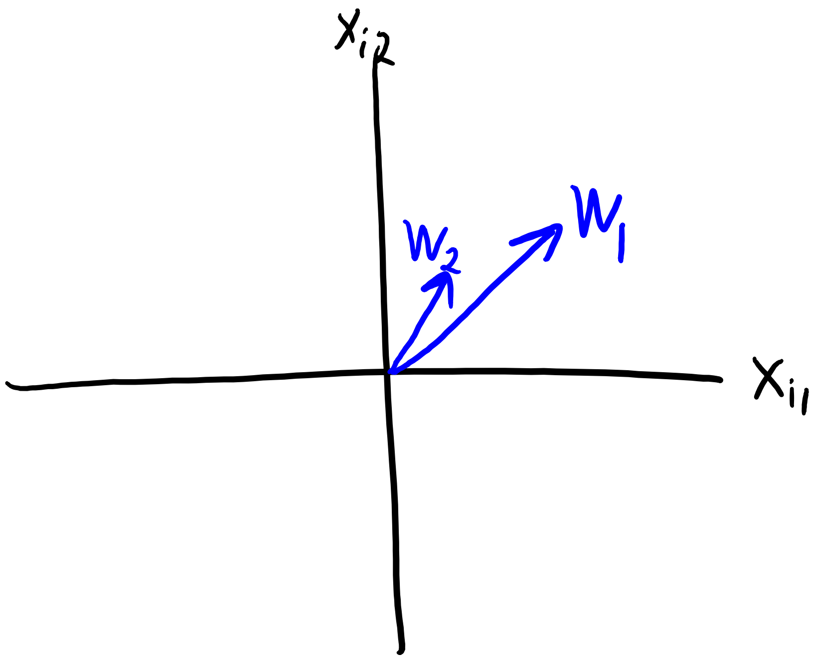
- But note that the "span" of many different vectors gives same line.
  - Mathematically: αw_1 defines the same line as w_1 for any scalar $\alpha \neq 0$.



  - PCA solution can only be defined up to scalar multiplication.
    - If (W,Z) is a solution, then $(\alpha W,(1/\alpha)Z)$ is also a solution. $\|(\alpha W)(\frac{1}{\alpha}Z)-X\|_F^2 = \|WZ-X\|_F^2$

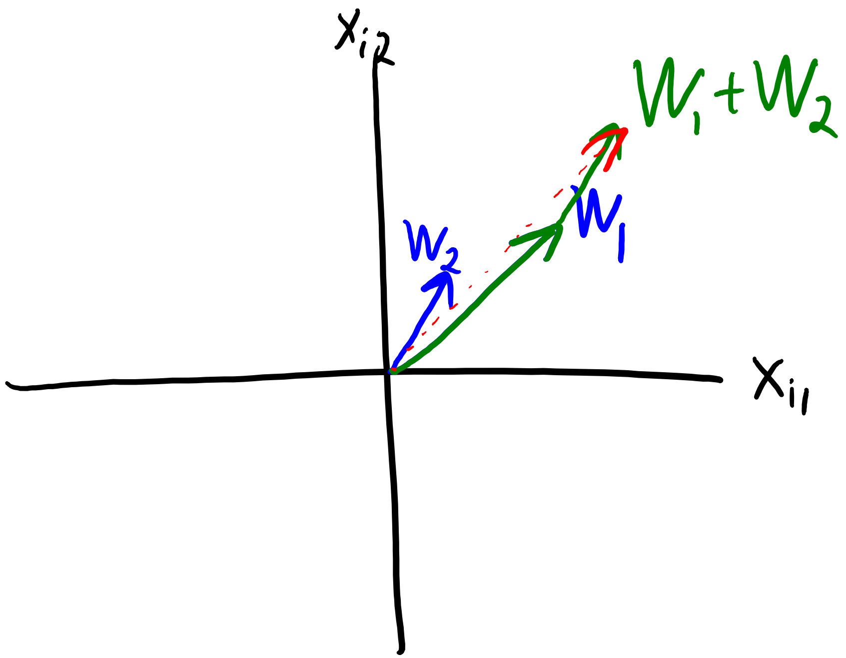
Span of 2 Vectors

- Consider two vector w_1 and w_2 ($k=2$).



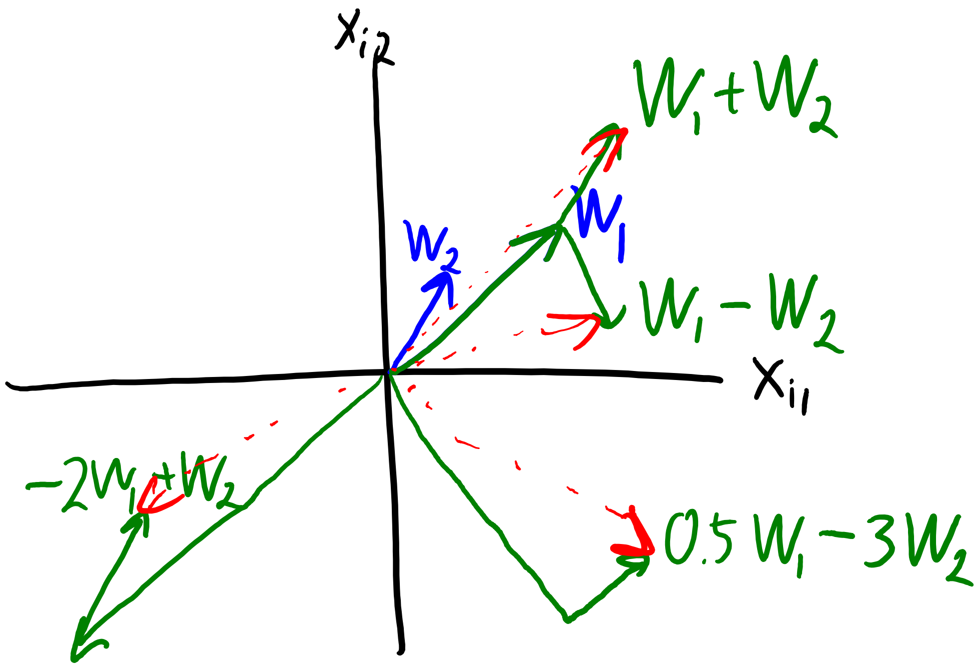
Span of 2 Vectors

- Consider two vector w_1 and w_2 ($k=2$).
  - The span(w_1,w_2) is all vectors of form $z_{i1}w_1 + z_{i2}w_2$ for a scalars z_{i1} and z_{i2}.



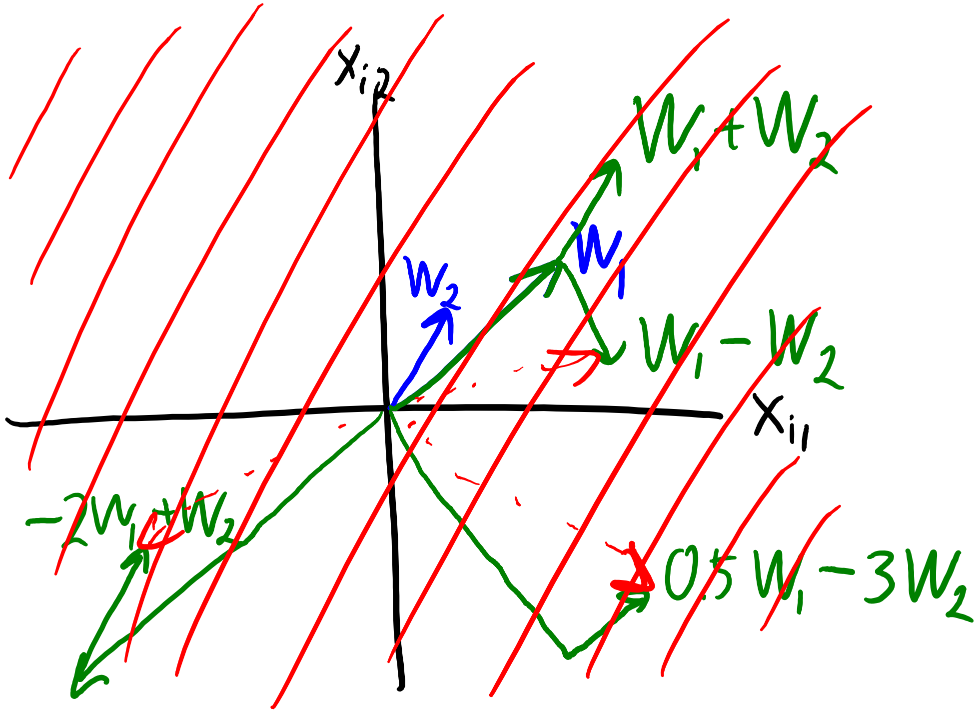
Span of 2 Vectors

- Consider two vector w_1 and w_2 (k=2).
  - The span(w_1,w_2) is all vectors of form $z_{i1}w_1 + z_{i2}w_2$ for a scalars z_{i1} and z_{i2}.



