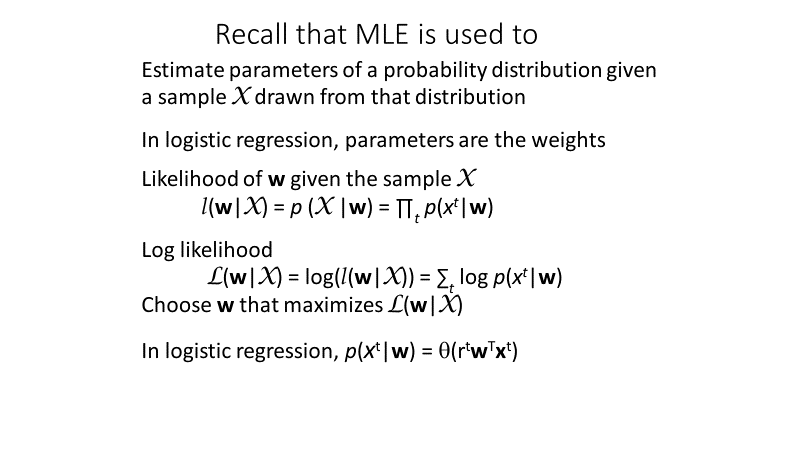
# [Logistic regression](../../Lecture/logistic%20regression.pptx)



**For application of MLE, why is it important to assume that samples used to estimate parameters of the distribution function are independent and identically distributed?**

The I.I.D. assumption allows us to simplify the likelihood function as the product of the individual probability of each example in the dataset:

We take the log of the likelihood, so the product becomes a sum, enabling us to easily calculate derivatives.

# 

# [Extending linear models by features](../../Lecture/extending%20linear%20models%20by%20features.pptx)

**The XOR data set is not linearly separable in attribute space.**

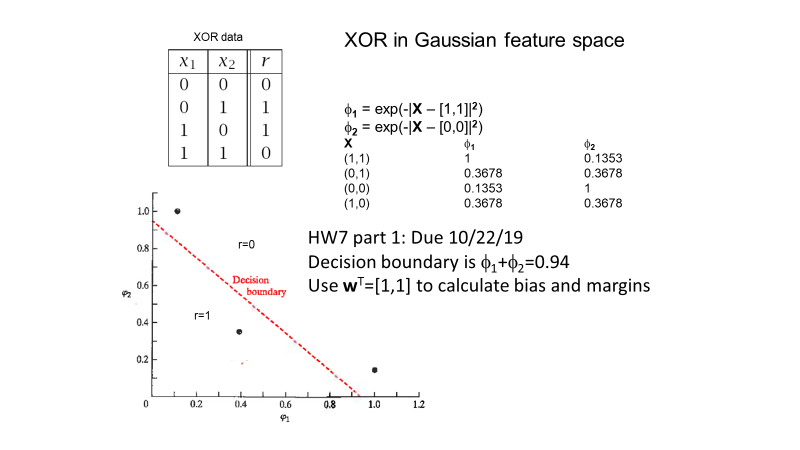
**What is the basic rationale for getting linearly separable features using either a Gaussian transformation or an ANN with 1 hidden layer containing 2 nodes and a bias?**

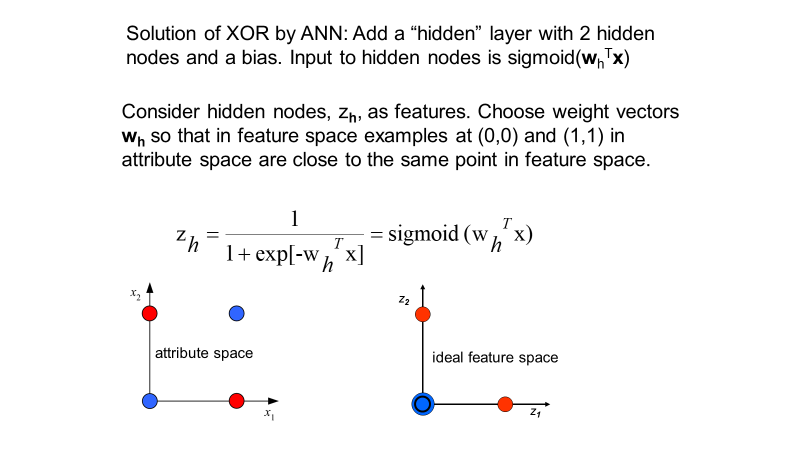
We make a transformation that moves two of the attribute vectors to essentially the same point.

