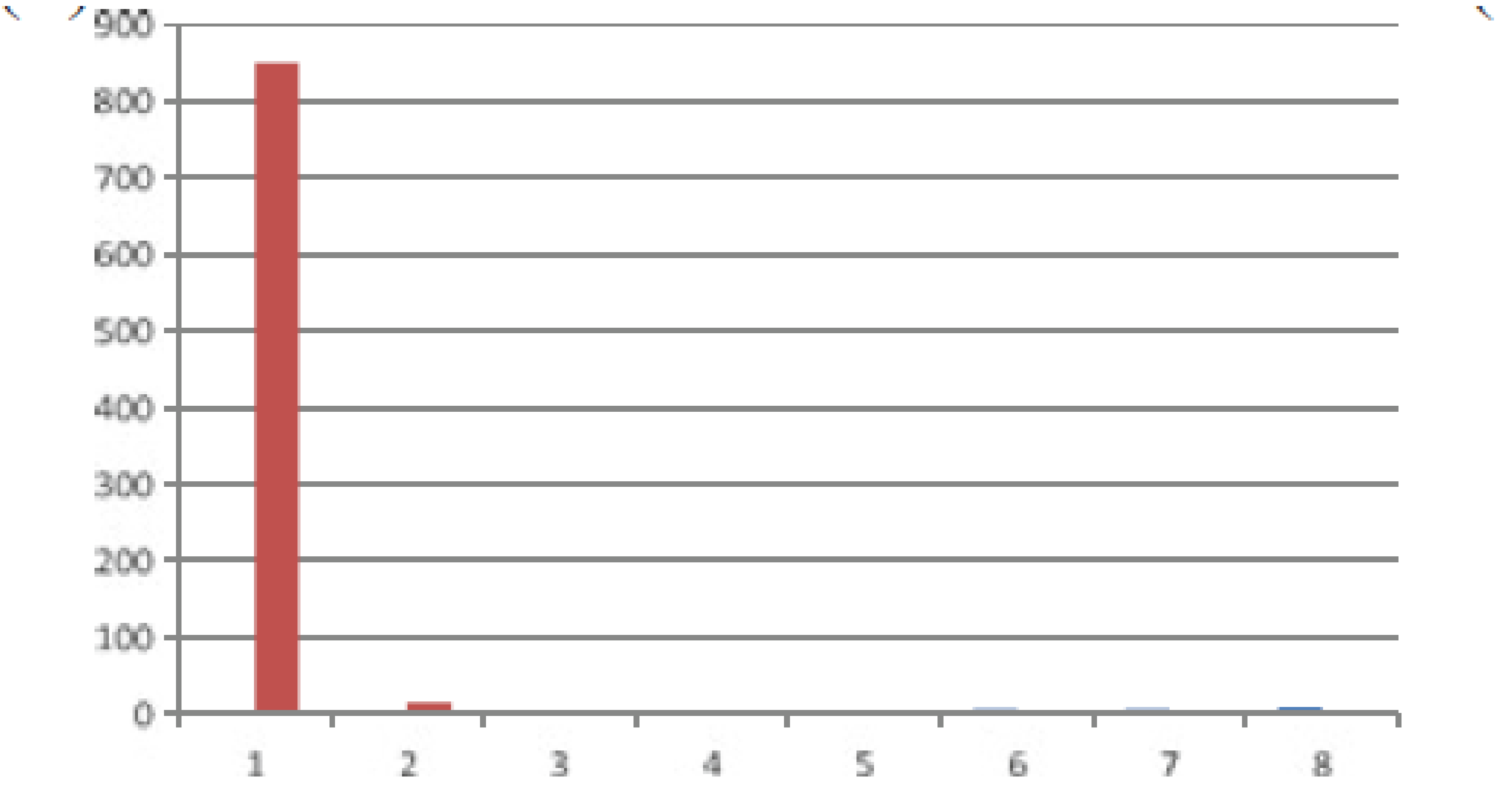
Data Pre-processing

Transformation and Aggregation

Things To Consider When Dealing With Non-normal Data

- How to deal with skewed data?

