I learned about independent component analysis, generalized linear model, principal component analysis, and factor analysis.

General Linear Model

The general linear model is when the output is a vector instead of a scalar.

A general linear model can be used to see the effect of block design on signal in voxels - to highlight which voxels which are activated by a block design.

A signal:

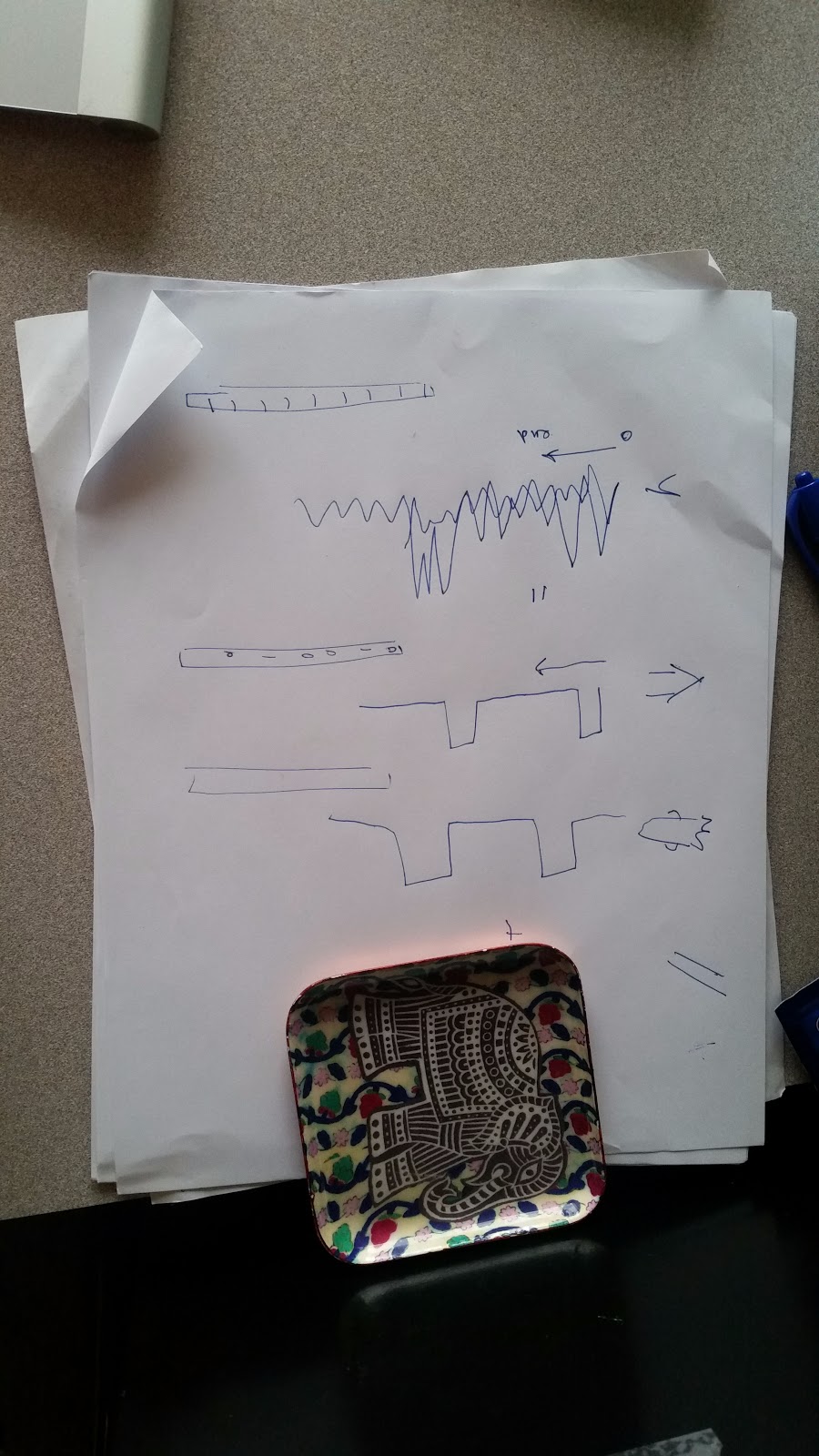
[1 0 4 -5 6 7 -8 -9 0 2 3 8 -8 -4 0]

Can be decomposed into the following weighted block designs (1 = stimuli presented, 0 = no stimuli presented)

[1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0]

[0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0]

* Error

 - cat’s drawings

Principal Component Analysis

Namely, PCA is different from ICA and FA because PCA is not creating a model or rigid fit for data. PCA doesn’t use any free parameters/ error parameters. PCA is a strict rotation of data to a new set of coordinate axes (i.e. the eigenvectors of the variance-covariance matrix).

