Computer Vision with Dirichlet Processes

Tom S. F. Haines
thaines@gmail.com

3rd December 2012

Roadmap

1. Dirichlet Processes (mini tutorial)
2. Background Subtraction
3. Delta-Dual Hierarchical Dirichlet Processes
4. Active Learning
5. Last Words

Note that all code can be obtained from *thaines.com*

What is a Dirichlet Process?

$$G \sim \mathsf{DP}(\alpha, G_0), \quad G \in A, \quad \alpha \in \mathbb{R}, \alpha > 0$$

G_0 is a probability distribution (measure) defined on the range A. DP, a Dirichlet process, then satisfies the property

$$[G(a_1), \ldots, G(a_n)]^T \sim \mathsf{Dir}(\alpha G_0(a_1), \ldots, \alpha G_0(a_n))$$

for any finite partition of A, $\bigcup_{i=1}^{n} a_i = A$, where Dir is the Dirichlet distribution ...

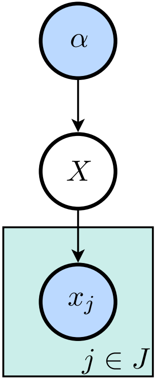
...but this is not very intuitive. Alternatives:

- A generalisation of the Dirichlet distribution.
- The stick breaking construction.
- The Chinese restaurant process.

Dirichlet Distribution

$$x \sim \mathsf{Mult}(X), \qquad X \sim \mathsf{Dir}(a), \qquad x \in H$$

a	$=$	Length n parameter vector; $a \in \mathbb{R}^n, a_i > 0$.
Dir	$=$	Dirichlet distribution; $P(X\|a) \propto \prod_{i=1}^{n} X_i^{a_i - 1}$.
X	$=$	Length n parameter vector; $X \in \mathbb{R}^n, X_i > 0$, $\sum X_i = 1$ (On a $(n-1)b$-simplex).
Mult	$=$	Multinomial distribution (Categorical if $\sum x_i = 1$.); $P(x\|X) \propto \prod_{i=1}^{n} X_i^{x_i}$.
x	$=$	Counts of how many of each entry have been drawn; $x \in \mathbb{N}^n$.
H	$=$	*Meaning* of the entries $i \in \{1, \ldots, n\}$, e.g. days of the week $(n = 7)$.



Distribution to Process

$$x \sim \mathsf{Mult}(X), \quad X \sim \mathsf{Dir}(a), \quad x \in H$$

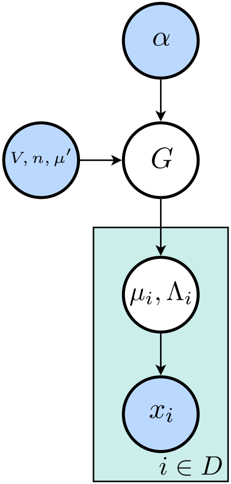
- Set $a \in \mathbb{R}^n = [\frac{\alpha}{n}, \ldots, \frac{\alpha}{n}]^T$, where $\alpha \in \mathbb{R}, \alpha > 0$.
- As $n \rightarrow \inf$ we get the Dirichlet Process ...
- ...mathematically. But there are conceptual differences.

Differences

$$x \sim M(G), \quad G \sim D(\alpha, G_0), \quad x \in H$$

Finite Case	Infinite Case
H = Set of arbitrary atoms, of size n.	H = Range of the base measure, G_0
G_0 = Not used.	G_0 = Base measure, a probability distribution over the atoms.
$\alpha \in \mathbb{R}^n$ = Parameter for the Dirichlet distribution.	$\alpha \in \mathbb{R}$ = The concentration parameter.
D = Dirichlet distribution.	D = Dirichlet process.
G = Finite vector of length n, sum of all entries is 1.	G = A probability distribution that can be interpreted as an infinite length vector.
M = Multinomial distribution.	$M = G$.