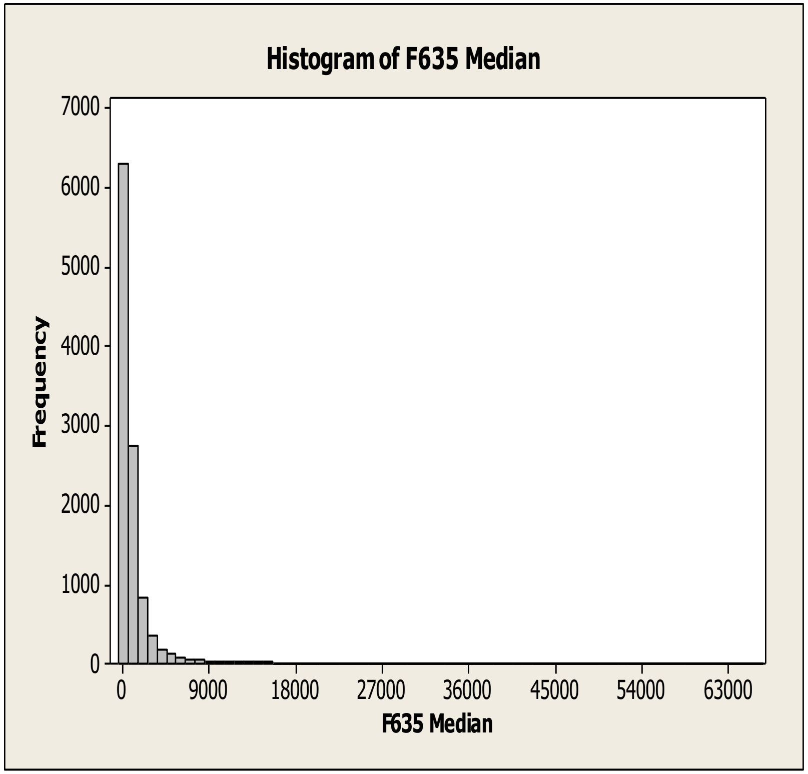


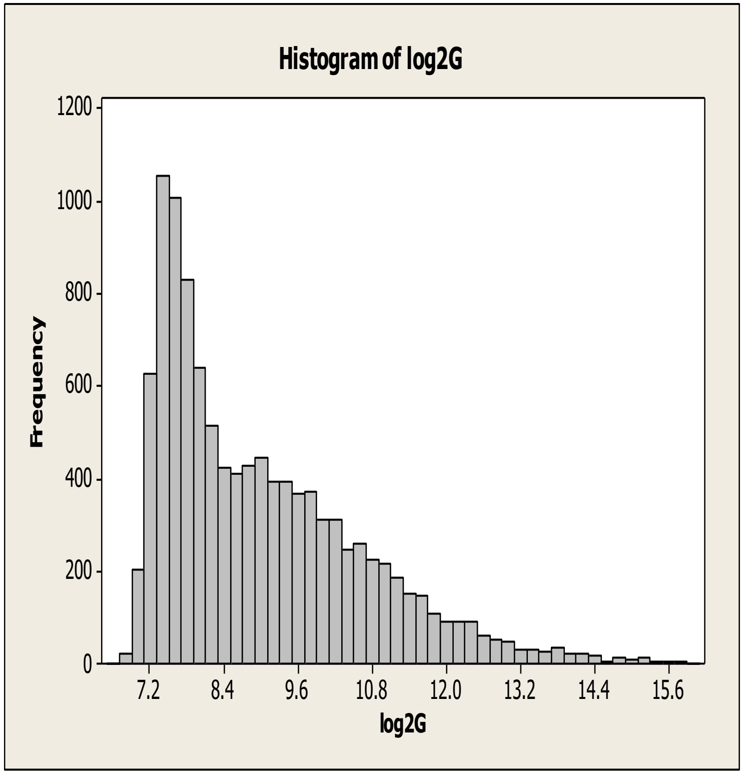
Simple Idea: Take a Log$_2$

- Difference in expression intensity exist on a multiplicative scale, log transformation brings them into the additive scale, where a linear model may apply.

