Pattern Analysis, SS 2017

Dr. Riess is very friendly and the atmosphere of the examination is pretty relaxed, so no need to worry.

1 Intro

At the beginning I had to present the usual big picture. It's important to illustrate the relationship between all the different topics (for example: Parzen-window → Mean shift → We can use this for clustering → Clustering (k-means, k-NN, GMM) → we need to specify number of clusters → Dirichlet process as an alternative and so on...)

2 Clustering

How does k-means work?

1. Random assignment or best guess
2. Assign samples to cluster with nearest cluster mean
3. Recompute means
4. Iterate 2) 3) until convergence

Specifying k isn't so easy in most cases, is there any heuristic we can use?

Compute $J = \sum_{c \in C} \sum_{x_i \in c} \|x_i - \mu_c\|_2^2$ for different k's and search for "knee" in resulting graph.

Can we improve this procedure?

Also compute J on uniformly distributed samples. Search for k where $J_{\text{uniform}} - J_{\text{training set}}$ is biggest. (You can look this heuristic up in *Hastie, Tibshirani, Friedman:The Elements of Statistical Learning, Section 12*)

Ok, now we can go a different way and do soft clustering via GMMs without having to specify the exact number of clusters.

I told him everything he said about Dirichlet Processes (Distribution over distributions, CRP as a way to constructivly "sample" from a Dirichlet distribution, why Chinese Restaurante, how does the algorithm look like, how to compute the affinities, ...).

Now we have those affinities. How do we compute which cluster we should assign a sample to?

Normalize sum of affinities to one → We now have a discrete pdf and are able to randomly draw a cluster assignment from it.

How do we draw samples from a one dimensional pdf?

Compute cdf, draw uniformly distributed number v between 0 and 1. Our sample x is the one where $cdf(x) = v$.

Imagine you'd like to segment a picture via clustering. How would you do that?

Transfrom each pixel into feature vector (x, y, r, g, b), apply clustering algorithm. Imporant: scale dimensions.

When you use Mean Shift for clustering, do you have to specify the number of cluster?

Not directly, but the kernel size influences the number of maxima we get. I then had to illustrate one a 1d sample distributing which local maxima mean shift would detect with different window sizes.

Transition to new topic: *Imagine your working for a company and your boss says, well that's nice and all... But all these clustering algorithms are very 90s. I want you to do image segmentation via MRFs. How could we do that?*

3 MRF

I first explained how a typical MRF would look like (grid structure), mentioned the Hammersly-Clifford theorem and wrote out the general pdf factorization of a Gibbs random field ($p(x) = \frac{1}{Z}e^{-\sum_c H_c(x_c)}$). After that I told him how the binary and pairwise potentials would look like if we would do image denoising.

Now, how would the potentials/energy functions look like if we would do segmentation?

This was all a bit fuzzy. We would assume smoothness of the values of the random variables (0 means pixel is part of background, 1 means pixel is part of foreground), so the pairwise potentials would be something like the difference between two neighbooring random variables. For the unary potantials he said that this is a bit harder and its ok that I don't know how to specify these.

Imagine for the moment we would have fully specified our energy functions, how can we minimize our overall energy function?

I began with the submodularity condition, every energy function fullfilling this condition is graph representable, this means we can minimize the function by finding the minimum s-t-cut.

How would the subgraph for a unary potential look like?

Showed him the same thing he showed us in the lecture.

Now thats cool and all, but why bother constructing this graph and minimizing the function via minimum cut?

There exist algorithms which run in polynomial runtime.

I made some lecture notes, which you can find here (there may be some errors and some lectures are missing):

`https://github.com/elbundy/SS17/tree/master/PA`