Nonparametric Bayesian Mixture Model



$$G \sim \mathrm{DP}(\alpha, P(\mu, \Lambda))$$
$$\Lambda \sim \mathcal{W}(V, n) \quad (\mathcal{W} = \text{Wishart distribution})$$
$$\mu \sim \mathcal{N}(\mu', (n\Lambda)^{-1}) \quad (\mathcal{N} = \text{Gaussian distribution.})$$

$$(\mu_i, \Lambda_i) \sim G$$
$$x_i \sim \mathcal{N}(\mu_i, \Lambda_i^{-1})$$

- This is a *DP Gaussian Mixture model*.
- An infinite number of components means it will assign the probability mass to the components it needs, and set the rest to (almost) zero.
- It *learns* the right number of components!
- (Often a prior (Gamma) would be put on α)

Stick Breaking Construction

- A constructive definition of a DP - probably the most straightforward.
- Typically used directly when employing variational methods.
- Makes explicit the following properties of G:
  - It is *discrete*, even if the base measure is not (The probability of drawing the same entity twice is not zero.).
  - An infinite number of different entities can be drawn (Assuming the base measure is not finite.).

Stick Breaking Construction

Remaining Stick$\rightarrow$ 

$l_0 = 1$

Base Measure$\rightarrow$ 

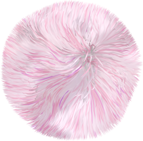
Stick Breaking Construction

Remaining Stick$\rightarrow$ 

$l_1 = v_1$

$v_1 \sim \text{beta}(1, \alpha)$

$\beta_1 = 1 - v_1$



Base Measure$\rightarrow$ 

Stick Breaking Construction

Remaining Stick$\rightarrow$ 

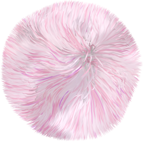
$l_2 = v_1 v_2$

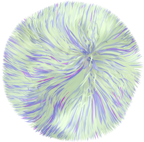
$v_1 \sim \text{beta}(1, \alpha)$

$\beta_1 = 1 - v_1$

$v_2 \sim \text{beta}(1, \alpha)$

$\beta_2 = v_1(1 - v_2)$





Base Measure$\rightarrow$ 

Stick Breaking Construction

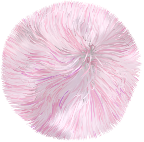
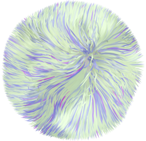
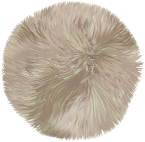
Remaining Stick→ 

$$l_3 = v_1 v_2 v_3$$

$v_1 \sim \text{beta}(1, \alpha)$ $v_2 \sim \text{beta}(1, \alpha)$ $v_3 \sim \text{beta}(1, \alpha)$

$\beta_1 = 1 - v_1$ $\beta_2 = v_1(1 - v_2)$ $\beta_3 = v_1 v_2 (1 - v_3)$

Base Measure→ 

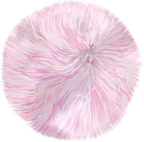
Stick Breaking Construction

Remaining Stick$\rightarrow$ 

$l_n = \prod_{i=1}^{n} v_i$

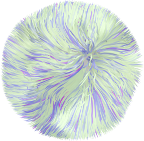
$v_1 \sim \text{beta}(1, \alpha)$

$\beta_1 = 1 - v_1$



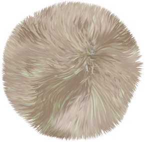
$v_2 \sim \text{beta}(1, \alpha)$

$\beta_2 = v_1(1 - v_2)$



$v_3 \sim \text{beta}(1, \alpha)$

$\beta_3 = v_1 v_2 (1 - v_3)$



. . .

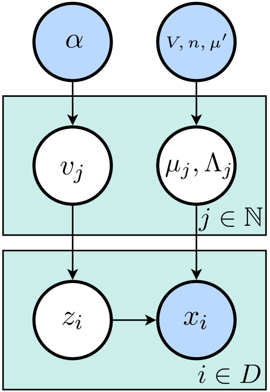
$v_n \sim \text{beta}(1, \alpha)$

$\beta_n = (1 - v_n) \prod_{i=1}^{n-1} v_i$

. . .

Base Measure$\rightarrow$ 

Stick Breaking Mixture Model



$v_j \sim \text{beta}(1, \alpha)$
$\Lambda_j \sim \mathcal{W}(V, n)$ ($\mathcal{W}$ = Wishart distribution)
$\mu_j \sim \mathcal{N}(\mu', (n\Lambda)^{-1})$ ($\mathcal{N}$ = Gaussian distribution.)