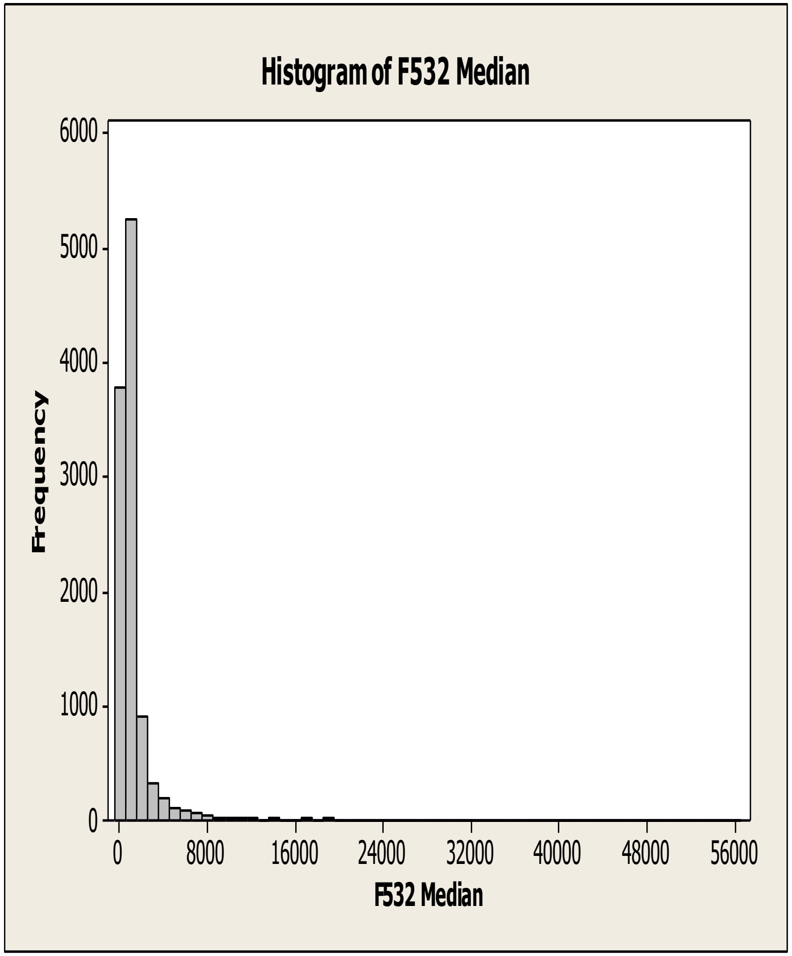
- Ex. 4 fold repression=0.25 (Log$_2$=-2)
- Ex. 4 fold induction=4 (Log$_2$=2)
- Ex. 16 fold induction=16 (Log$_2$= 4)
- Ex. 16 fold ***repression***=0.0625 (Log$_2$=-4)
- Evens out highly skewed distributions
- Makes variation of intensities…independent of absolute magnitude

