Lecture 12 (Nov 30, 2020)

No more assignments or quizzes

Focus on your projects only!!!

DO PART I ONLY

k-means

1. do problems described in pdf

PART II (optional no grade)

1. take the best k\* from part 1 and implement your own kmeans with the following three metrics:
2. Euclidean
3. Manhattan (Minkowski p=1)
4. Minkowski (p=1.5)

What distance mtric gives you the most “pure” clusters?

Project Presentations: Dec 7 (Monday)

Final exam: Dec 12-13 (Sat, Sun)

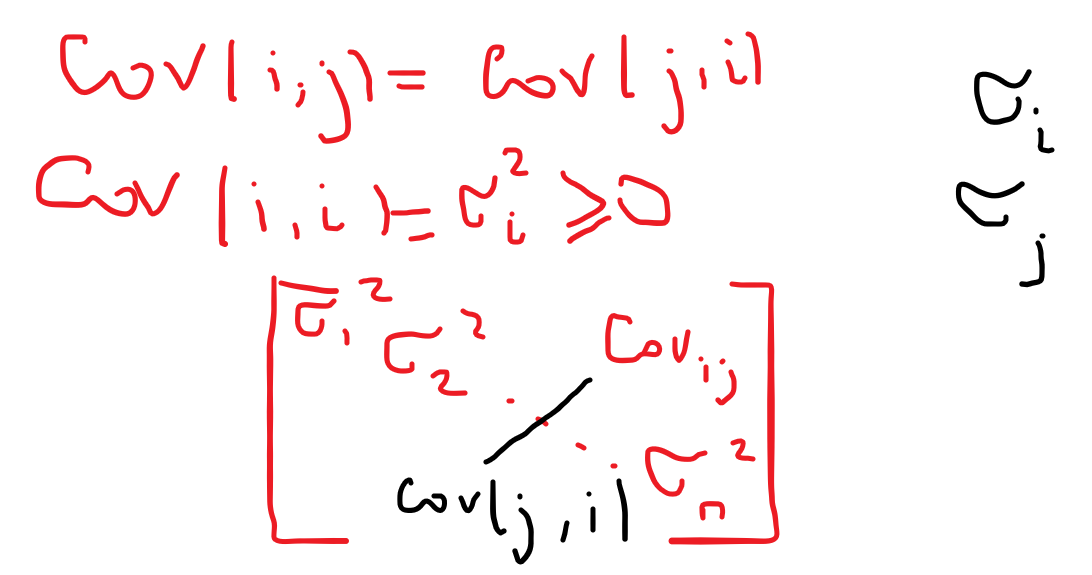
Code, video, description due by Dec 10, 2020