Therefore, we have three points in a two-dimensional feature space; three points in a two-dimensional feature space are always linearly separable.

This is the Gaussian result.

This is the idealized ANN result; we don’t quite realize that, but we get fairly close.





**Explain how a validation set can be used to find the best degree of a polynomial**

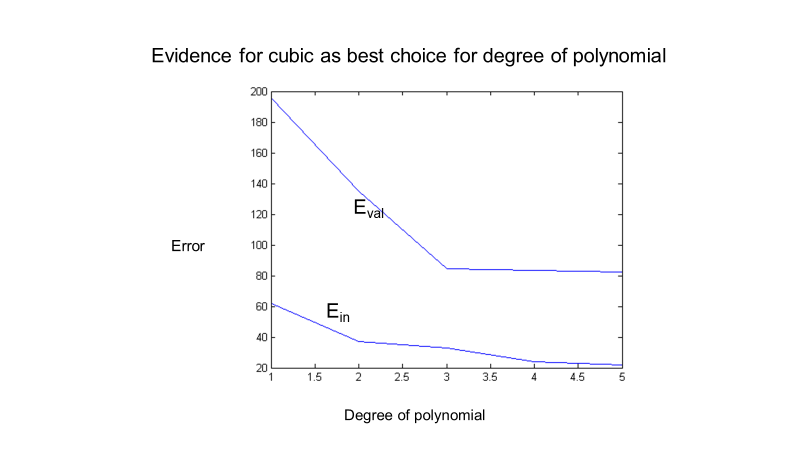
**feature with sacrificing any training data.**

**How is the training set used?**

The training set is used to optimize the weights, the linear combination of the polynomial features, by minimizing an error function (e.g. sum of squared deviations).

**How is the validation set used?**

We don’t determine weights. The training set was used to minimize the in-sample error, now we take those weights and calculate the validation error to get the degree of the polynomial.



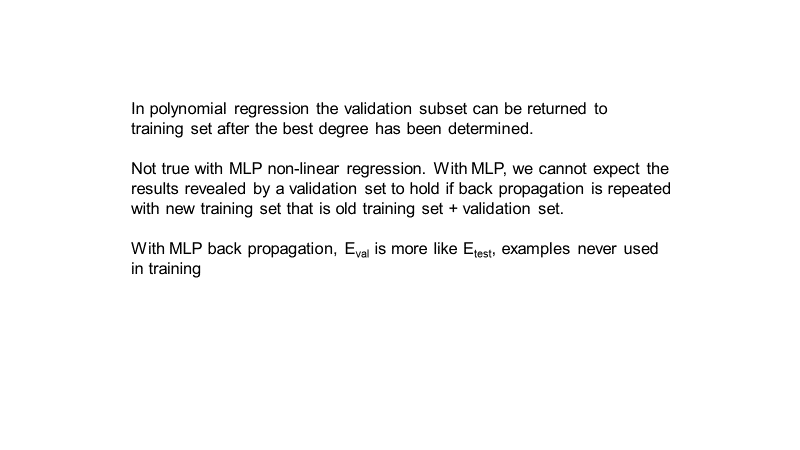
**How do we use a plot of validation error vs degree of polynomial?**

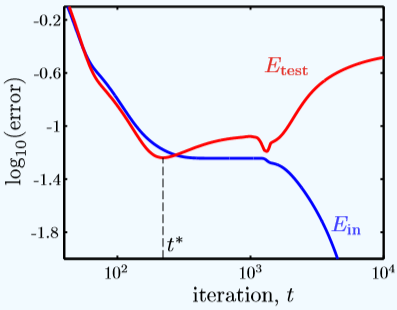
We’re looking for the point where the slope changes radically (elbow), in the above case the best choice is a 3rd degree polynomial. The in-sample error doesn’t help us at all because it keeps going down with the higher degree of polynomial.

**What do we do after be degree of polynomial has been chosen?**

Once we have the degree (the best complexity for the problem at hand), we use the full dataset (combine the training and validation sets) to get the best estimation of the weights.

**How does the validation-set approach have to be modified when applied to ANNs?**





**What’s wrong with these approaches to back propagation:**

**Combine test and training sets and retrain for t\* iterations of back propagation.**

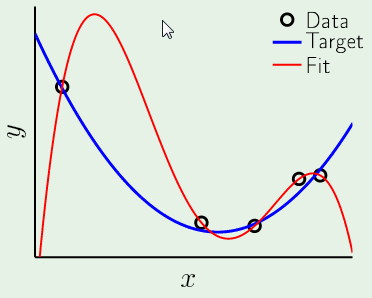
That would not necessarily give us anything like the result we got with just the training set.