$P(z_i = n) = (1 - v_n) \prod_{k=0}^{n-1} v_k$
$x_i \sim \mathcal{N}(\mu_{z_i}, \Lambda_{z_i}^{-1})$

- We have replaced G with something we can almost compute.
- You cap the number of sticks to make it computable.
- Using an indicator vector for z this can be implemented using the mean field variational approach.

Chinese Restaurant Process

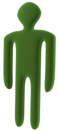
- Closely related to the Blackwell-MacQueen urn scheme.
- It integrates out G:

  If $x_i \sim G, \quad G \sim DP(\alpha, G_0)$ then it calculates $P(x_i | x_1, \ldots, x_{i-1}, \alpha, G_0)$.
- Draws from it are exchangeable - the order of the x_i is irrelevant.

Chinese Restaurant Process

$\frac{\alpha}{\alpha}$

- Customer enters the restaurant, has to choose where to sit.



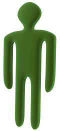
Chinese Restaurant Process

- An infinite number of tables are actually available, but as empty tables are equivalent the choice is meaningless.
- When sitting at an empty table a draw from the base measure (menu) is made - all customers at that table are then associated with that draw.

Chinese Restaurant Process

$\frac{\alpha}{\alpha+1}$ $\frac{1}{\alpha+1}$

- Tables are weighted by the number of customers sitting at them.

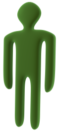


Chinese Restaurant Process

$$\frac{\alpha}{\alpha+2}$$

$$\frac{1}{\alpha+2}$$

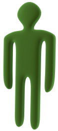
$$\frac{1}{\alpha+2}$$



Chinese Restaurant Process

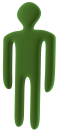
$\frac{\alpha}{\alpha+3}$ $\frac{2}{\alpha+3}$ $\frac{1}{\alpha+3}$

- Two people have sat at one of the tables - the same value has been drawn from the distribution twice.
- Consequentially, a continuous base distribution has been converted into a discrete distribution.



Chinese Restaurant Process

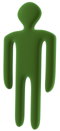
$\frac{\alpha}{\alpha+4}$ $\frac{3}{\alpha+4}$ $\frac{1}{\alpha+4}$



Chinese Restaurant Process

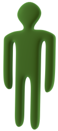
$\frac{\alpha}{\alpha+5}$ $\frac{3}{\alpha+5}$ $\frac{2}{\alpha+5}$

- The *rich get richer* - a table with lots of customers will attract more customers.



Chinese Restaurant Process

$\frac{\alpha}{\alpha+6}$ $\frac{1}{\alpha+6}$ $\frac{3}{\alpha+6}$ $\frac{2}{\alpha+6}$

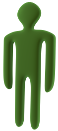


Chinese Restaurant Process

$\frac{\alpha}{\alpha+7}$ $\frac{1}{\alpha+7}$ $\frac{4}{\alpha+7}$ $\frac{2}{\alpha+7}$

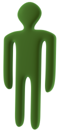
- The expected number of tables given α and n customers is:

$$\sum_{i=0}^{n-1} \frac{\alpha}{\alpha+i} = \alpha(\Psi(\alpha+n) - \Psi(\alpha)) \simeq \alpha \log(1+\frac{n}{\alpha})$$



Chinese Restaurant Process

$\frac{\alpha}{\alpha+8}$ $\frac{2}{\alpha+8}$ $\frac{4}{\alpha+8}$ $\frac{2}{\alpha+8}$

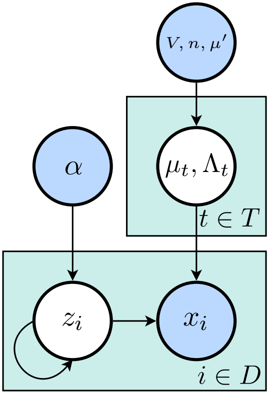


Chinese Restaurant Process

$$\frac{\alpha}{\alpha+\sum_{i=1}^{n} m_i} \qquad \frac{m_3}{\alpha+\sum_{i=1}^{n} m_i} \qquad \frac{m_2}{\alpha+\sum_{i=1}^{n} m_i} \qquad \frac{m_1}{\alpha+\sum_{i=1}^{n} m_i}$$

- m_i - The number of customers at table i.
- Whilst only four tables are shown the process goes on forever, leading to an infinite number of occupied tables, given infinite customers.