* 1-2 page description how to run
* code
* - video recording about your project

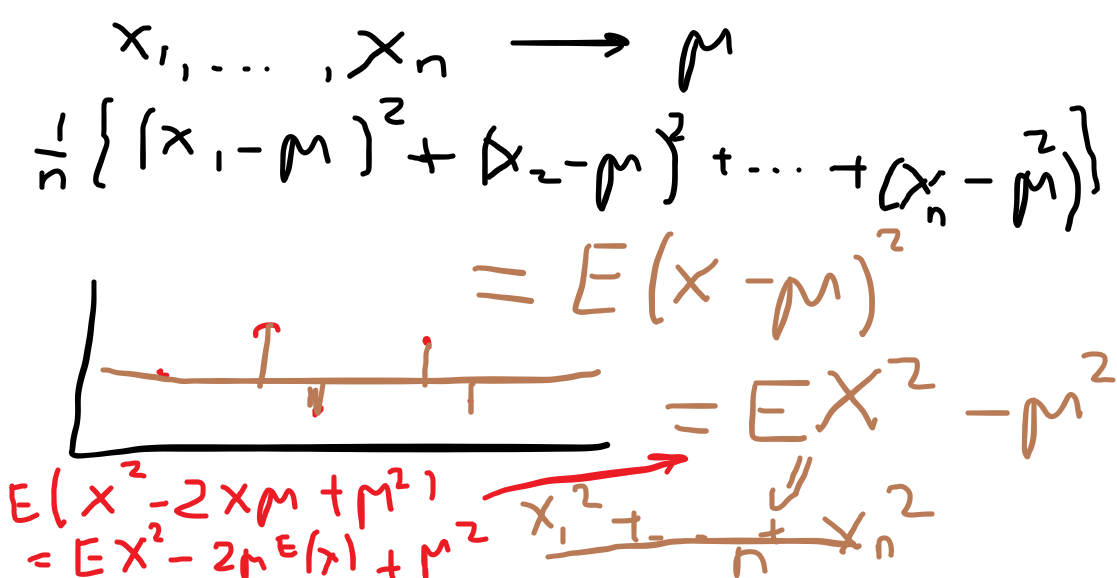
powerpoint presentation on Dec 7

Principal Component Analysis (PCA)