**Combine test and training sets and retrain until Ein = Etest(t\*)**

That might not even be possible or desirable. Could cause no training cycles if they’re initially equal and could cause infinite cycles if they never intersect.

The results we got at t\* cycles of back propagation are undoubtable the best result to use.

# Overfitting



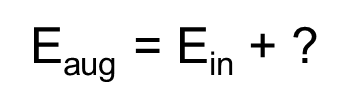
**What’s wrong with this result that has Ein = 0?**

It won’t generalize well because the fit is absolutely nothing like the target. Although we have zero in sample error, we can’t say this 4th order polynomial has a shape in the range of the data that is anything like the target function.

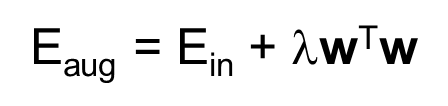
**What should I expect from another set of 5 points in the same range of x?**

Something totally different because the shape of this 4th order polynomial result is just the result of the error that has been assigned to these data points extracted out of the target function. So basically, it’s an expression of the fit to the error in those data points.

# Regularization by weight decay



**What do we add to Ein to get an augmented error function that will penalize selection of large weight vectors?**



A constant that has yet to be determined and multiply it by the square modulus of the weight vector. Obviously, large weight vectors will be penalized by that form.

**With polynomial features, how do I find the best choice of and the best model with**

**weight-decay regularization.**

We have to take the data set, divide it into training samples and validation samples, determine the weights for a given value of , take those weights and see what the error in the validation set is, do that for various values of , plot the validation error and see what value of produces an elbow. Then we combine the training and validation set and get the weights using the optimum value of .

**What happens when the value of gets very large?**

At isn’t any help and too large of a causes the model to forget about the data (all it cares about is . Somewhere between the two is the optimum choice of .



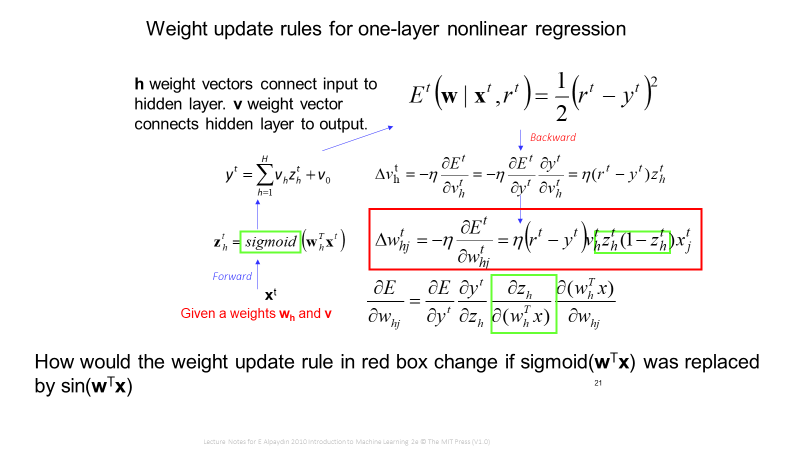
**How do we change weight update rules of back propagation to penalize selection of large weight vectors in back propagations?**



Obviously, we change the weight update rule to not only involve the learning weight and the gradient of the error function with respect to the weight, but also take minus lambda times the weight itself. This term is basically going to always cause to make the weight smaller.

**With hidden layer features, how do I find the best choice of ?**

For neural networks, find the value of lambda that minimizes error in the test set.



The derivative of sigmoid with respect to (it’s argument)

is the derivative of the sigmoid function with respect to it’s argument

The derivative of sin is cos, but we can’t just use that because the weight update rule is written in terms of the value of z

The z is defined as the sin of the dot product, so we can’t just replace that by cos