Chinese Restaurant Mixture Model



$\Lambda_t \sim \mathcal{W}(V, n)$ ($\mathcal{W}$ = Wishart distribution)

$\mu_t \sim \mathcal{N}(\mu', (n\Lambda)^{-1})$ ($\mathcal{N}$ = Gaussian distribution.)

$$P(z_i = t) = \begin{cases} \frac{m_t}{\alpha + \sum_{i \in T} m_i} & t \in T \\ \frac{\alpha}{\alpha + \sum_{i \in T} m_i} & t \notin T \end{cases}$$

$m_t = |\{i; z_i = t\}|$

$x_i \sim \mathcal{N}(\mu_{z_i}, \Lambda_{z_i}^{-1})$

- T is the set of 'tables' that have samples 'sitting' at them - a finite set.
- Consequentially, this is a finite structure, that can be Gibbs sampled without approximation.
- All three of the following applications use this, or variants of this.

Roadmap

Background Subtraction

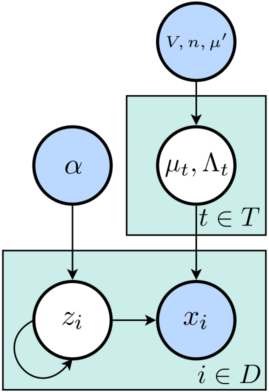
- Finds the interesting regions of a video.
- 'Blue screening without a blue screen'.
- Below by row: Input, ground truth, presented.

moved object	time of day	light switch	waving trees	camouflage	bootstrap	foreground aperture

Method

- Construct a per-pixel model of the background. . .
  . . . using a Dirichlet process Gaussian mixture model.
- Use Bayes rule to convert this density estimate to a class membership probability (foreground or background).
- Construct a Markov random field and regularise, solving with belief propagation (GPU friendly.).

Gibbs Sampling



- Gibbs sample the Chinese restaurant model, weighting new values by their probability of coming from the existing model.
- Integrate out μ_t and Λ_t - conjugate prior means we can use the student-t distribution and update incrementally.
- Sample the z_i using $P(z_i = t) \propto$
$$\begin{cases} \frac{m_t}{\alpha + \sum_{i \in T} m_i} P(x|V_t, n_t, \mu_t) & t \in T \\ \frac{\alpha}{\alpha + \sum_{i \in T} m_i} P(x|V, n, \mu') & t \notin T \end{cases}$$
- We have a never ending stream of data points - we sample each point only once, and immediately throw it away.

Forgetting

- As time passes the background can change - the model needs to forget the old background.
- This is achieved by capping the confidence and scaling such values back when they pass a threshold.
- This causes older sample to be repeatedly scaled to irrelevance as time passes, but only if the mode has changed.

Regularisation

- Standard Markov random field over image.
- We have $P(\text{data}|\text{background})$, we need $P(\text{background}|\text{data})$ - assume that $P(\text{data}|\text{foreground})$ is the uniform distribution and apply Bayes rule.
- An edge preserving cost is used between pixels, with a Cauchy distribution-like cost that depends on colour difference.
- Solved with belief propagation - graph cuts is optimal, but does not run as well on a GPU.

Further Details

- Background subtraction is an old area - it takes a certain amount of engineering to be competitive . . .
- Compensate for lighting change, using a mean shift based estimate.
- Custom colour model to reduce the effect of shadows.
- GPU implementation for speed.

Quantitative Results

- Big charts of numbers can be found in paper. . .
- . . . executive summary:

  SABS (synthetic): 27% improvement.

  Wallflower: 33% less mistakes.

  Star: 4% improvement.

  (Compared to nearest competitor in each case.)

Output - Wallflower

moved object | time of day | light switch | waving trees | camouflage | bootstrap | foreground aperture

(First row = input; second row = ground truth; third row = output)

Output - Star

cam ft ws mr lb sc ap br ss

(First row = input; second row = ground truth; third row = output)

Conclusions

- The Dirichlet process allows for a really good density estimate - it models multi-modal distributions and learns the amount of noise.
- Consequentially, it does really well at dynamic backgrounds that stump other algorithms. Its also great with camouflage.
- The method of forgetting learns model changes quickly, but keeps the old model around for a long time, to be reused if needed (Exponential falloff).