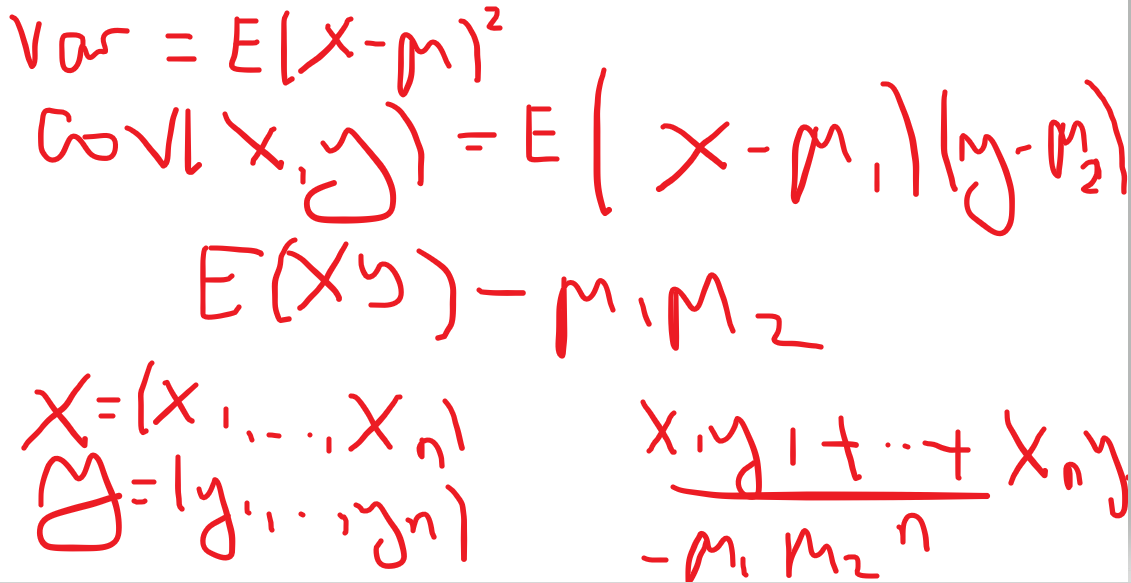
x

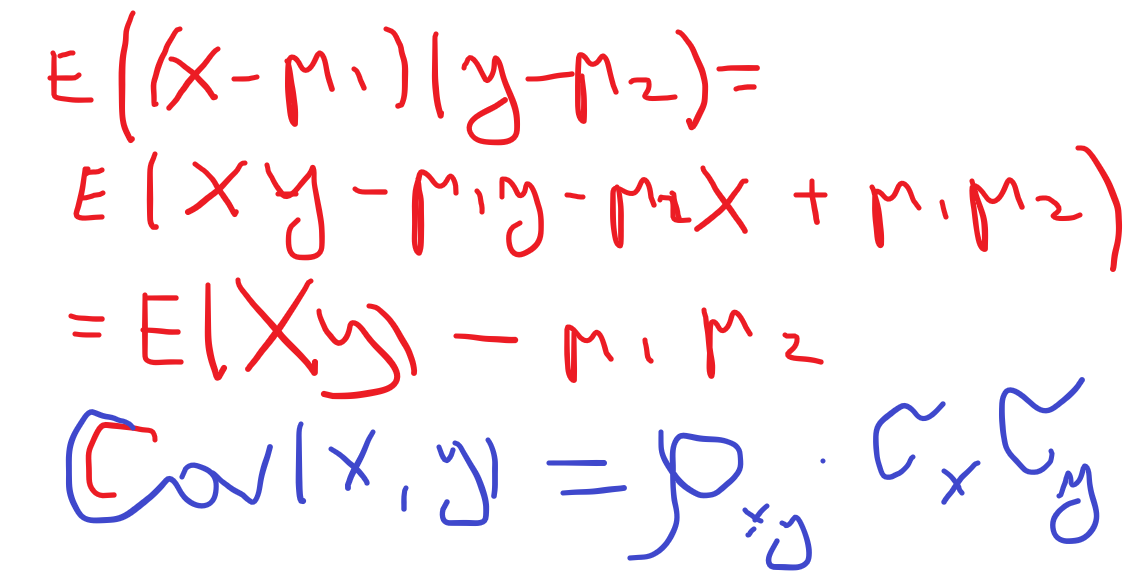


Variance in one dimensional case:



Two variable case





how can we use covariance matrix to reduce dimensionality of our problem

(want to reduce the number of features and still have good accuracy on classification)

How: choose a different coordinate system to represent our features?

How to we choose this? We compute eigenvectors and eigenvalues of the covariance matrix M

lambda x I – M = 0

Most of the variation of your data will be along the first k dimensions (for large n we may find that k << n)

Therefore, we can explain our data by using just k “new features”

Each new feature is a linear combination of the original features

Advantage: simpler description, faster algorithms

REVIEW:

prediction vs. classification

supervised learning: training set and assigned labels

task: define rules that explain our labels

data -------🡪 training and testing

“train” classifier (compute parameters)

How do you describe the performance (or accuracy)

confusion matrix:

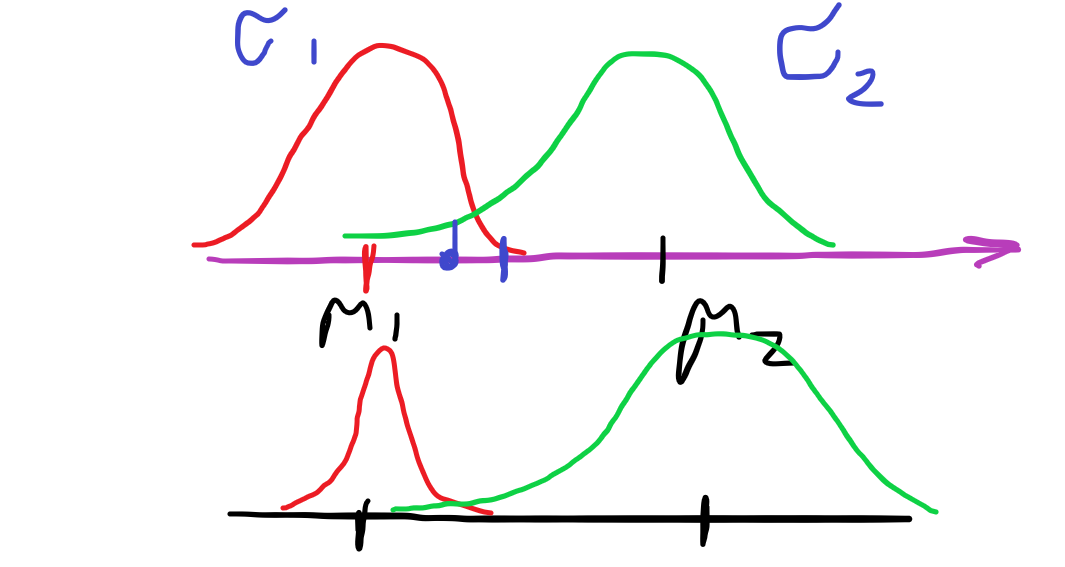
ROC curve (how good is your prediction relative to simple “coin flipping”)

classifiers:

k-NN - compute the best k (larger k does not mean better accuracy)

logistic regression - very widely used and is not very sensitive to errors

linear models



QDA vs. LDA

* data is normally distributed
* sigma\_1 = sigma\_2 (covariance matrices) then separating surface is linear (hyperplane)
* w1x1 + w2x2 + … + wnxn + C = 0
* if sigma\_1 and sigma\_2 are different then the separating surface is quadratic