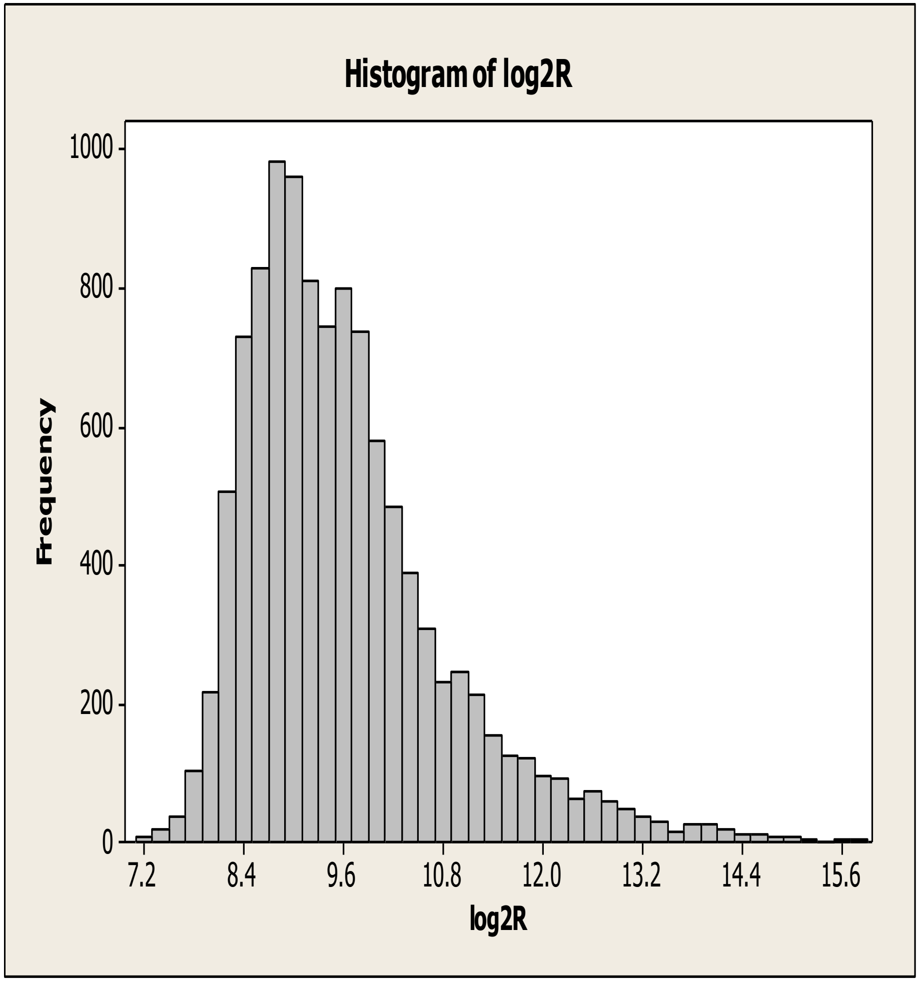
Log Transformation:
Makes the distribution less skewed





Example 2


Histogram of F532 Median
Frequency
6000
5000
4000
3000
2000
1000
0
0
8000
16000
24000
32000
40000
48000
56000
F532 Median


Histogram of log2R
Frequency
1000
800
600
400
200
0
7.2
8.4
9.6
10.8
12.0
13.2
14.4
15.6
log2R

Box-Cox Transformation

- When selecting the optimal transformation for an attribute is that we do not know in advance which transformation will be the best
- The Box-Cox transformation aims to transform a continuous variable into an almost normal distribution

Data Transformation

Box-Cox Transformations

- A good transformation can be achieved by mapping the values using following the set of manipulations:

$$y = \begin{cases} x^{\lambda-1}/\lambda, & \lambda \neq 0 \\ log(x), & \lambda = 0 \end{cases}$$

- All linear, inverse, quadratic and similar transformations are special cases of the Box-Cox transformations.

Data Transformation

Box-Cox Transformations

- Please note that all the values of variable *x* in the previous slide must be positive. If we have negative values in the attribute we must add a parameter *c* to offset such negative values:

$$y = \begin{cases} (x+c)^{\lambda-1}/g\lambda, & \lambda \neq 0 \\ log(x+c)/g, & \lambda = 0 \end{cases}$$

- The parameter *g* is used to scale the resulting values, and it is often considered as the geometric mean of the data