Roadmap

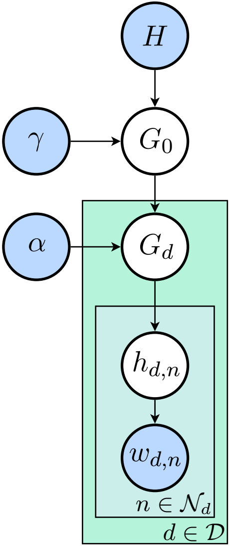
Topic Models

- Take a document and ignore the order of the words, to get a bag of words.
- Model a corpus of documents as draws from mixtures of distributions over words.
- The mixing ratio is document specific, whilst the distributions are shared - its a generalisation of a density estimate.
- The distributions are referred to as topics - they often match up with human perception, e.g. news articles will have topics such as sport, politics etc.

Abnormal Behaviour Detection

- Topic modelling can be generalised - for video discrete features are extracted as words and short clips used as documents. The topics then represent behaviours.
- This has motivated the construction of topic models with abnormal behaviour detection in mind, of which delta-dual hierarchical Dirichlet processes (dDHDP) is one example.
- A low model probability for a video clip indicates a previously unseen behaviour.

Hierarchical Dirichlet Processes



- Created by Yee Whye Teh et al.
- Generalisation of latent Dirichlet allocation (LDA) that learns the correct number of topics.
- Note that it uses one Dirichlet process as the base measure for another.

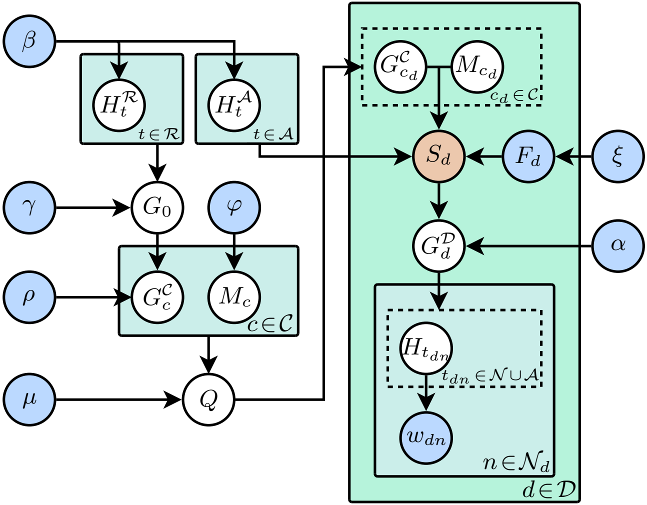
Dual Hierarchical Dirichlet Processes

- Created by Xiaogang Wang et al.
- Clusters documents, so each document is grouped with documents that have a similar distribution over topics.
- This allows normal topics that appear in an unusual configuration with other normal topics to look abnormal, e.g. a person crossing the road is normal, but not whilst cars are driving through the crossing.

Delta topic models

- Topic models are traditionally unsupervised, but for abnormal behaviour we want supervision.
- Because tagging which visual features constitute a topic is tedious this needs to be a form of semi-supervision.
- Delta topic models, a concept introduced by Andrzejewski et al., achieves this goal.
- You mark which documents have or do not have particular topics, but not which words were drawn from said topic.
- Delta-dual hierarchical Dirichlet processes combines this idea with DHDP.

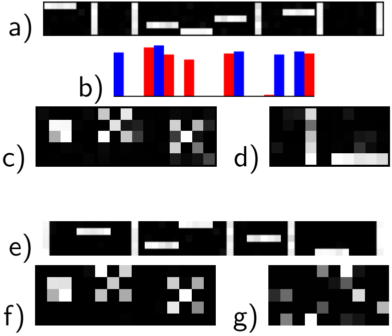
Graphical Model



Solving

- Gibbs sample it.
- There are a lot of random variables...
- ...and iterating how to sample each of them would be time consuming and boring - read the paper (And then the papers it references.).
- Have to use techniques such as (a modified version of) the left to right algorithm.
- Note that F_d is known during training, but unknown during testing.