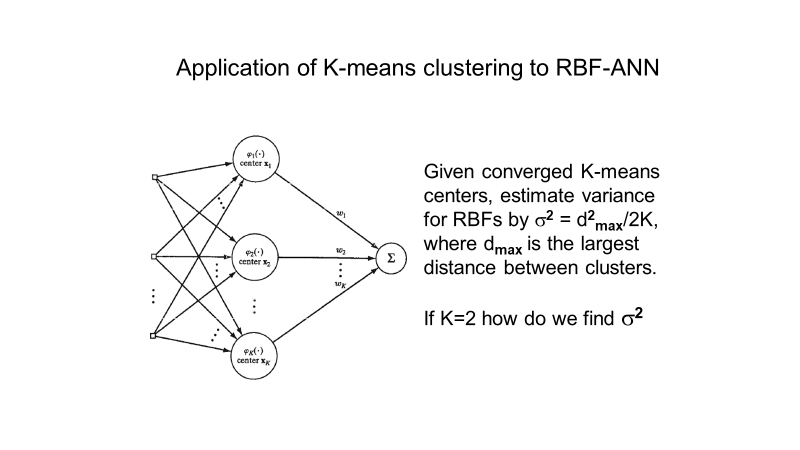
We have to express the cos as a function of the sin

The answer is to calculate this derivative

Whatever nonlinear function I use to calculate z, calculate the derivative and express the result as a function of z.

[https://www.wolframalpha.com](https://www.wolframalpha.com/)

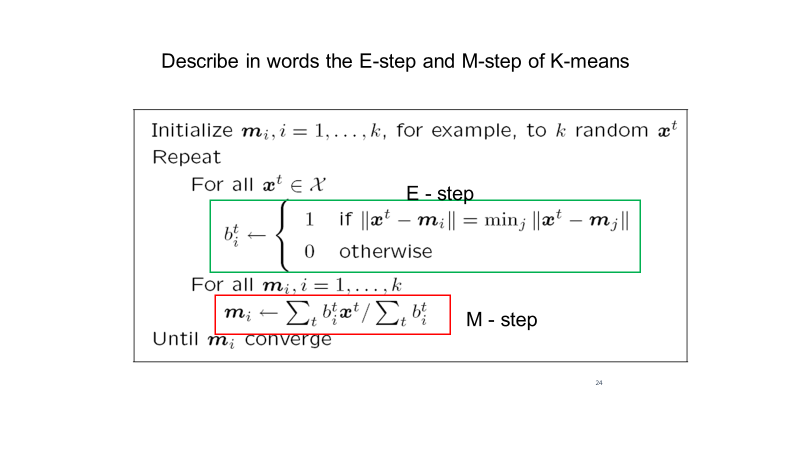
# Radial basis function network



I need to have vectors for the mean values of the two clusters. Then take the difference of the mean values (d) and divide it by 4.

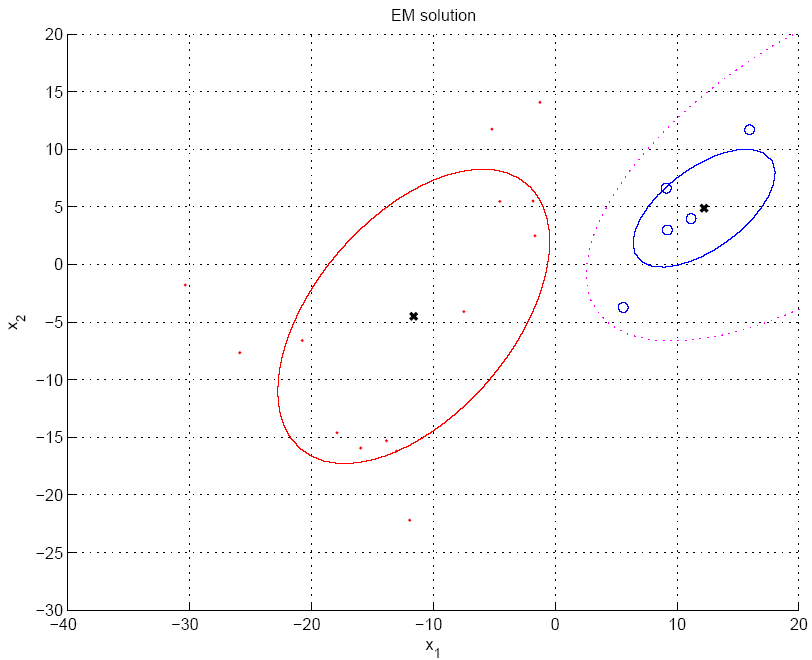
**What if we had 3 clusters?**

I have 3 mean values. That describes where each cluster is in attribute space. I need to find the largest distance between the 3 vectors. So calculate the 3 differences, get the magnitudes, compare and take the max.



An E-step is assigning labels to clusters. If attribute vector t belongs in cluster i, it’s label is 1, and the label for every other cluster is 0. I need to know the means because that’s how I know if attribute t belongs in cluster i, how close it is to the mean of all the clusters, and which is closest.