Span of 2 Vectors

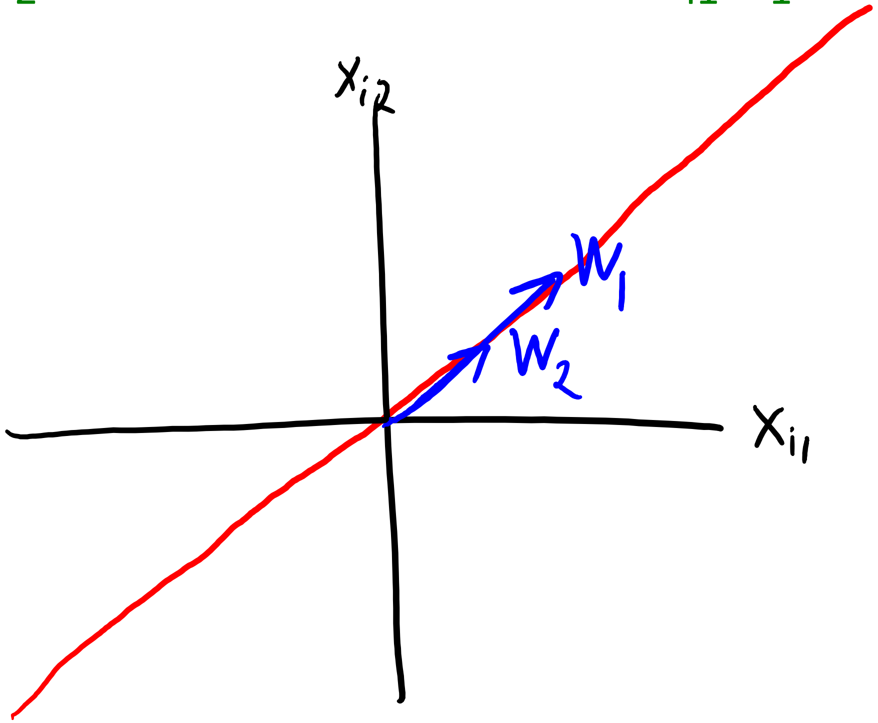
- Consider two vector w_1 and w_2 (k=2).
  - The span(w_1,w_2) is all vectors of form $z_{i1}w_1 + z_{i2}w_2$ for a scalars z_{i1} and z_{i2}.



  - For most non-zero 2d vectors, span(w_1,w_2) is a plane.
    - In the case of two vectors in R^2, the plane will be *all* of R^2.

Span of 2 Vectors

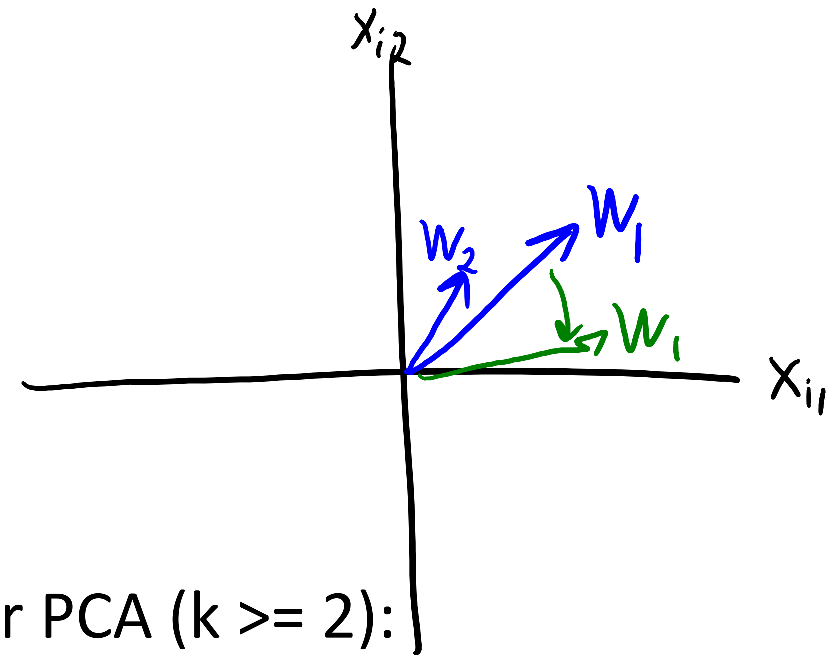
- Consider two vector w_1 and w_2 (k=2).
  - The span(w_1,w_2) is all vectors of form $z_{i1}w_1 + z_{i2}w_2$ for a scalars z_{i1} and z_{i2}.



  - For most non-zero 2d vectors, span(w_1,w_2) is plane.
    - Exception is if w_2 is in span of w_1 ("collinear"), then span(w_1,w_2) is just a line.

Span of 2 Vectors

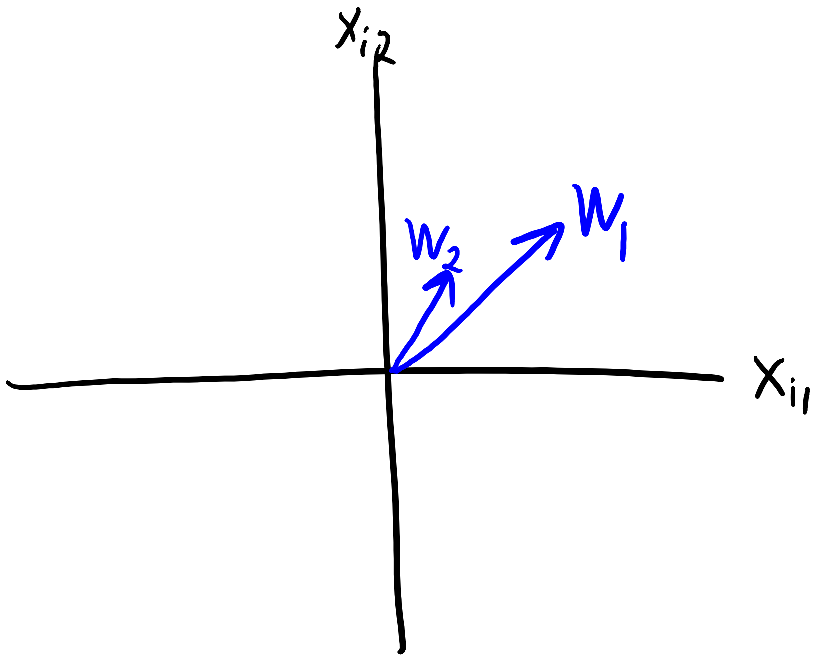
- Consider two vector w_1 and w_2 (k=2).
  - The span(w_1,w_2) is all vectors of form $z_{i1}w_1 + z_{i2}w_2$ for a scalars z_{i1} and z_{i2}.



  - New issues for PCA (k >= 2):
    - We have label switching: span(w_1,w_2) = span(w_2,w_1).
    - We can rotate factors within the plane (if not rotated to be collinear).

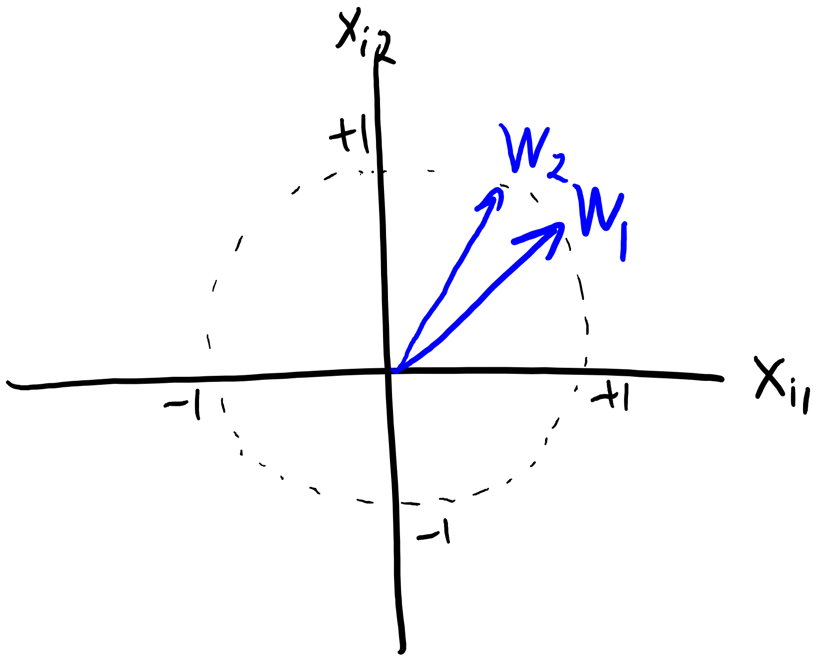
Span of 2 Vectors

- 2 tricks to make vectors defining a plane “more unique”:
  - Normalization: enforce that $||w_1|| = 1$ and $||w_2|| = 1$.



