# Questions on image classification

For meeting with Laurens van de Maaten, 6 Jan 2014

## What we did so far

* Followed the Coursera course on Probabilistic Graphical Models from Daphne Koller.
* Set-up a multi-class CRF on coastal infrared time-variance images, but simplified it to single-class due to poor performance.
* Started manual classification of a set of Argus images from the Dutch coast in 6 classes: sky, sea, beach, dune, object, label

## Theory

* Bayesian network analogy
  + Are Bayesian networks and CRF’s somehow related? If so, what is the key difference? Is it the use of pairwise potentials, i.e. reasoning from the edges instead of the nodes?  
    *Generative/discriminative. So CRF only solves for conditional probability.*
  + Is it correct that feature values are continuous in CRFs, while discrete in Bayesian networks?  
    *Yes.*
  + How is Bayes Theorem involved in CRF inference?  
    Bayes Theorem is used in clustering dependent factors in a Bayes model. In our line of thought we explain the purpose of model training from a mathematical point of view also using Bayes Theorem. However, we are unsure if this is at all applicable.  
    *Discriminative/generative. Only solving for conditional probability.*
* Relation to other algorithms
  + How are clustering algorithms based on regression, e.g. K-means, related to CRF classification methods? *CRF without pairwise potentials is equal to logistic regression and similar to K-means.*
  + CRF inference is explained to us based on belief propagation. We understand that more advanced algorithms are available for inference (e.g. SVM, boosting). How do these algorithms relate to belief propagation? And how does training a CRF differ between these inference algorithms? *Logistic regression with pairwise potentials is a CRF, while a SVM with pairwise potentials is a SSVM. Both can be solved with belief propagation algorithms. The main difference between logistic regression and SVM is the loss function, which is continuous in case of logistic regression and has a kink in case of a SVM. Another algorithm that can be used is Monte Carlo. Logistic regression has the advantage of being related to probabilistics and hence provides insight in the uncertainty of the answer.*
* Can classification clusters be splitted in the feature space?

## Model structure

* We use the pyStruct python package for model construction, training and inference. Do you have any experience with this package? If so, are there better alternatives?

*Not really.*

## Model training

* How would you estimate our chances of success in developing automated methods for classification of Argus images for purposes like intertidal beach mapper, people counter, vegetation coverage?
* We aim at multi-step classification methods. Is this a good idea? Do you have experience with such methods? We start with a course classification in sky, sea, beach, dune, object and label. Subsequently we classify the beach area in, for example, intertidal area and people.  
  *Classes should be specific. Classification can be hierarchical where superclasses are also influenced by subclasses, e.g. a superpixel is more likely to be sea if it is likely to be a wave.*
* Do rules of thumb exist for manual classification? How to choose the number of classes, number of photos and the number of subsequent steps in classification?  
  *Classes should be specific, hundreds of photos are likely needed.*
* What are typical computational efforts we should expect when we want to train a CRF with hundreds of superpixels and a handful of classes based on hundreds of manually classified images? *Weeks not likely, but considerable time needed. Prepare Megatron.*

## Features

* What leads to a stronger training: consistency in location features or consistency in intensity features?  
  E.g.: should not-connected runnels in the intertidal beach be classified as beach or sea?
* Preliminary classification results show that the class “objects” matches regularly on non-object superpixels. This is probably due to the large variety of objects that fall into this class. What would be your advice? Keep on training or classify the objects in a smarter way, e.g. beach vs. dune objects, red vs. blue, etc? *See hierarchical classification and specific classes.*
* Do you have any experience with the relative location prior (Gould *et al.*, 2008)?
* Is it wise to include features like location of the camera, position of the sun, weather type, etc?  
  *Only if it doesn’t work. Last resort.*
* We have a set of 62 features obtained from literature. Many of those are single valued features, but some provide matrix values for a single superpixel, e.g. texture features that provide autocorrelation data for different offsets and angles. Is it appropriate to train a model using each item from such matrix as single valued feature or should we keep the dependency somehow? *Each item is a feature. 62 features is not much.*

## Segmentation

* We use the SLIC segmentation algorithm. Do you have experience with this algorithm? If so, are there better alternatives?
* The SLIC algorithm sometimes returns superpixels of zero size. This results in a non-existent superpixel and hence no regular grid. How would you cope with this non-regularity of the superpixel grid in your CRF? Either use a non-regular gridded CRF or use a dummy class?
* Is it feasible to use different resolution images and/or different number or size of segments within a single training set?

## Answered questions

* What is a useful mathematical description of CRF construction and inference? In case of Bayesian networks we can use minimize the scope of the factors based on dependencies. Is a similar reasoning applicable for CRFs?  
  *Product of edge factors and a partition function.*
* Is it true that many dependent features can be used in a CRF without decrease of model performance? If so, why isn’t this the case for naïve Bayes?  
  *Yes. Every feature has its own partition function and is therefore normalized individually. This takes away the dependence problem.*
* A pairwise potential is expressed as an energy function, for example a negative exponent. A larger difference thereby results in a larger penalty. In case of classes (strings), how does this work? What are typical energy functions?  
  *This is not true for CRF’s. Comes from PyMC package.*
* Are the edges between superpixels and their features directed or undirected? See also the question on Bayes Theorem.  
  *Undirected, Markov network.*
* Can pairwise potentials also be used to stimulate adjacent superpixels to be classified as common bordering classes, like beach to sea and beach to dune? If so, how does it work?  
  *Yes, this is done through the edge factors (local happiness).*

## New questions

* How does logistic regression / crf assure that multiple, interdependent features do not bias the results?
  + We understand binary classification based on a single feature. How to extend this idea to muiltiple features?
  + And to multiple classes?
  + How does sigmoid space (y = 0.5 at x = 0) relate to feature space? Is normalization already taken care of?
  + Do clusters have to be continuous, or can they consist of multiple blobs?
* What to do with feature selection?
  + Is it useful to remove non-active features?
  + Do you have suggestions for new, state-of-the-art features?
  + Are you familiar with the relative location prior (Gould, 2008)?
  + Is it useful to employ 6 or 9 channels instead of 3 (LAB/HSV)?
* What to do with choice of classes?
  + Classes used for annotation leave the option for grouping hierarchic classes open. We expect better performance with grouped classes. What do you think?
  + Biggest errors reside around the beach. How can we improve on that?