Data Transformation

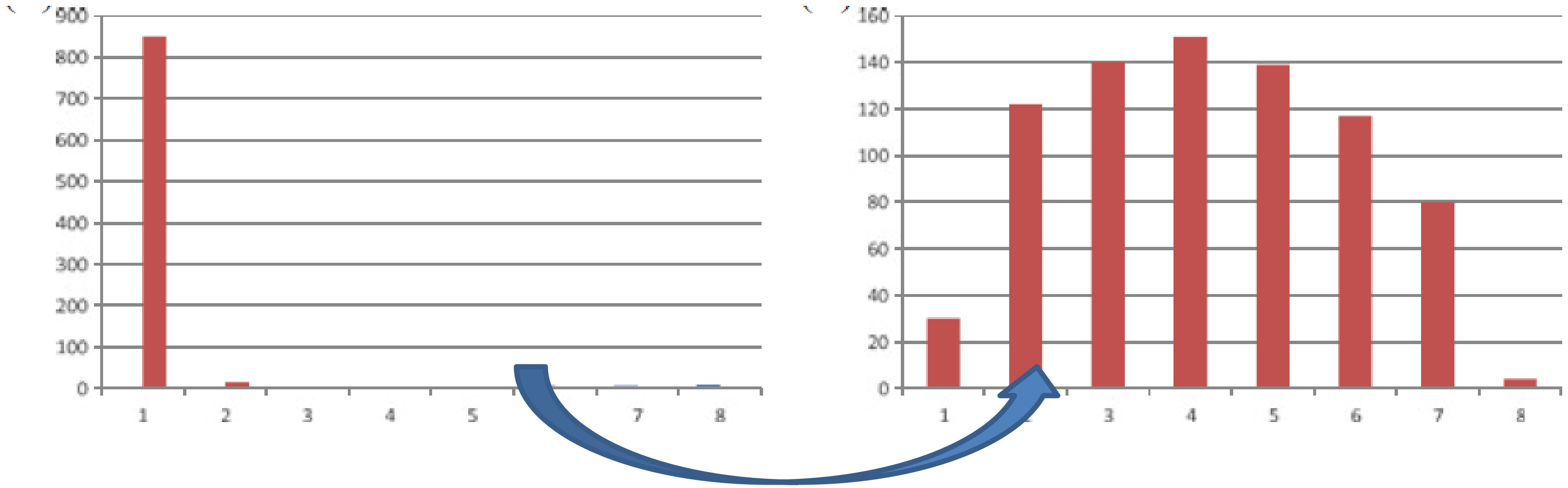
Box-Cox Transformations

- The value of λ is iteratively found by testing different values in the range from -3*.0 to 3.0* in small steps until the resulting attribute is as close as possible to the normal distribution.

Data Transformation

Spreading the Histogram

- Spreading the histogram is a special case of Box-Cox transformations
- As Box-Coxtransforms the data to resemble a normal distribution, the histogram is thus spread as shown here



Data Transformation

Spreading the Histogram

- If you are not interested in converting the distribution to a normal one, but just spreading it, we can use two special cases of Box-Cox transformations
  1. Using the logarithm (with an offset if necessary) can be used to spread the right side of the histogram: $y = log(x)$
  2. If we are interested in spreading the left side of the histogram we can simply use the power transformation $y = x^g$

Data Transformation: Aggregation

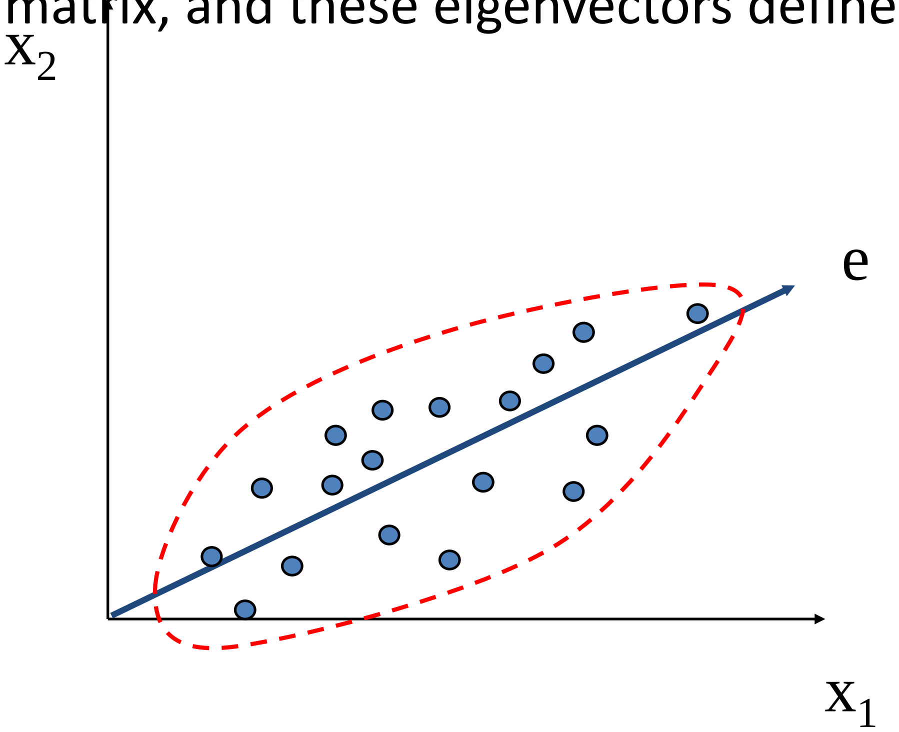
- Combining two or more attributes (or objects) into a single attribute (or object)
- Purpose
  - Data reduction
    - Reduce the number of attributes or objects
  - Change of scale
  - More “stable” data
    - Aggregated data tends to have less variability