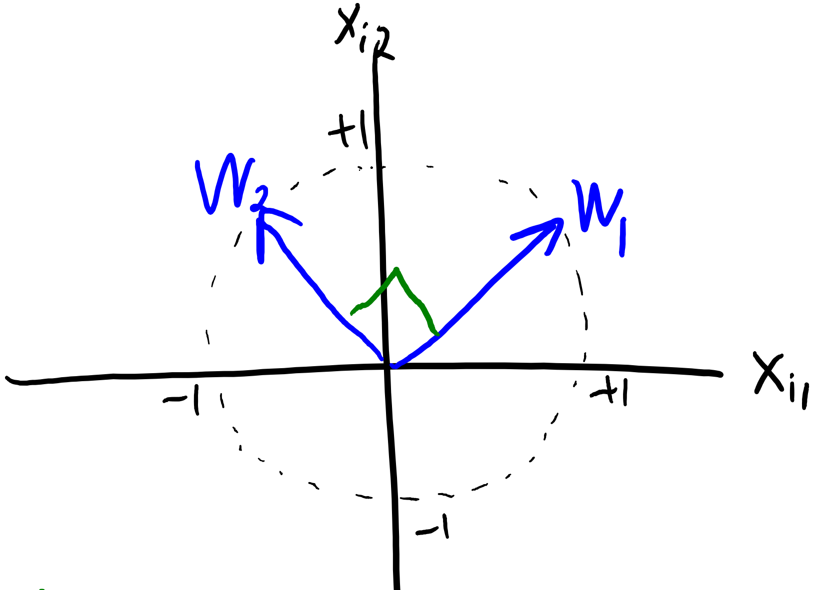
Span of 2 Vectors

- 2 tricks to make vectors defining a plane "more unique":
  - Normalization: enforce that $||w_1|| = 1$ and $||w_2|| = 1$.



Span of 2 Vectors

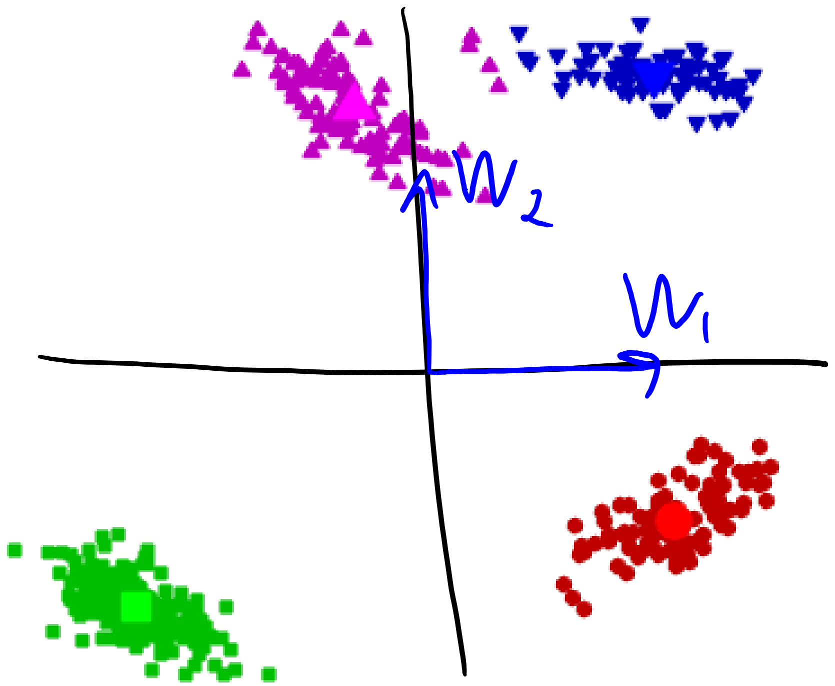
- 2 tricks to make vectors defining a plane "more unique":
  - Normalization: enforce that $||w_1|| = 1$ and $||w_2|| = 1$.
  - Orthogonality: enforce that $w_1^T w_2 = 0$ ("perpendicular").



  - Now I can't grow/shrink vectors (though I can still reflect).
  - Now I can't rotate one vector (but I can still rotate *both*).

Digression: PCA only makes sense for k ≤ d

- Remember our clustering dataset with 4 clusters:



- It doesn't make sense to use PCA with k=4 on this dataset.
  - We only need two vectors [1 0] and [0 1] to exactly represent all 2d points.

Span in Higher Dimensions

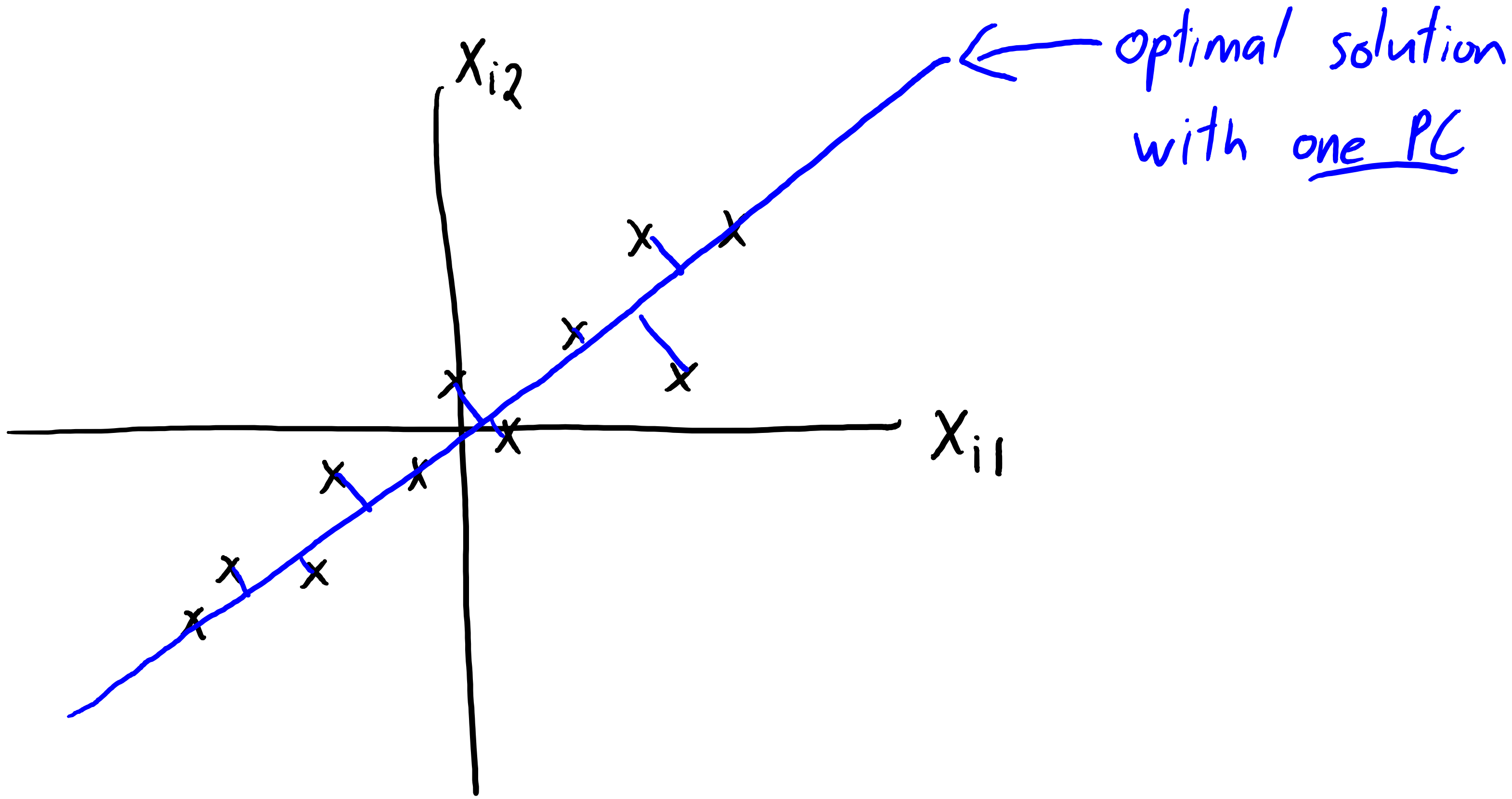
- In higher-dimensional spaces:
  - Span of 1 non-zero vector w_1 is a line.
  - Span of 2 non-zero vectors w_1 and w_2 is a plane (if not collinear).
    - Can be visualized as a 2D plot.
  - Span of 3 non-zeros vectors $\{w_1, w_2, w_3\}$ is a 3d space (if not "coplanar").
  - …
- This is how the W matrix in PCA defines lines, planes, spaces, etc.
  - Each time we increase 'k', we add an extra "dimension" to the subspace.