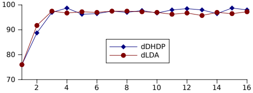
Demonstration

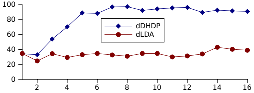


$5X5$ grid of words, visualised as pixels in an image, with 10 topics - 5 vertical and 5 horizontal lines. Only one orientation is in each document.

a-d is dDHDP, e-g is dLDA.

Both find abnormal topics (c & f), only dDHDP finds normal topics in abnormal context (d & g).





Mile End - Problem

- Mile end data set - 50 minutes of video of a traffic junction near QMUL.
- Two kinds of abnormality are used for *supervised training* - a u-turn (above, left) and driving from the middle area to the right whilst traffic continues to travel vertically (above, right).
- Many other abnormalities exist.

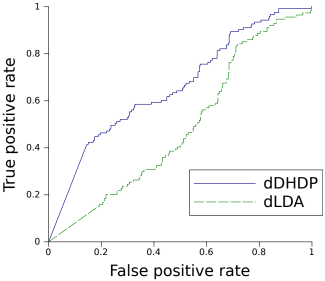
Mile End - Results

dDHDP $= 83.7\%$

364	13	37	87.9%
4	6	1	54.5%
22	3	45	64.3%

dLDA $= 74.2\%$

351	22	41	84.8%
0	11	0	100.0%
56	8	6	8.6%



- Trained on 8 minutes of video, tested on 42 minutes.
- Supervision used 2 examples of each behaviour.
- Confusion matrices - supervised detection only.
- ROC curve - supervised and unsupervised detection combined.

Conclusions

- The approach captures a class of behaviours that previous approaches could not...
- ...but it does so at the expense of a very complex model.

- It takes a long time to train the model...
- ...though can run in real time when analysing new documents.

Roadmap

Active Learning

- Training a classifier consists of **collecting data**, then **labelling the data** and, finally, **fitting a model**.
- Data collection can often be automated, and model fitting is a problem of computation... labelling however typically requires human interaction, and is hence *expensive*.
- Active learning endeavours to minimise this expense. It orders the training exemplars to get as much performance as possible with the least effort.
- When to stop training is usually left to the user.

Discovery & Classification

- **Discovery** is when not all classes are known, and need to be found.
- **Classification** is where the classes are considered to be known but the boundaries between them need to be refined.
- Active learning is typically used to solve one of these problems at a time.
- Here we present an approach that tackles both problems *simultaneously*, with the express purpose of *maximising classification performance*.

Scenario

- We have a *pool* of items with which to train a *classifier*.
- The task of the active learner is to, given the current classifier, select the best item to be labelled by the *oracle*.
- After each item has had a label supplied the classifier is updated with the new information (It helps if an incremental learning method is used.).

Assumptions

- *Assumption 1*: That the item with the greatest probability of being misclassified should be selected.
- *Assumption 2*: That the classes have been drawn from a **Dirichlet process**. This is equivalent to assuming the items in the pool come from a **Dirichlet process mixture model**.

- An infinite number of classes to which entities may belong.
- Classifier is Bayesian, but this can be ignored with a *pseudo-prior*.

The Algorithm

Class assignment that the classifier, which cannot consider new classes, gives:

$$\text{cc} = \operatorname*{argmax}_{c \in C} P_c(c|\text{data})$$

Class assignment probability, including the possibility of a new class under a Dirichlet process assumption:

$$P_n(c \in C \cup \{\text{new}\}|\text{data}) \propto \begin{cases} \frac{m_c}{\sum_{k \in C} m_k + \alpha} P_c(\text{data}|c) & \text{if } c \in C \\ \frac{\alpha}{\sum_{k \in C} m_k + \alpha} P(\text{data}) & \text{if } c = \text{new} \end{cases}$$

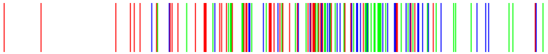
Probability of misclassification:

$$P(\text{wrong}|\text{data}) = 1 - P_n(\text{cc}|\text{data})$$

Concentration parameter (α) needs to be estimated - use the Gibbs sampling method from Escobar & West '95. Entity selection is done probabilistically, using $P(\text{wrong})$ as a weighting.

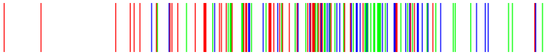
Demonstration

- Use Fisher *iris (orchid) classification problem* from 1936, reduced to 1D via PCA.