The M-step uses the labels of the E-step to calculate new means.



**What is the meaning of soft label in Gaussian mixtures**

This label is the probablity that attribute vector t belongs to cluster i. This soft label has a value every cluster, being the probably

Each attribute vector has a soft label for every cluster indicating the probability that it belongs to that cluster.

**What is the advantage of Gaussian mixtures over K-means as a clustering method for RBF ANN?**

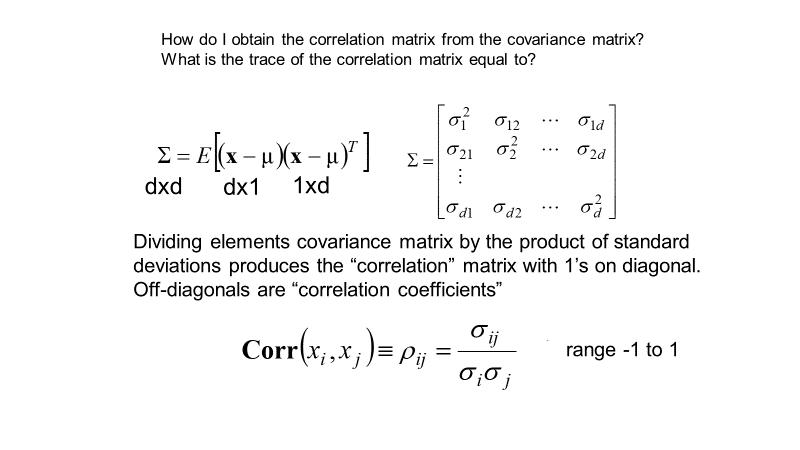
K-means only determines the mean, so you need to use another method to estimate the variance. Gaussian mixtures give both means and variances, both parameters we need for the Gaussian basis functions of the regular basis function ANN.

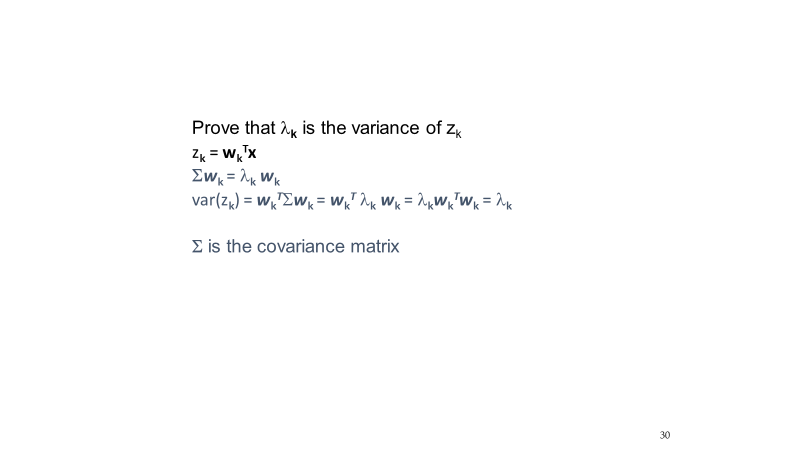
# [Genetic Algorithm](file:///C:\Users\James\Documents\School\FALL%202019\CPTS%20434%20-%20Neural%20Networks\Lecture\Genetic%20Algorithms.ppt)