To explain more on the process of PCA

1. Rigid rotation of data onto new orthogonal axes (the eigenvectors of the variance-covariance matrix)
2. Projection of original dimensions onto orthogonal axes (new dimensions) to see how much variance is accounted for by an original dimension
   1. The original dimensions with larger projects account for more variance!

Cat’s example (consider the following as a motivating example):

* 8 dimensions/ factors that can account for the results of a task (size of the house, context of the image presented, recognition of the object)
* PCA to determine that most of the variance of the results of the task are captured by just 2 dimensions/ factors (so maybe house size and house color account for the perception of other variables too)
* Why use PCA? To reduce the redundancy of data and to capture the variance of data in fewer dimensions
* TL;DR: Use **PCA to reduce factor redundancy and reduce dimensionality**!!
  + Leah mentioned that Gilles might want to represent complex data in a graph and might need fewer dimensions than he currently has … room for application much? Say you have 8 scores to present but 2 of the capture 90% of the variance - then you’re set man! Use those 2 scores and present a simple xy graph instead of an 8 dimensional visualization

Tying Linear Algebra to PCA

**Why have eigenvalues?**

* Eigenvalues of an eigenvector correspond / are proportional to the amount of variance captured by that vector (eigenvalues = d1, d2, d3; eigenvectors = v1, v2, v3)
* Variance captured by vx = dx / (d1 + d2 + d3)
* Isn’t that amazing?

**Symmetric matrices, orthogonal eigenbases**

* In linear algebra, we learn that a symmetric matrix have orthogonal eigenbasis - this fact is crucial to how PCA, which is a decomposition of a symmetric variance-covariance matrix into eigenvectors and eigenvalues yield orthogonal eigenvectors and eigenvalues that are independent :) - isn’t that amazing? Linear algebra ties into and is leveraged heavily in the properties of decomposition

Much excitement

**To explain PCA to someone who hasn’t used it before**, best to start with a 2-dimensional case and then show which dimension captures more of the data from there - then generalize to the 3D and XD case

* Motivating example: two tests, a math test and a reading test - plot scores of each student on a 2D graph. Capture variance linearly. Show why the 1st eigenvector with the biggest eigenvalue accounts for the most variance (because if you plot the data on a line of that first eigenvector, you capture more variance than if you plot the data on two dimensions). Project dimensions onto the eigenvector and notice which dimension captures the most variance.
  + Beyond the first eigenvector, the next eigenvector with the next largest eigenvalue also captures some variance (how much? Variance captured by vx = dx / (d1 + d2 + d3 + ...+ dn) where d1 - dn are the eigenvalues)
* Cool but irrelevant to most people sidenote - might be important to explain before introducing eigenvalues:
  + The eigenvectors of the data are orthogonal b/c symmetric matrix; thus, these eigenvectors are simply a rotation.
* Sometimes, we call the eigenvectors that capture the most variance, the characteristic vectors!! The dimensions that capture the most variance have the largest projections onto characteristic vectors

**Okay, finally, what’s the outcome of PCA and how does it relate to SVD?**

7/21/17

**Notes on SVD**:

[Intuitive explanation](https://math.stackexchange.com/questions/3869/what-is-the-intuitive-relationship-between-svd-and-pca) of how SVD is already built into PCA

* SVD returns the eigenvectors returned by PCA along with the sqrt of eigenvalues returned by PCA
* SVD decomposes the data matrix A into USV’ where
  + U col’s = eigenvectors of AA’
  + V = A’A
  + S = diagonal matrix, sqrt of eigenvalues of AA’
* PCA decomposes variance-covariance matrix AA’ into WDW’ or U(S^2)U’
  + W = U → col’s = eigenvectors of AA’
  + D = S^2 = diagnonal matrix, eigenvalues of AA’
* TL; DR:
  + SVD: A = USV’
  + PCA: AA’ = U(S^2)U’

[More Notes](https://mitpress.mit.edu/books/statistical-analysis-fmri-data) on PCA in fMRI:

* An assumption about PCA in fMRI is:
  + Any single source of noise will produce normally distributed activity fluctuations that only account for a small amount of variance
  + THUS, PCA can be used to find eigenvectors that account for the smallest amount of variance AND THEN the eigenvectors that account for noticeably variance than others eigenvectors would be removed from all data
  + You can decide how many low variance capturing vectors to remove based on plotting eigenvalue vs. eigenvalue rank and seeing when there’s a blimp indicating that before the blimp you were probably looking at signal whereas after the blimp you were probably looking at noise
* Types of distributions in PCA and ICA
  + PCA assumes that signal data AND noise data is normally distributed
  + ICA assumes that signal data is not normally distributed while noise data is
    - Evidence would suggest signal data is not normally distributed, drawing into question the use of PCA for things other than noise filtering in fMRI
  + In general, PCA assumes data is multivariate normal distribution

[ICA](https://mitpress.mit.edu/books/statistical-analysis-fmri-data) : wow this is very cool

* What does ICA actually give you as components?
* Consider two types of ICA - temporal ICA and spatial ICA
* Temporal ICA is what you read about in the cocktail problem
  + 4 speakers, 4 microphones
  + Input = track on 4 microphones
  + Output = track of 4 individual speakers s.t. If you weight these tracks you get each of the 4 microphone tracks
* Temporal ICA for fMRI
  + 90,000 voxels, 240 TRs
  + Input = @ each voxel, the activity over the course of all TRs
  + Output = @ each voxel, the activity can be computed as a weighted sum of fundamental activity timecourses
* Spatial ICA is what is often used in fMRI data
  + 90,000 voxels, 240 TRs
  + Input = @ each TR, the activations of the entire brain (over all voxels)
  + Output = @ each TR, the activations can be computed as a weighted sum of spatial network maps (ex. DMN binary map, stimuli 1 activation map, stimuli 2 activation map)
* UHH ok, you gave me the math and the concepts but the components still have incomplete information, namely:
  + **Assigning Variance (Relative Importance)** - can’t use eigenvalues so what do you use to gauge importance of component?
    - Part of ICA was to ‘whiten’ the variance-covariance matrix to reduce computation time and number of outputted components
    - Whitening involves reducing covariance (non scaled correlation) to 0 and variance to 1
    - Computing variance is lost in the process of ICA but we can still get a sense of how much variation each component captures
    - If you remove a component and look at the resulting matrix - then compare the before and after matrices by calculated a sum of square difference (SSD), then the variance captured by that component is:
      * SSD\_i/ sum of SSD\_i
  + **Naming the Components** **(Assigning Meaning to Components)** -
    - Components can be determined based on the following assumption
      * Activation will occur in the same pattern the stimuli occur
    - For any spatial map component that you’re trying to find the meaning of
      * evaluate a correlation between the weights of each spatial map over time and all possible stimuli occurrences over time
      * Eg. spatial map: .3 .7 .8 .2 .1 .9
      * Eg. stimuli:    0  1   0  1  0  1
    - How do components work if … you don’t have a stimuli vector? I.e. how are components meaningfully understood using resting state data?
* TL; DR: ICA creates basis vectors for inputs
  + How do the components of ICA differ from the components of PCA? [outputs of component analysis]
    - ICA components include error/ free parameters so ICA captures 100% of the variance while PCA is not a perfect model and captures as much variance as possible without any free parameters
    - ICA components are statistically independent!
      * What is statistical independence?
      * That means that information about random variable X will not provide any more information about random variable Y
      * So how do you measure statistical independence?
      * I thought statistical independence had to do with probability
      * Statistical independence as a proof means that f\_xy(x,y) = f\_x(x) \* f\_y(y) where f is a probability density function (you can compute this by plotting frequencies of each variable input)
      * But proving statistical independence is nearly impossible so instead ICA finds statistically independent components in a way that has necessary but not sufficient measures of statistical independence
        + e.g. the correlation between x and y needs to be 0 for statistical independence but isn’t enough for statistical independence
        + The correlation only checks for a 1st order and 2nd order relationship between x and y while statistical independence requires that no relationship exists of ANY order
        + What is order? The largest exponent on the expected value of a variable (mean = E(x) → order = 1; variance = E(x - mean) ^ 2 → order = 2)
* How are the data put into ICA different from the data put into PCA? [inputs of component analysis]

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Statistically independent | orthogonal | Correlated (same as orthogonal) | Normally distributed | Non-normally distributed (super-gaussian) |
| ICA | x |  |  |  | x |
| PCA |  | x | x | x |  |

* How does ICA work with images that look more crisp?
  + Same idea of noise filtering as with PCA
  + The assumption is that a single source of noise accounts for little variation in the data so you can try to eliminate the components that account for little variation in the data