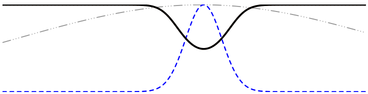


Demonstration

- Use Fisher *iris (orchid) classification problem* from 1936, reduced to 1D via PCA.

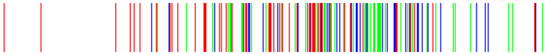


1 query:

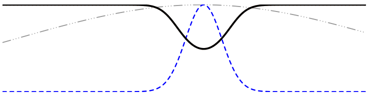


Demonstration

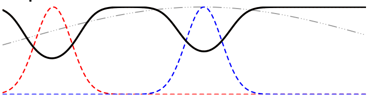
- Use Fisher *iris (orchid) classification problem* from 1936, reduced to 1D via PCA.



1 query:

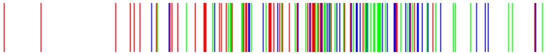


2 queries:

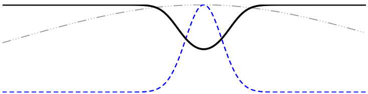


Demonstration

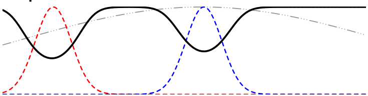
- Use Fisher *iris (orchid) classification problem* from 1936, reduced to 1D via PCA.



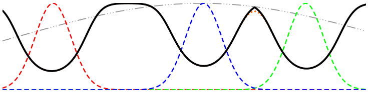
1 query:



2 queries:

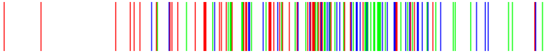


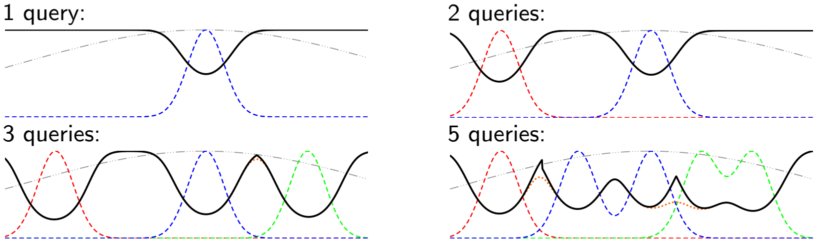
3 queries:



Demonstration

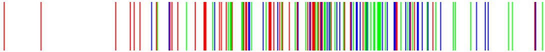
- Use Fisher *iris (orchid) classification problem* from 1936, reduced to 1D via PCA.



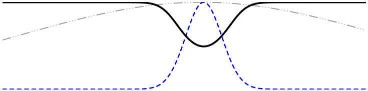


Demonstration

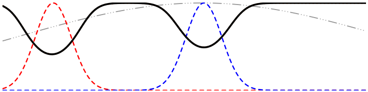
- Use Fisher *iris (orchid) classification problem* from 1936, reduced to 1D via PCA.



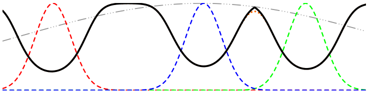
1 query:



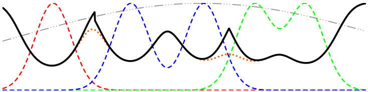
2 queries:



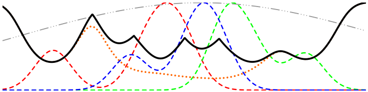
3 queries:



5 queries:

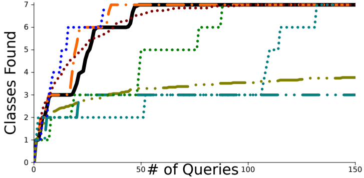


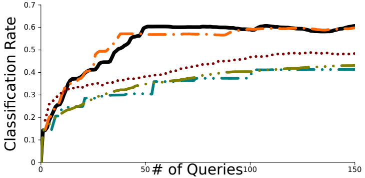
12 queries:



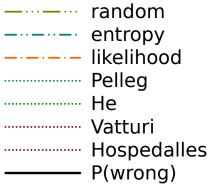
Shuttle

- Standard dataset from the UCI repository - included to compare with other algorithms.
- Seven classes; 78% of exemplars are in the largest class, 0.01% in the smallest.