Making PCA Unique

- We've identified several reasons that optimal W is non-unique:
  - I can multiply any w_c by any non-zero α.
  - I can rotate any w_c almost arbitrarily within the span.
  - I can switch any w_c with any other $w_{c'}$.

- PCA implementations add constraints to make solution unique:
  - Normalization: we enforce that $||w_c|| = 1$.
  - Orthogonality: we enforce that $w_c^T w_{c'} = 0$ for all $c \neq c'$.
  - Sequential fitting: We first fit w_1 ("first principal component") giving a line.
    - Then fit w_2 given w_1 ("second principal component") giving a plane.
    - Then we fit w_3 given w_1 and w_2 ("third principal component") giving a space.

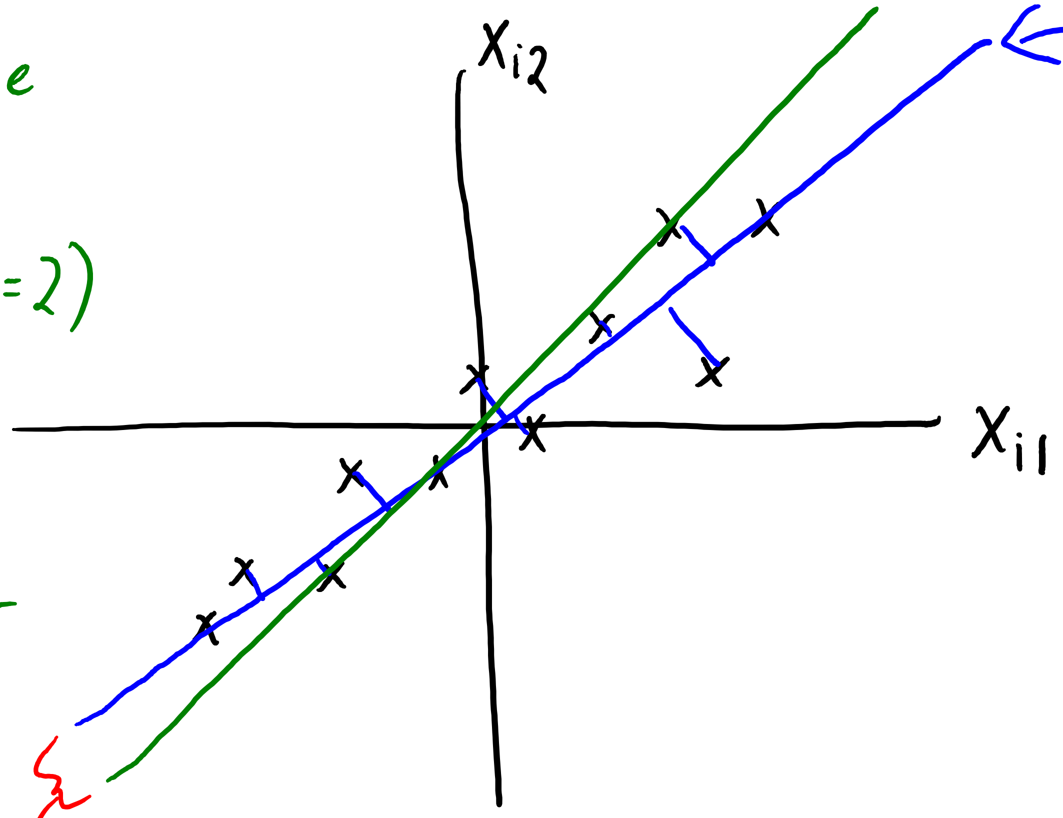
Basis, Orthogonality, Sequential Fitting



Basis, Orthogonality, Sequential Fitting

Any non-parrallel line gives optimal solution to second PC (when $d=2$)

I can get 0 error on every data point.



optimal solution with one PC

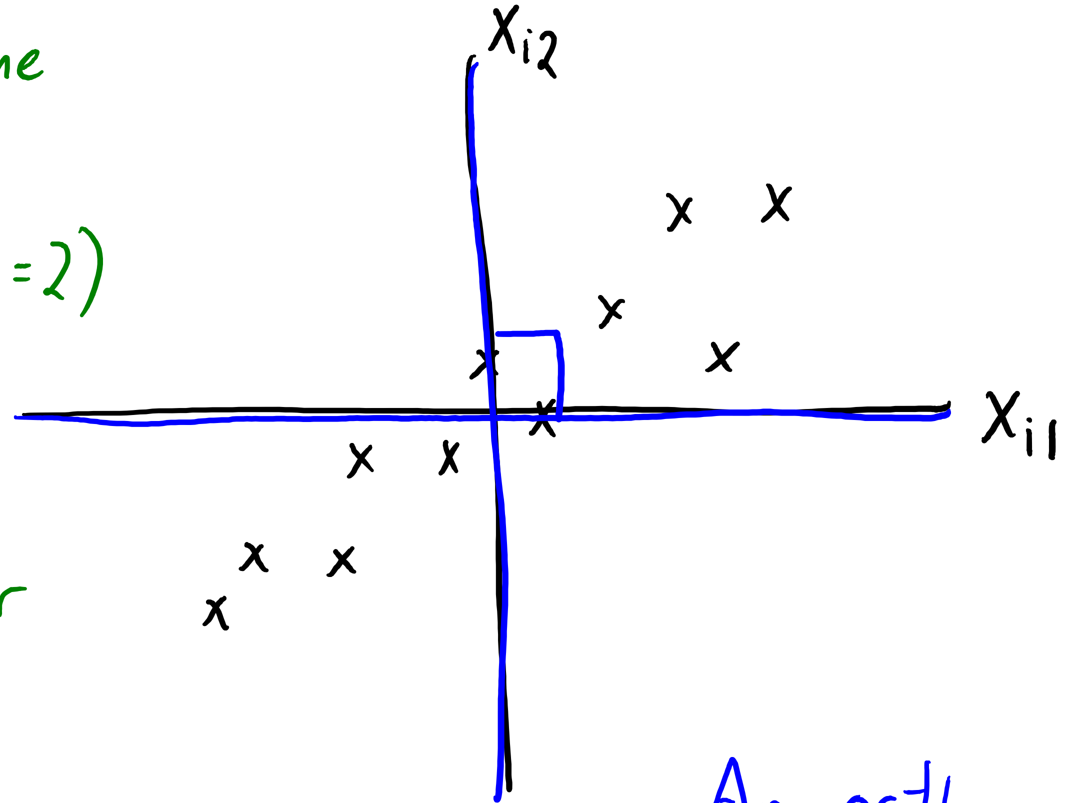
An optimal solution but not orthogonal.

(both PCs give similar information)

Basis, Orthogonality, Sequential Fitting

Any non-parrallel line gives optimal solution to second PC (when $d=2$)

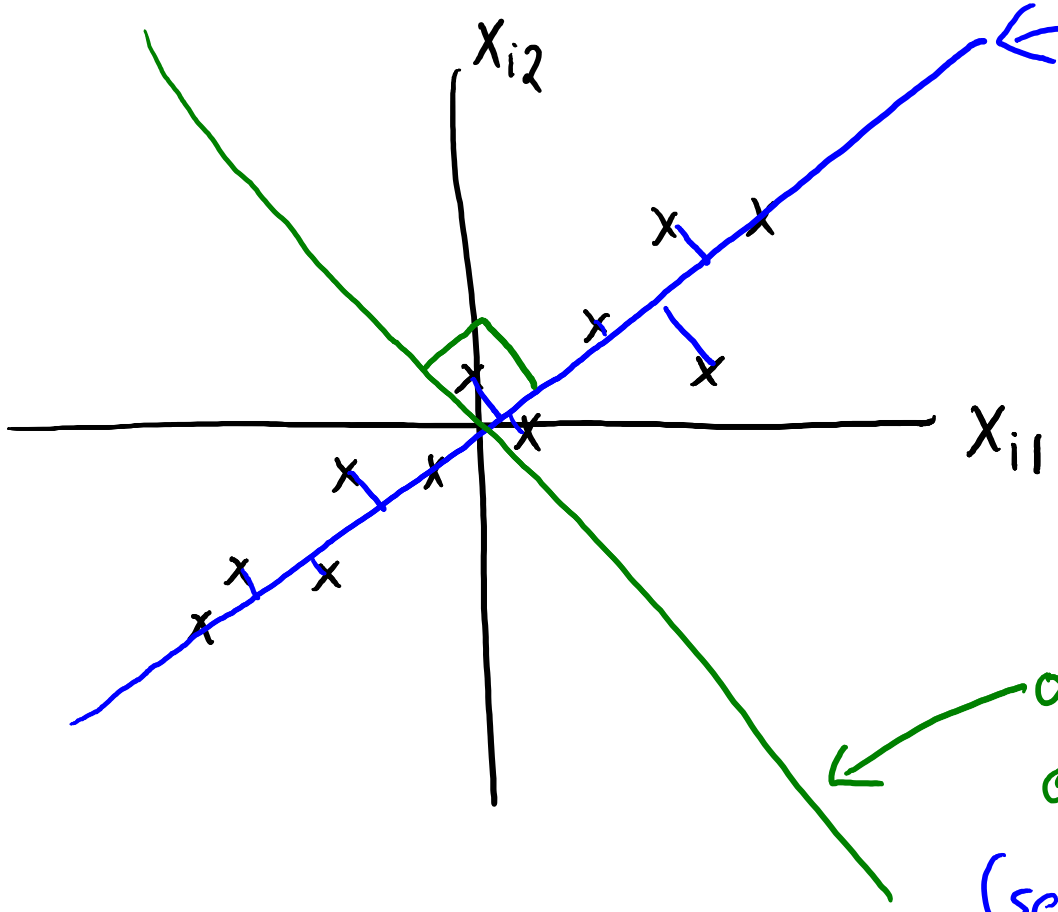
I can get 0 error on every data point.