Dimensionality Reduction: Principal Component Analysis (PCA)

- Given *N* data vectors from *n*-dimensions, find *k* ≤ *n* principal components (*orthogonal vectors*) that can be best used to represent data
- Steps:
  - Normalize input data: Each attribute falls within the same range
  - Compute *k* orthonormal (unit) vectors, i.e., *principal components*
  - Each input data (vector) is a linear combination of the *k* principal component vectors
  - The principal components are sorted in order of decreasing "significance" or strength
  - Since the components are sorted, the size of the data can be reduced by eliminating the weak components, i.e., those with low variance. (i.e., using the strongest principal components, it is possible to reconstruct a good approximation of the original data
- Works for numeric data only
- Used when the number of dimensions is large

Principal Component Analysis (PCA)

- Find a projection that captures the largest amount of variation in data
- The original data are projected onto a much smaller space, resulting in dimensionality reduction. We find the eigenvectors of the covariance matrix, and these eigenvectors define the new space

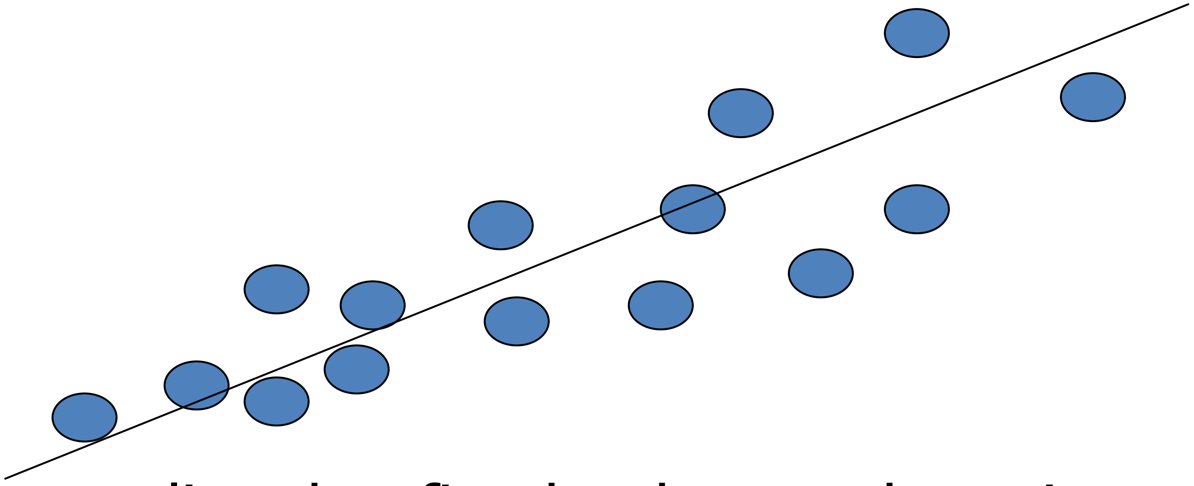


Algebraic Interpretation

- Given m points in a n dimensional space, for large n, how does one project on to a low dimensional space while preserving broad trends in the data and allowing it to be visualized?