**Going from fitness to discrete probability and selecting chromosome for modification to create the next generation.**

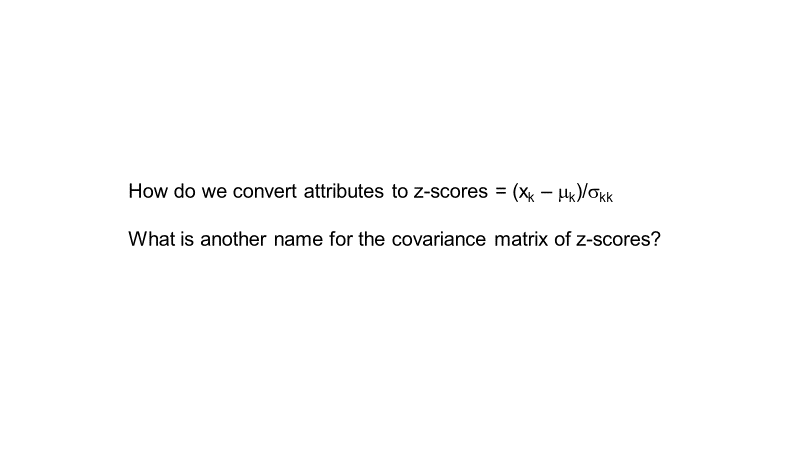
|  |  |
| --- | --- |
|  |  |
|  |  |

# [Principal Component Analysis](../../Lecture/dimensionality%20reduction.ppt)

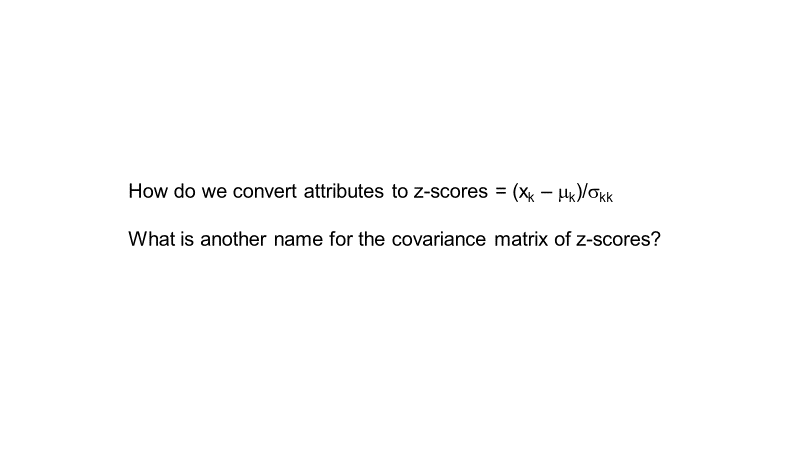




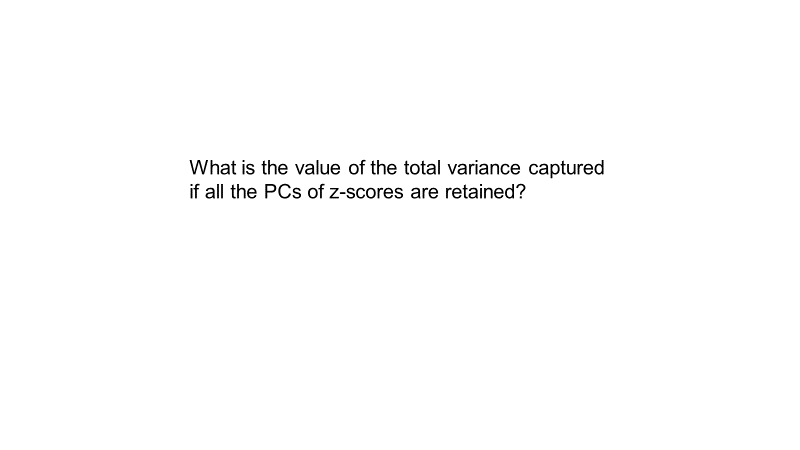
Proving that the variance of the principal component is equal to its eigenvalue.