An orthogonal solution (PCs are not redundant)

but PCs have nothing to do with data

Basis, Orthogonality, Sequential Fitting



optimal solution with one PC ("first" PC)

optimal solution that is orthogonal to "first" PC

(sequential fitting of PCs makes directions unique)

http://setosa.io/ev/principal-component-analysis

PCA with SVD

- How do we fit with normalization/orthogonality/sequential-fitting?
  - It can be done with the "singular value decomposition" (SVD).
  - Take CPSC 302.

- 4 lines of Julia code:
  - mu = mean(X,1)
  - X -= repmat(mu,n,1)
  - (U,S,V) = svd(X)
  - W = V[:,1:k]'

Computing Zhat is cheaper now:

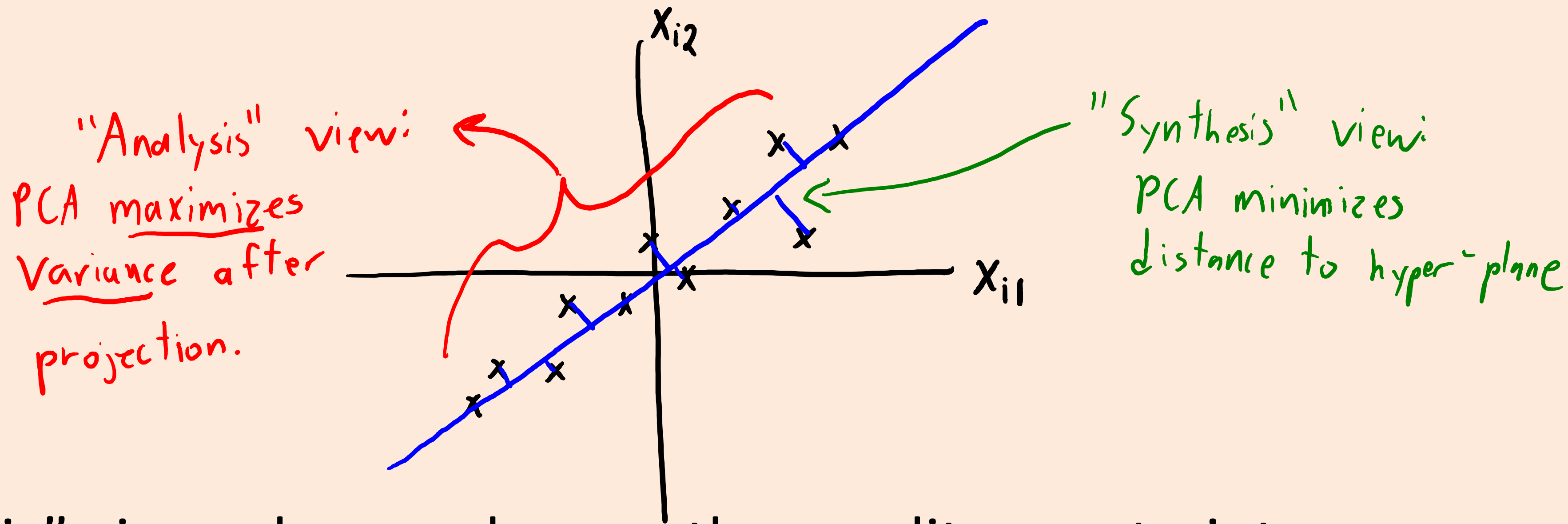
$$\tilde{Z} = \tilde{X} W^T (\underbrace{WW^T}) ^{-1} = X W^T$$

$$WW^T = \begin{bmatrix} - & w_1 & - \\ - & w_2 & - \\ & \vdots & \\ - & w_k & - \end{bmatrix} \begin{bmatrix} | & | & & | \\ w_1^T & w_2^T & \cdots & w_k^T \\ | & | & & | \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & & 0 \\ 0 & & \ddots & & 0 \\ \vdots & & & & \\ 0 & \cdots & & 0 & 1 \end{bmatrix} = I$$

"Synthesis" View vs. "Analysis" View

- We said that PCA finds hyper-plane minimizing distance to data x_i.
  - This is the "synthesis" view of PCA (connects to k-means and least squares).



- "Analysis" view when we have orthogonality constraints:
  - PCA finds hyper-plane maximizing variance in z_i space.
  - You pick W to "explain as much variance in the data" as possible.

Summary

- Alternating minimization and stochastic gradient:
  – Algorithms for minimizing PCA objective.
- Choosing ‘k’:
  – We can choose ‘k’ to explain “percentage of variance” in the data.
- PCA non-uniqueness:
  – Due to scaling, rotation, and label switching.
- Orthogonal basis and sequential fitting of PCs:
  – Leads to non-redundant PCs with unique directions.

- Next time: cancer signatures and NBA shot charts.

Making PCA Unique

- PCA implementations add constraints to make solution unique:
  - Normalization: we enforce that $||w_c|| = 1$.
  - Orthogonality: we enforce that $w_c^T w_{c'} = 0$ for all $c \neq c'$.
  - Sequential fitting: We first fit w_1 ("first principal component") giving a line.
    - Then fit w_2 given w_1 ("second principal component") giving a plane.
    - Then we fit w_3 given w_1 and w_2 ("third principal component") giving a space.
    - …
- Even with all this, the solution is only unique up to sign changes:
  - I can still replace any w_c by $-w_c$:
    - $-w_c$ is normalized, is orthogonal to the other $w_{c'}$, and spans the same space.
  - Possible fix: require that first non-zero element of each w_c is positive.

Proof: “Synthesis” View = “Analysis” View ($WW^T = I$)

- The variance of the z_{ij} (maximized in “analysis” view):

$$\frac{1}{nk}\sum_{i=1}^{n}\|z_i - \mu_z\|^2 = \frac{1}{nk}\sum_{i=1}^{n}\|Wx_i\|^2 \quad (\mu_z = 0 \text{ and } z_i = Wx_i \text{ if } \|w_c\| = 1 \text{ and } w_c^T w_{c'} = 0)$$

$$= \frac{1}{nk}\sum_{i=1}^{n} x_i^T W^T W x_i = \frac{1}{nk}\sum_{i=1}^{n} Tr(x_i^T W^T W x_i) = \frac{1}{nk}\sum_{i=1}^{n} Tr(W^T W x_i x_i^T)$$

“cyclic” property of trace

$$= \frac{1}{nk} Tr\Big(W^T W \underbrace{\sum_{i=1}^{n} x_i x_i^T}_{X^TX}\Big) = \frac{1}{nk} Tr(W^T W X^T X)$$

linearity of trace

- The distance to the hyper-plane (minimized in “synthesis” view):

$$\|\underbrace{ZW}_{=XW^T} - X\|_F^2 = \|XW^TW - X\|_F^2 = Tr\big((XW^TW - X)^T(XW^TW - X)\big)$$

$\|A\|_F^2 = Tr(A^TA)$

$$= Tr(W^TWX^TXW^TW) - 2Tr(W^TWX^TX) + Tr(X^TX)$$

$$= Tr(W^T \underbrace{WW^T}_{I} WX^TX) - 2Tr(W^TWX^TX) + Tr(X^TX)$$

$$= -Tr(W^TWX^TX) + (\text{constant})$$

Solved by same ‘W’

Probabilistic PCA

- With zero-mean ("centered") data, in PCA we assume that

$$x_i \approx W^T z_i$$

- In probabilistic PCA we assume that

$$x_i \sim \mathcal{N}(W^T z_i, \sigma^2 I) \qquad z_i \sim \mathcal{N}(0, I)$$

- Integrating over 'Z' the marginal likelihood given 'W' is Gaussian,

$$x_i \mid W \sim \mathcal{N}(0, W^T W + \sigma^2 I)$$

- Regular PCA is obtained as the limit of σ^2 going to 0.