Algebraic Interpretation – 1D

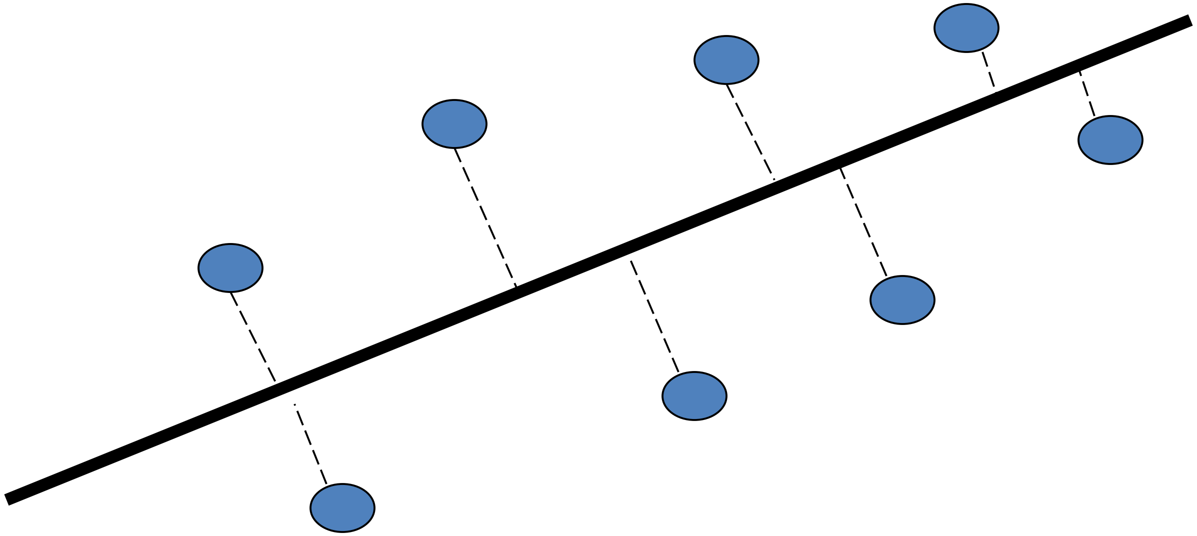
- Given m points in a n dimensional space, for large n, how does one project on to a 1 dimensional space?



- Choose a line that fits the data so the points are spread out well along the line

Algebraic Interpretation – 1D

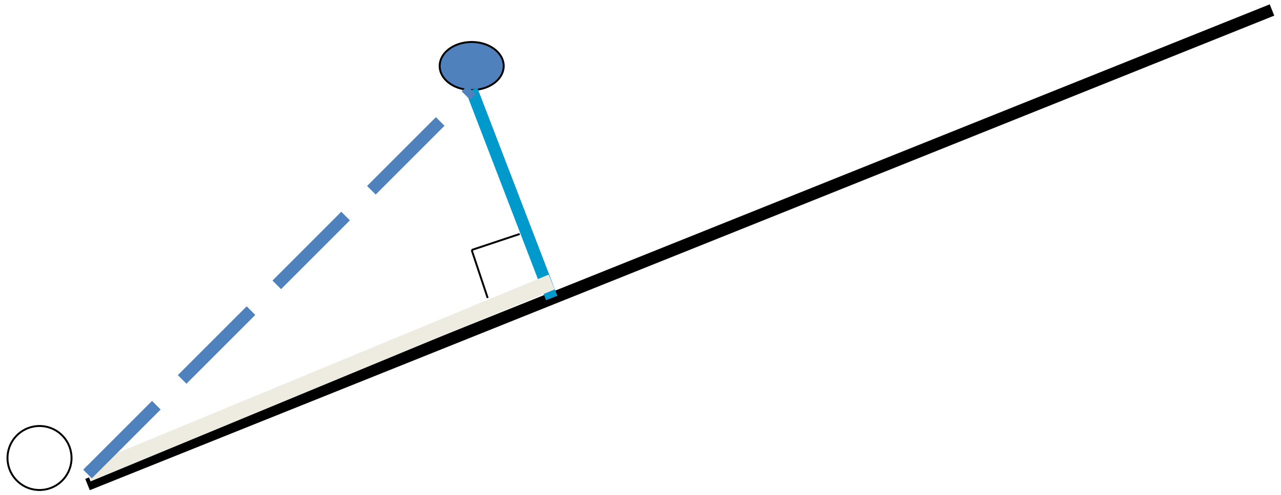
- Formally, minimize sum of squares of distances to the line.