Estimate the mean and the square of the standard deviation



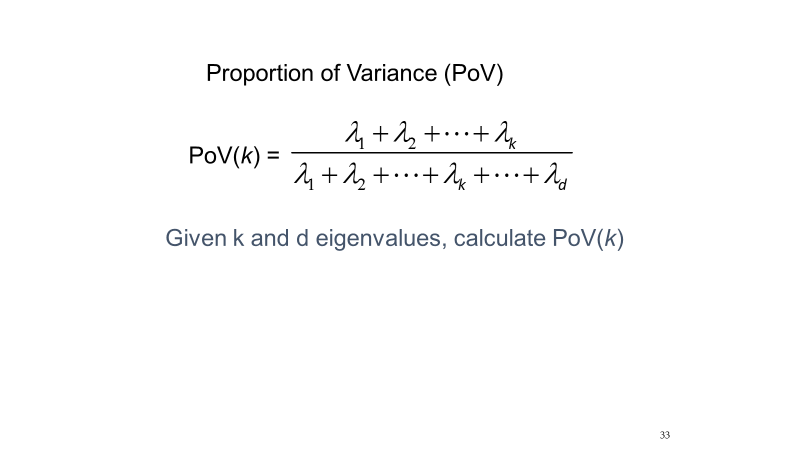
Correlation Matrix

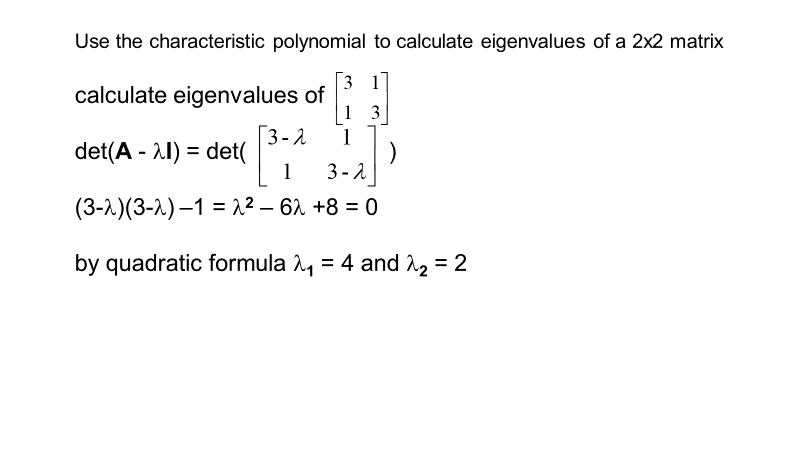


It’s the number of attributes.

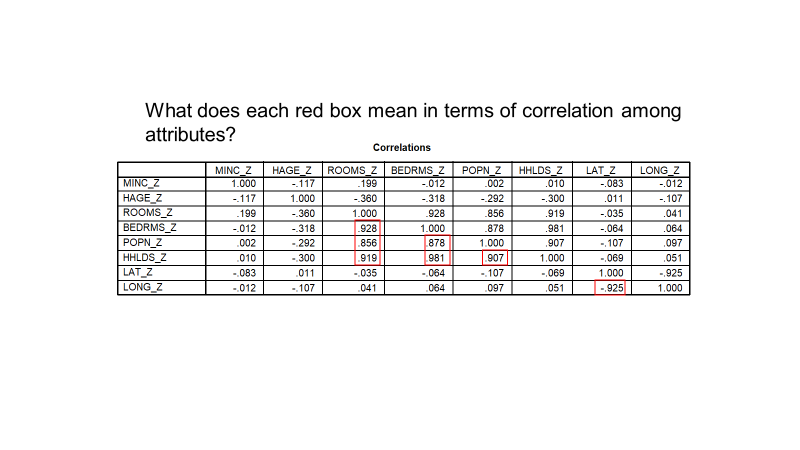
The trace of a matrix is invariant under diagonalization, which means that the sum of the eigenvalues of the correlation matrix will be the same before and after diagonalization.

The diagonal element of the correlation matrix is 1, so if you add them up it’s the number of attributes.





[https://www.wolframalpha.com](https://www.wolframalpha.com/)



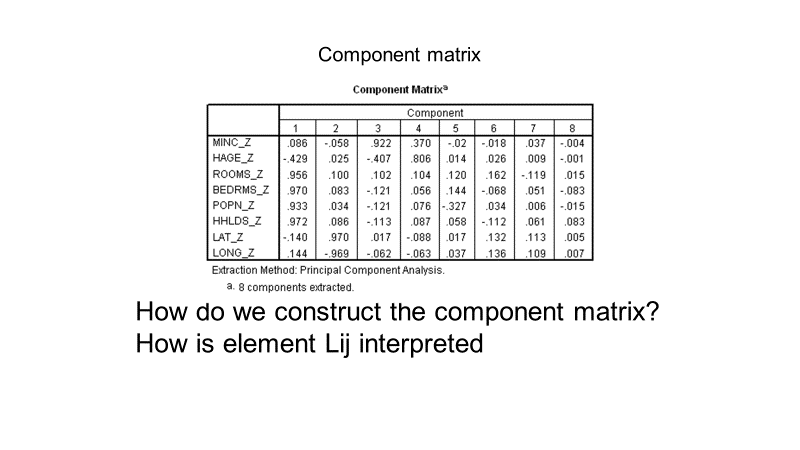
ROOMS is strongly correlated with BEDRMS, POPN, HHLDS

BEDRMS is strongly correlated with POPN, HHLDS

POPN is strongly correlated with HHLDS

LAT is strongly correlated with LONG

This is because the values are close to 1. If close to -1, strong negative correlation.



For every principal component, there is a weight vector that is the solution of the eigenvalue equation. That weight vector tells us how to take linear combinations of attributes in order to get PCs. If I take the weight vector and multiply by the sqrt of the eigenvalue, I get a column of the component matrix.