Generalizations of Probabilistic PCA

- Probabilistic PCA model:

$$x_i \mid W \sim \mathcal{N}(0, W^TW + \sigma^2 I)$$

- Why do we need a probabilistic interpretation?

- Shows that PCA fits a Gaussian with restricted covariance.
  - Hope is that $W^TW + \sigma^2 I$ is a good approximation of X^TX.

- Gives precise connection between PCA and factor analysis.

Factor Analysis

- Factor analysis is a method for discovering latent factors.
- Historical applications are measures of intelligence and personality.

Trait	Description
Openness	Being curious, original, intellectual, creative, and open to new ideas.
Conscientiousness	Being organized, systematic, punctual, achievement-oriented, and dependable.
Extraversion	Being outgoing, talkative, sociable, and enjoying social situations.
Agreeableness	Being affable, tolerant, sensitive, trusting, kind, and warm.
Neuroticism	Being anxious, irritable, temperamental, and moody.

- A standard tool and widely-used across science and engineering.

https://new.edu/resources/big-5-personality-traits

PCA vs. Factor Analysis

- PCA and FA both write the matrix 'X' as

$$X \approx ZW$$

- PCA and FA are both based on a Gaussian assumption.

- Are PCA and FA the same?
  - Both are more than 100 years old.
  - People are still arguing about whether they are the same:
    - Doesn't help that some packages run PCA when you call their FA method.

Google

pca vs. factor analysis

All Images Videos News Maps More Search tools

About 358,000 results (0.17 seconds)

[PDF] Principal Component Analysis versus Exploratory Factor ...
www2.sas.com/proceedings/sugi30/203-30.pdf
by DD Suhr - Cited by 118 - Related articles
1. Paper 203-30. **Principal Component Analysis vs.** Exploratory **Factor Analysis.** Diana D. Suhr, Ph.D. University of Northern Colorado. Abstract. **Principal** ...

pca - What are the differences between Factor Analysis and ...
stats.stackexchange.com/.../what-are-the-differences-between-**factor-anal**...
Aug 12, 2010 - **Principal Component Analysis** (PCA) and Common **Factor Analysis** (CFA) ..... differently one has to interpret the strength of loadings in **PCA vs.**

What are the differences between principal components ...
support.minitab.com/...**factor-analysis**/differences-between-**pca-and-facto**...
Principal Components Analysis and **Factor Analysis** are similar because both procedures are used to simplify the structure of a set of variables. However, the ...

[PDF] Principal Components Analysis - UNT
https://www.unt.edu/rss/class/.../Principal%20Components%20**Analysis**.p...
PCA vs. Factor Analysis. • It is easy to make the mistake in assuming that these are the same techniques, though in some ways exploratory factor analysis and ...

Factor analysis versus Principal Components Analysis (PCA)
psych.wisc.edu/henriques/**pca**.html
Jun 19, 2010 - **Factor analysis versus PCA.** These techniques are typically used to analyze groups of correlated variables representing one or more common ...

[PDF] Principal Component Analysis and Factor Analysis
www.stats.ox.ac.uk/~ripley/MultAnal_HT2007/PC-FA.pdf
where D is diagonal with non-negative and decreasing values and U and V ..... **Factor analysis** and **PCA** are often confused, and indeed SPSS has **PCA** as.

How can I decide between using principal components ...
https://www.researchgate.net/.../How_can_I_decide_between_using_prin...
Factor analysis (FA) is a group of statistical methods used to understand and simplify patterns ... Retrieved from http://pareonline.net/getvn.asp?v=10&n=7 ... **Principal component analysis** (PCA) is a method of factor extraction (the second step ...

[PDF] Exploratory Factor Analysis and Principal Component An...
www.lesahoffman.com/948/948_Lecture2_EFA_**PCA**.pdf
2 very different schools of thought on exploratory **factor analysis** (EFA) vs. principal components **analysis** (PCA): ➢ EFA and PCA are TWO ENTIRELY ...

Factor analysis - Wikipedia, the free encyclopedia
https://en.wikipedia.org/wiki/**Factor_analysis**
Jump to Exploratory **factor analysis versus** principal components **...** - [edit]. See also: **Principal component analysis** and Exploratory **factor analysis**.

[PDF] The Truth about PCA and Factor Analysis
www.stat.cmu.edu/~cshalizi/350/lectures/13/lecture-13.pdf
Sep 28, 2009 - nents and **factor analysis**, we'll wrap up by looking at their uses and

PCA vs. Factor Analysis

- In probabilistic PCA we assume:

$$x_i \sim \mathcal{N}(W^T z_i, \sigma^2 I)$$

- In FA we assume for a diagonal matrix D that:

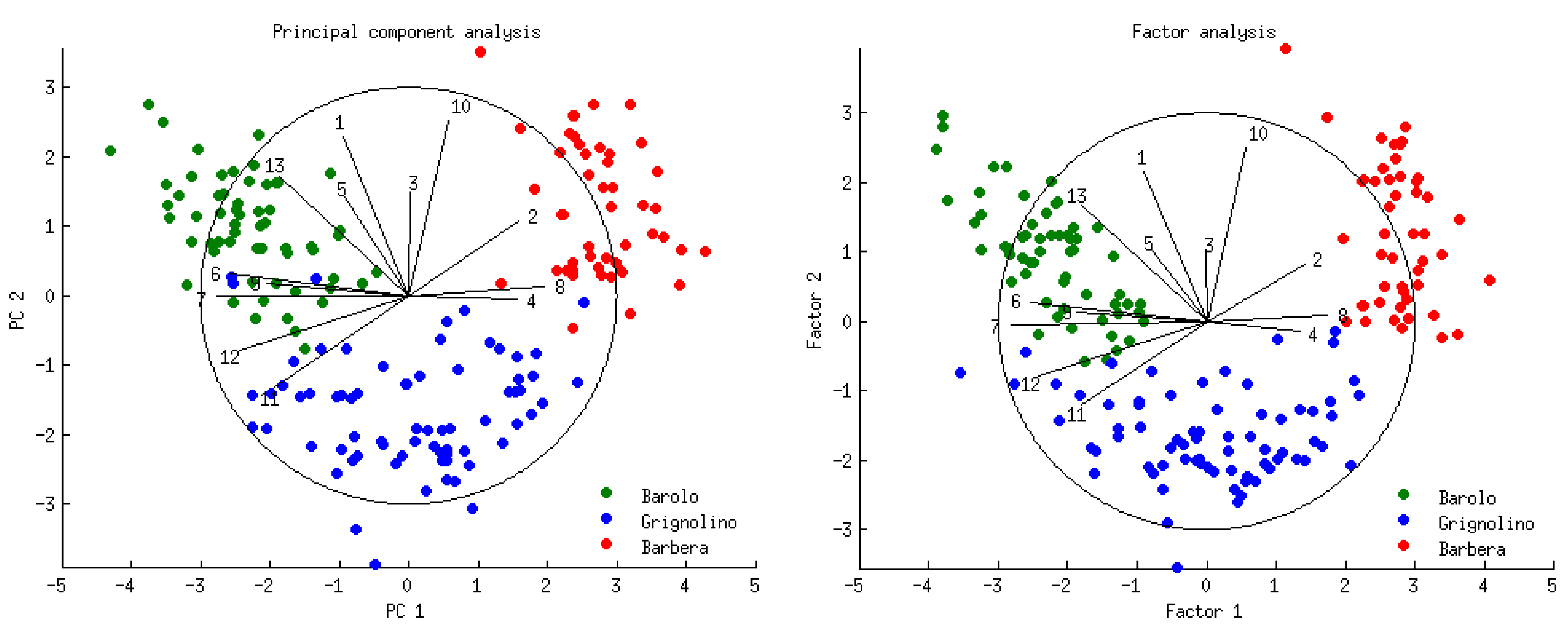
$$x_i \sim \mathcal{N}(W^T z_i, D)$$

- The posterior in this case is: $x_i \mid W \sim \mathcal{N}(0, W^T W + D)$

- The difference is you have a noise variance for each dimension.
  - FA has extra degrees of freedom.

PCA vs. Factor Analysis

- In practice there often isn't a huge difference:



http://stats.stackexchange.com/questions/1576/what-are-the-differences-between-factor-analysis-and-principal-component-analysi

Factor Analysis Discussion

- Differences with PCA:
  - Unlike PCA, FA is not affected by scaling individual features.
  - But unlike PCA, it's affected by rotation of the data.
  - No nice "SVD" approach for FA, you can get different local optima.