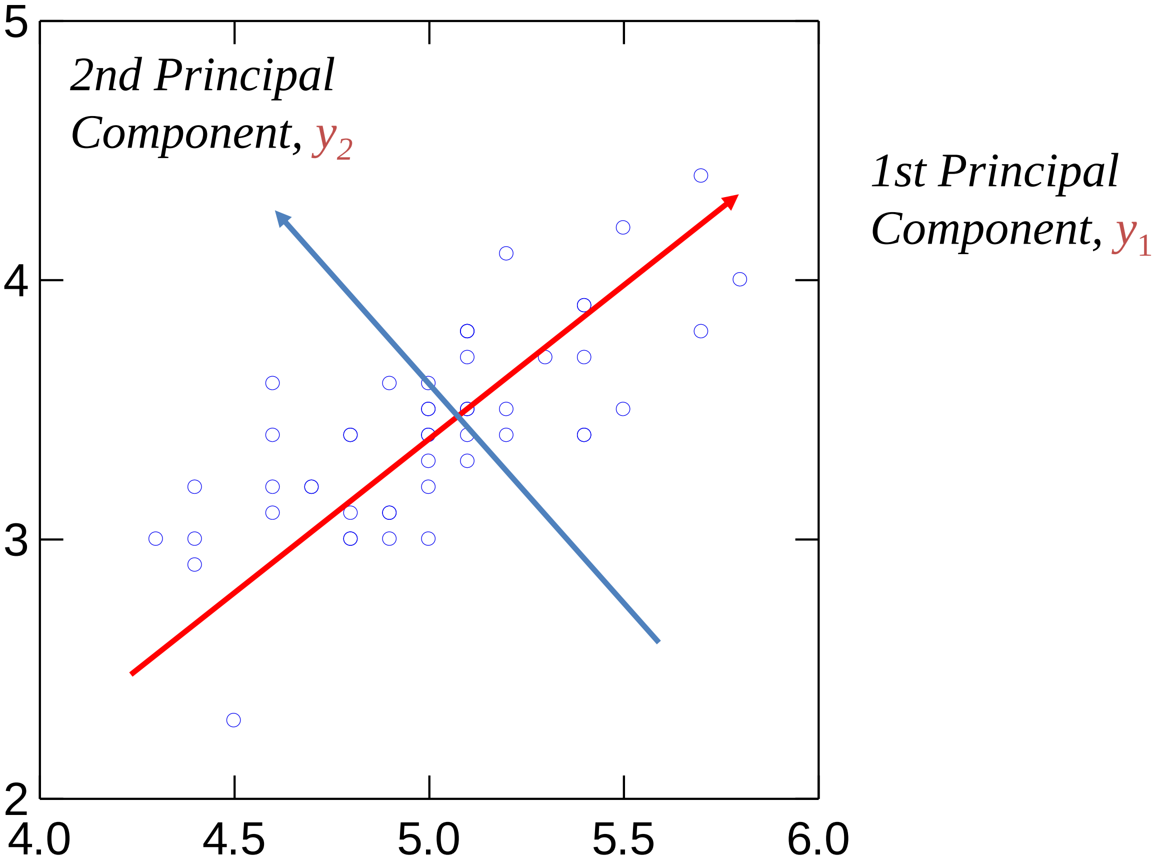


- Why sum of squares? Because it allows fast minimization, assuming the line passes through 0

Algebraic Interpretation – 1D

- Minimizing sum of squares of distances to the line is the same as maximizing the sum of squares of the projections on that line, thanks to Pythagoras.




2nd Principal
Component, y_2
1st Principal
Component, y_1
5
4
3
2
4.0
4.5
5.0
5.5
6.0