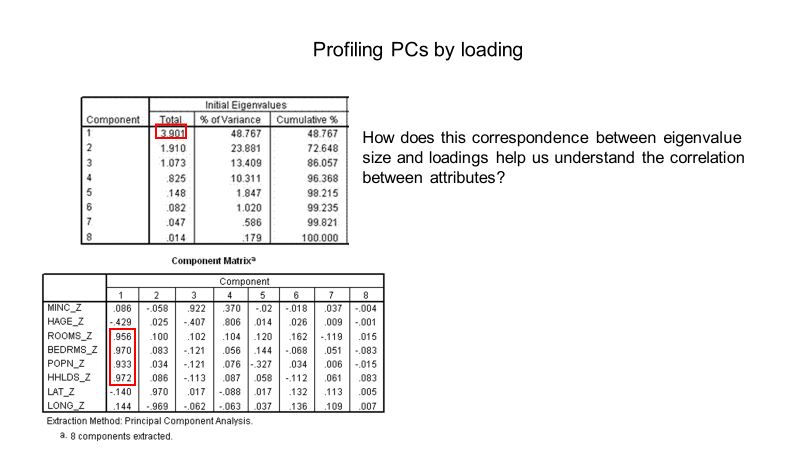
Column 2 of the component matrix is the weight vector solution of the eigenvalue equation for the next smallest eigenvalue and I multiply that weight vector by the sqrt of the eigenvalue and I get this particular column.

The entries in this matrix are called loadings. The rows are z-scores of attributes and the columns are PCs. So a row shows how that attribute relates to each PC.

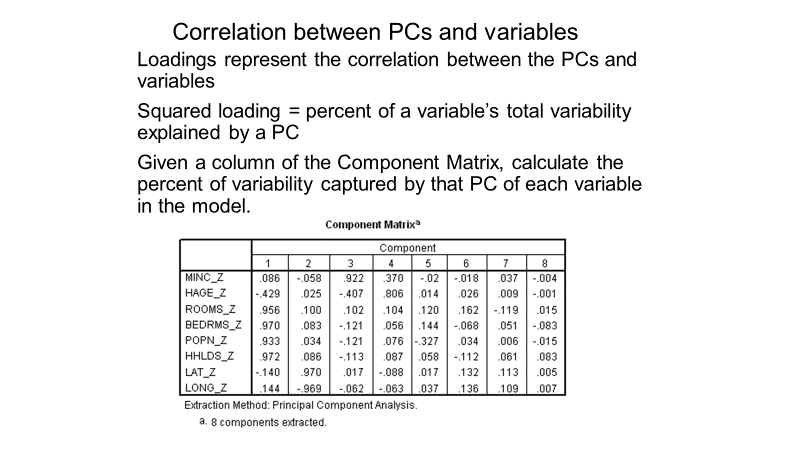
The attribute that loads PC 1 the most is HHLDS because .972 is the max in the column.

MINC loads PC 3 more than any of the others.

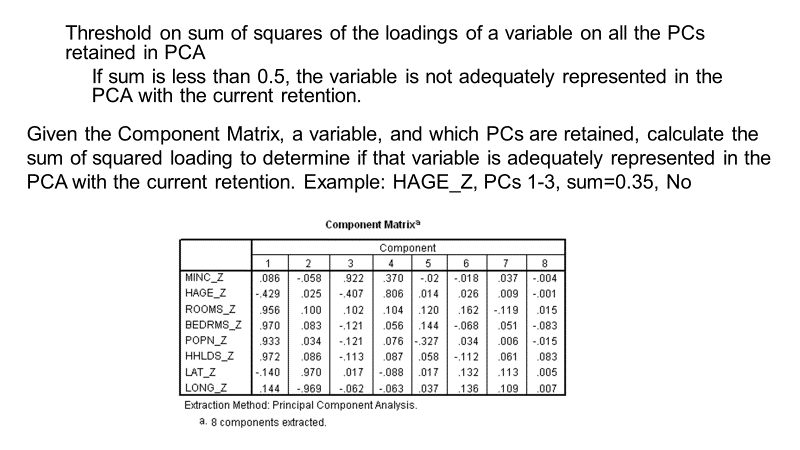
Column 1 shows how much each attribute loads PC 1.



Note that the total eigenvalue is almost equal to 4, meaning that PC 1 is probably the equivalent of 4 attributes. Looking at the Component Matrix, there are 4 significant loadings, so there is 1 factor that contributes to the strong correlation between those 4 variables. That factor is represented by PC 1, in this case block size.



Use <quiz2-helper.xlsx>



Use <quiz2-helper.xlsx>