- Similar to PCA, FA is invariant to rotation of 'W'.
  - So as with PCA you can't interpret multiple factors as being unique.

Motivation for ICA

- Factor analysis has found an enormous number of applications.
  – People really want to find the "hidden factors" that make up their data.

- But PCA and FA can't identify the factors.

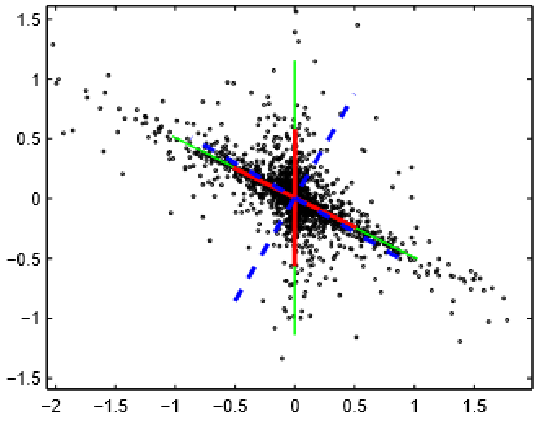


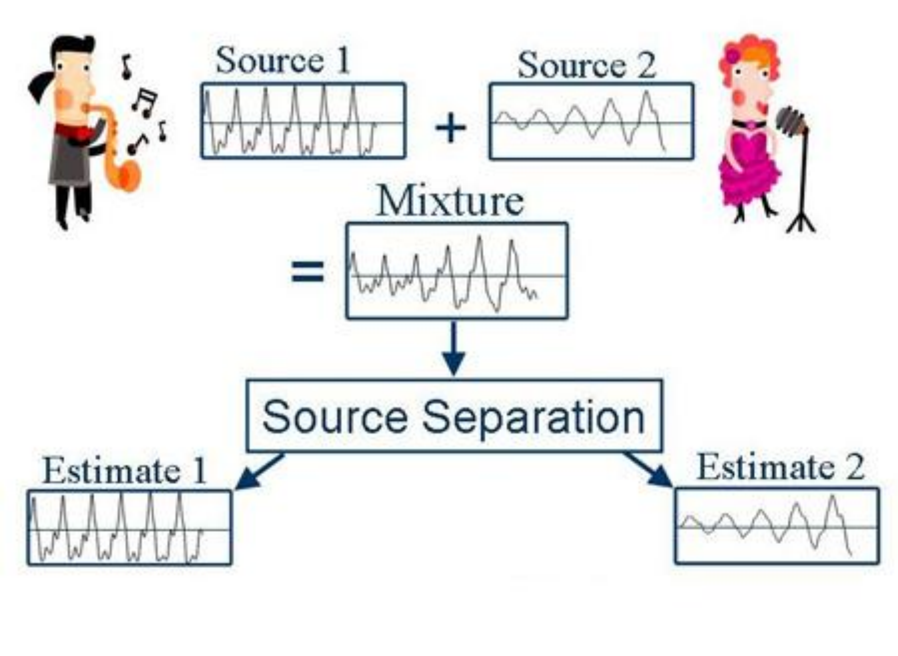
Figure : Latent data is sampled from the prior $p(x_i) \propto \exp(-5\sqrt{|x_i|})$ with the mixing matrix $\mathbf{A}$ shown in green to create the observed two dimensional vectors $\mathbf{y} = \mathbf{A}\mathbf{x}$. The red lines are the mixing matrix estimated by `ica.m` based on the observations. For comparison, PCA produces the blue (dashed) components. Note that the components have been scaled to improve visualisation. As expected, PCA finds the orthogonal directions of maximal variation. ICA however, correctly estimates the directions in which the components were independently generated.

Motivation for ICA

- Factor analysis has found an enormous number of applications.
  - People really want to find the "hidden factors" that make up their data.
- But PCA and FA can't identify the factors.
  - We can rotate W and obtain the same model.
- Independent component analysis (ICA) is a more recent approach.
  - Around 30 years old instead of > 100.
  - Under certain assumptions it can identify factors.
- The canonical application of ICA is blind source separation.

Blind Source Separation

- Input to blind source separation:
  - Multiple microphones recording multiple sources.



- Each microphone gets different mixture of the sources.
  - Goal is reconstruct sources (factors) from the measurements.

http://music.eecs.northwestern.edu/research.php

Independent Component Analysis Applications

- ICA is replacing PCA and FA in many applications:

Some ICA applications are listed below:[1]

- optical Imaging of neurons[17]
- neuronal spike sorting[18]
- face recognition[19]
- modeling receptive fields of primary visual neurons[20]
- predicting stock market prices[21]
- mobile phone communications [22]
- color based detection of the ripeness of tomatoes[23]
- removing artifacts, such as eye blinks, from EEG data.[24]

- Recent work shows that ICA can often resolve direction of causality.

https://en.wikipedia.org/wiki/Independent_component_analysis#Applications

Limitations of Matrix Factorization

- ICA is a matrix factorization method like PCA/FA,

$$X = ZW$$

- Let's assume that X = ZW for a "true" W with k = d.
  - Different from PCA where we assume k ≤ d.

- There are only 3 issues stopping us from finding "true" W.

3 Sources of Matrix Factorization Non-Uniquness

- Label switching: get same model if we permute rows of W.
  - We can exchange row 1 and 2 of W (and same columns of Z).
  - Not a problem because we don't care about order of factors.
- Scaling: get same model if you scale a row.
  - If we mutiply row 1 of W by α, could multiply column 1 of Z by $1/\alpha$.
  - Can't identify sign/scale, but might hope to identify direction.
- Rotation: get same model if we rotate W.
  - Rotations correspond to orthogonal matrices Q, such matrices have $Q^TQ = I$.
  - If we rotate W with Q, then we have $(QW)^TQW = W^TQ^TQW = W^TW$.
- If we could address rotation, we could identify the "true" directions.

A Unique Gaussian Property

- Consider an independent prior on each latent features z_c.
  - E.g., in PPCA and FA we use $N(0,1)$ for each z_c.

- If prior $p(z)$ is independent and rotation-invariant ($p(Qz) = p(z)$), then it must be Gaussian (only Gaussians have this property).

- The (non-intuitive) magic behind ICA:
  - If the priors are all non-Gaussian, it isn't rotationally symmetric.
  - In this case, we can identify factors W (up to permutations and scalings).

PCA vs. ICA

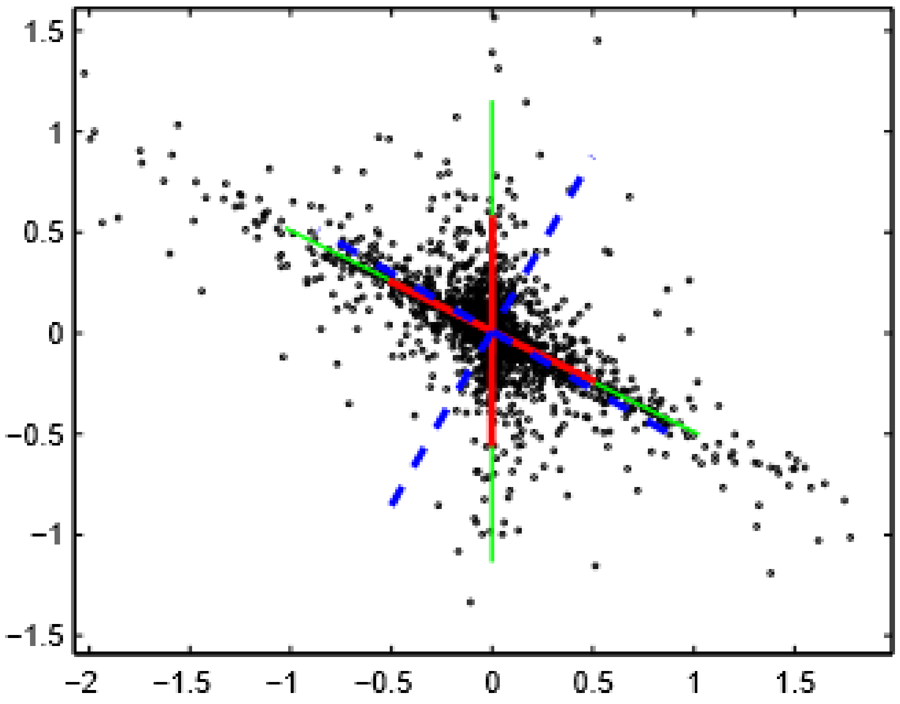


Figure : Latent data is sampled from the prior $p(x_i) \propto \exp(-5\sqrt{|x_i|})$ with the mixing matrix $\mathbf{A}$ shown in green to create the observed two dimensional vectors $\mathbf{y} = \mathbf{A}\mathbf{x}$. The red lines are the mixing matrix estimated by `ica.m` based on the observations. For comparison, PCA produces the blue (dashed) components. Note that the components have been scaled to improve visualisation. As expected, PCA finds the orthogonal directions of maximal variation. ICA however, correctly estimates the directions in which the components were independently generated.

http://www.inf.ed.ac.uk/teaching/courses/pmr/lectures/ica.pdf

Independent Component Analysis

- In ICA we approximate X with ZW,
  assuming $p(z_{ic})$ are non-Gaussian.