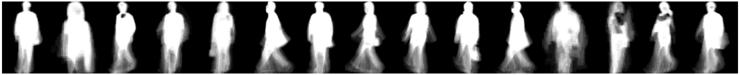


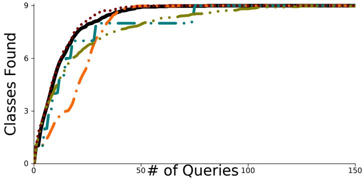
	shuttle	
	discovery	classification
random	486.2	53.5
entropy	423.5	51.8
likelihood	950.5	79.4
Pelleg	534.0	
He	768.5	
Vatturi	970.5	
Hospedales	933.2	61.8
P(wrong)	923.4	79.8

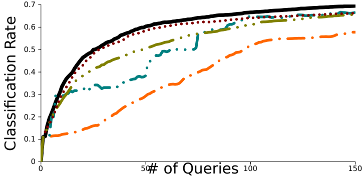


Gait

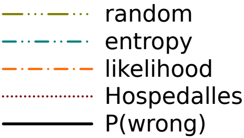
- Gait problem - recognising one of nine camera angles from a gait energy image. Geometric progression for sample sizes.





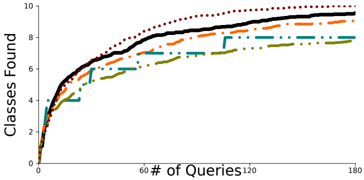


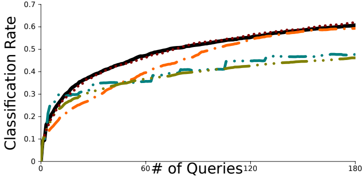
	gait	
	discovery	classification
random	1170.5	78.9
entropy	1183.8	75.3
likelihood	1171.7	56.5
Hospedalles	1253.1	84.8
P(wrong)	1241.9	88.4



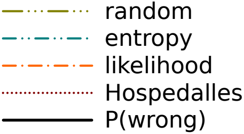
Digits

- Digits problem: Recognising the ten handwritten digits.



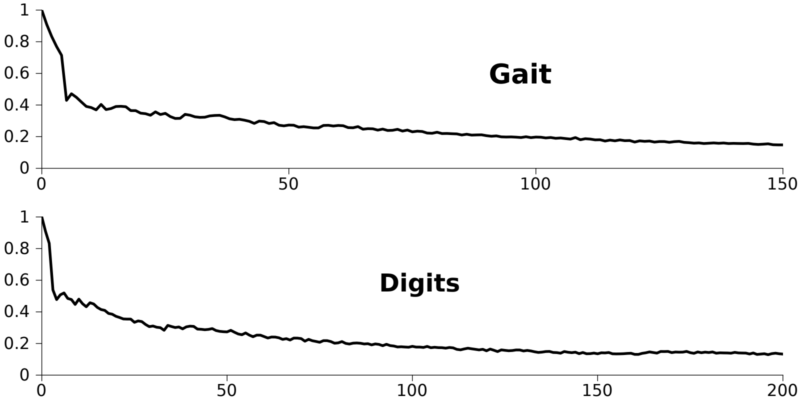


	digits	
	discovery	classification
random	915.2	54.6
entropy	974.0	57.1
likelihood	1060.2	61.9
Hospedales	1207.4	69.5
P(wrong)	1133.6	69.7



Interest in Finding New Classes

- Plots of the interest in finding a new class versus the number of queries.
- Glitch in graph due to concentration (α) estimation method requiring at least two classes.



Conclusions

- Simple to implement, good results.
- Minimal, if any, effort required for parameter tuning.
- Basic concept with many possible specialisations/improvements (Though surprisingly hard to find!).

Papers

- *Background Subtraction with Dirichlet Processes*, ECCV 2012
- *Delta-Dual Hierarchical Dirichlet Processes: A pragmatic abnormal behaviour detector*, ICCV 2011
- *Active Learning using Dirichlet Processes for Rare Class Discovery and Classification*, BMVC 2011

Last Words

- Dirichlet processes are great if you have to learn the correct number of instances of something in a fully Bayesian framework.
- Does a very good job at density estimation.
- Pitman-Yor processes are similar, but have a power law rather than logarithmic relationship.
- The dependent Dirichlet process allows for relationships between otherwise independent Dirichlet processes.