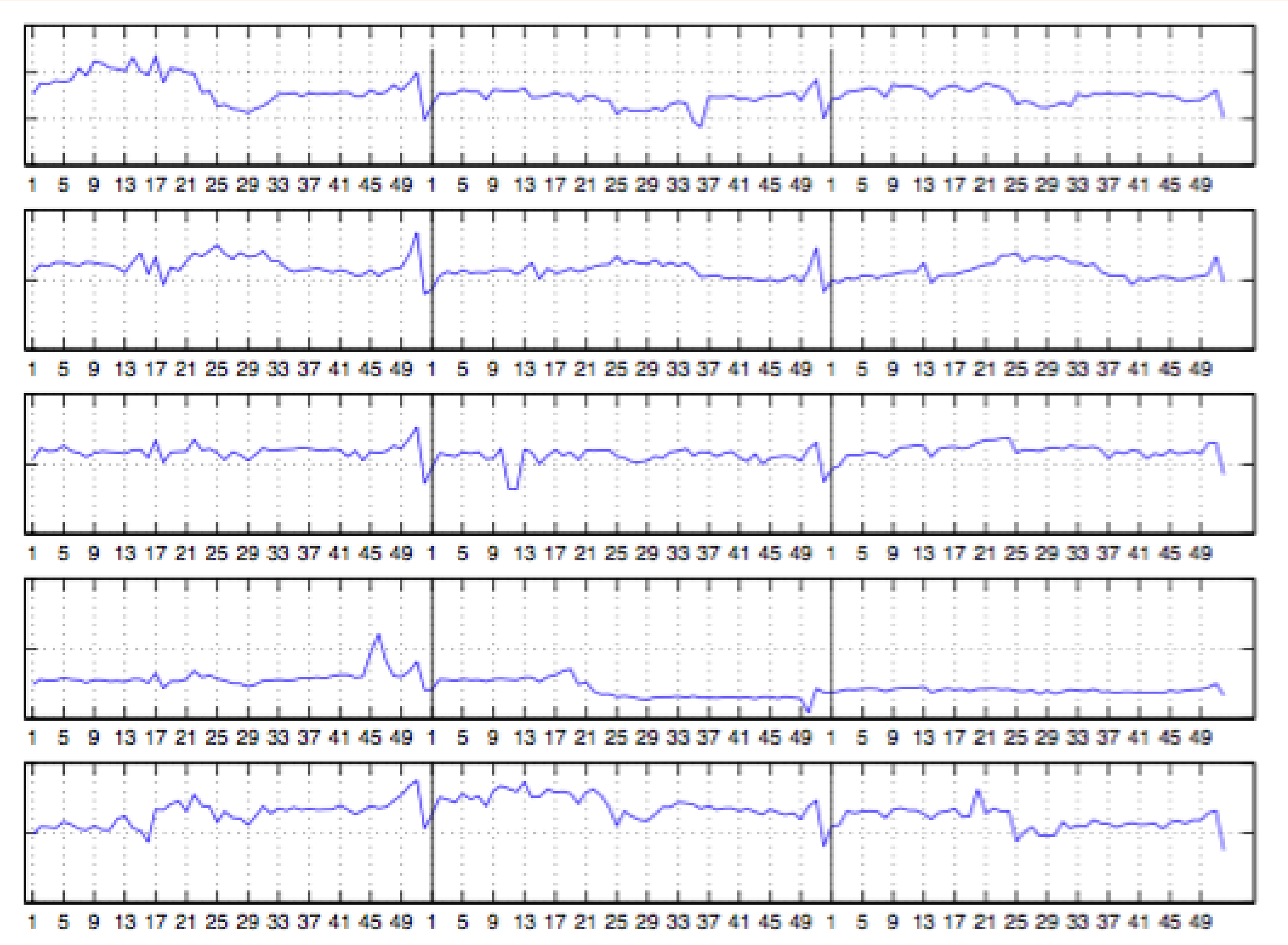
- Usually we "center" and "whiten" the data before applying ICA.

- There are several penalties that encourage non-Gaussianity:
  - Penalize low kurtosis, since kurtosis is minimized by Gaussians.
  - Penalize high entropy, since entropy is maximized by Gaussians.

- The fastICA is a popular method maximizing kurtosis.

ICA on Retail Purchase Data

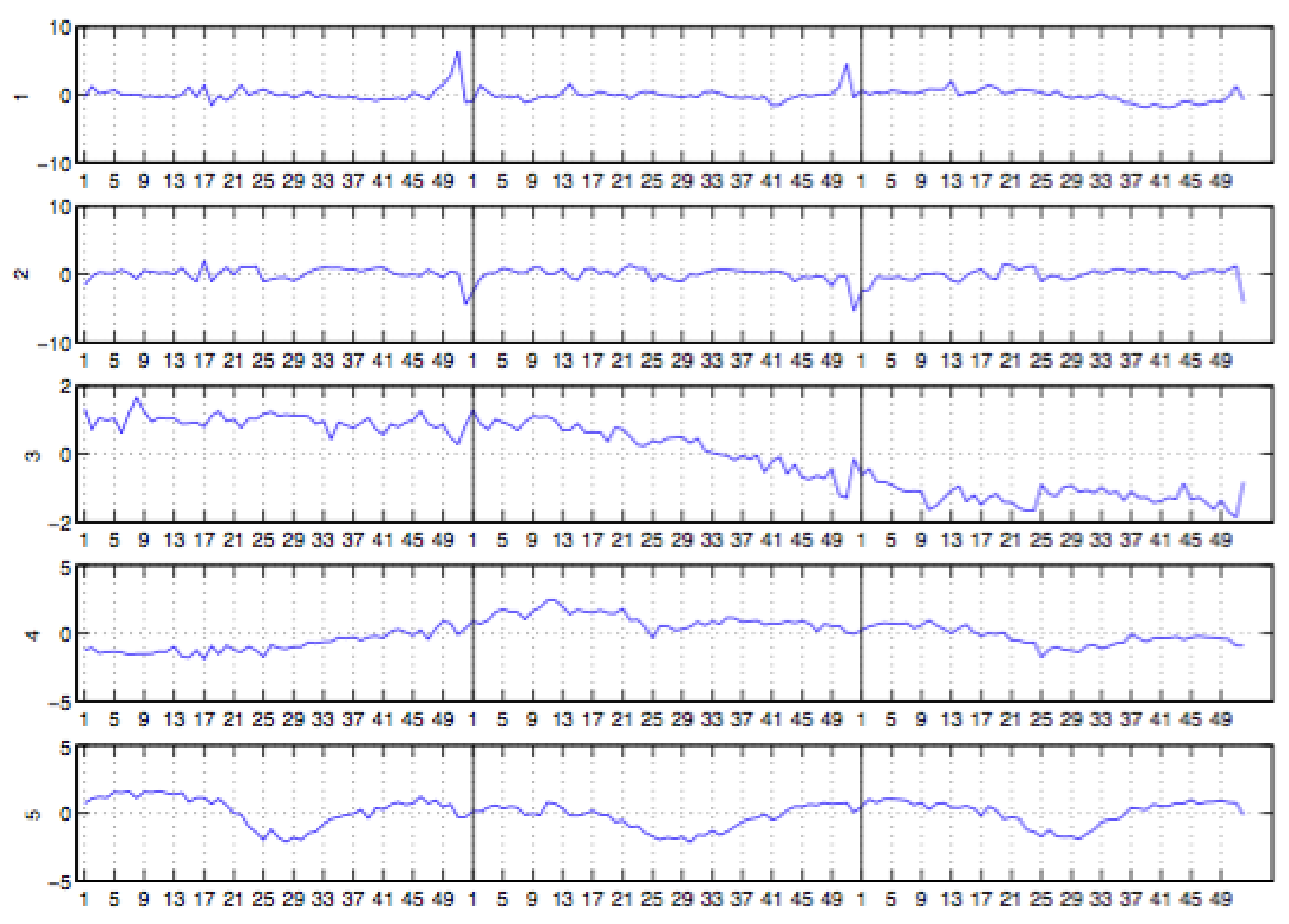
- Cash flow from 5 stores over 3 years:



http://www.stat.ucla.edu/~yuille/courses/Stat161-261-Spring14/HyvO00-icatut.pdf

ICA on Retail Purchase Data

- Factors found using ICA:



http://www.stat.ucla.edu/~yuille/courses/Stat161-261-Spring14/HyvO00-icatut.pdf