ESANN 2012 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence
and Machine Learning. Bruges (Belgium), 25-27 April 2012, i6doc.com publ., ISBN 978-2-87419-049-0.
Available from http://www.i6doc.com/en/livre/?GCOI=28001100967420.

# Learning Task Relatedness via Dirichlet Process Priors for Linear Regression Models

Marcel Hermkes and Nicolas M. Kuehn and Carsten Riggelsen [*]

University of Potsdam - Institute of Earth and Environmental Science
Karl–Liebknecht Str. 24/25, 14476 Golm–Potsdam - Germany

**Abstract**. In this paper we present a hierarchical model of linear regression functions in the context of multi–task learning. The parameters of the linear model are coupled by a Dirichlet Process (DP) prior, which implies a clustering of related functions for different tasks. To make approximate Bayesian inference under this model we apply the Bayesian Hierarchical Clustering (BHC) algorithm. The experiments are conducted on two real world problems: (i) school exam score prediction and (ii) prediction of ground–motion parameters. In comparison to baseline methods with no shared prior the results show an improved prediction performance when using the hierarchical model.

## 1 Introduction

In this study we consider the problem of multi–task learning [1–5], strictly speaking, learning multiple related predictive functions, for which the assumption is that the training data for each task is not identical distributed, but that similar tasks share some information. This learning task can be stated as follows: The data are observations from $K$ different tasks. The data set of the $k$–th task has the form $D_k = \{(\mathbf{x}_{ki}, y_{ki})\}_{i=1}^{n_k}$, where $n_k = |D_k|$ is the cardinality of the $k$–th data set, $\mathbf{x}_{ki} \in \mathbb{R}^d$ is th $i$–th covariate of the $k$–th task and $y_{ki} \in \mathbb{R}$ is the corresponding target value. Furthermore, $D = (\mathbf{X}, \mathbf{y})$ denotes the complete data set with $\mathbf{X} = \{\mathbf{X}_k\}_{k=1}^K$ and $\mathbf{y} = \{\mathbf{y}_k\}_{k=1}^K$. The aim in multi–task learning is to learn function estimators $f_k$ simultaneously to share some information in an arbitrary way.

A common technique in multi–task learning to share information across tasks is Hierarchical Bayesian modelling [1, 5], which makes the assumption that model parameters are drawn from a common prior distribution. By learning these parameters jointly the individual tasks will interact and regulate each other. A drawback of such a prior by reason of its modality is that the relationship between all tasks are treated equally, but it is desirable that only similar tasks share information to permit negative transfer. To deal with these issues we propose a nonparametric hierarchical Bayesian model where the common prior is drawn from a DP. The DP prior induces a partitioning of tasks with an infinite number of components, so that only similar tasks within each cluster share the same parameterization. A similar approach was previously proposed in the context of classification by Roy and Kaelbling [3] using a Naive Bayes classifier, and Xue et al. [4] using logistic regression. Related to Roy and Kaelbling, we apply the BHC [6] algorithm for performing inference in our model.

[*] This work was (partly) funded by the German Research Foundation, grant DFG RI 2037/2-1.

## 2 Background

In this section we give a short introduction into the topic of DP and Dirichlet Process Mixture (DPM) models as well as an algorithm for approximate inference in DPMs. An overview of DPs is given by Teh [7].

### 2.1 Dirichlet Process Mixture model

A DP is a stochastic process, whose realizations are probability distributions, i.e. it is a distribution over distributions. The distributions drawn from a DP are discrete. Let $\Theta$ be the latent parameters drawn from a random distribution $G$, which itself sampled from a DP with a base distribution $H$ and a positive concentration parameter $\alpha$, then the generative model can be written as $\theta_i \sim G$ and $G \sim DP(\alpha, H)$.

Let $\theta_1, \ldots, \theta_k$ be an i.i.d. sequence drawn from $G$. Due to the discreteness property of the DP, the values of draws are repeated, so the unique values of $\theta_1, \ldots, \theta_k$ are denoted by $\theta_1^\star, \ldots, \theta_m^\star$ and $n_j$ is the number of occurrences $\theta_j^\star$ in the random sequence. Thus, the predictive distribution of $\theta_{k+1}$ given $\theta_1, \ldots, \theta_k$ with $G$ integrated out can be written as $p(\theta_{k+1}|\theta_1, \ldots, \theta_k, \alpha, H) = \frac{1}{K+\alpha}(\alpha H + \sum_{j=1}^{m} n_j \delta_{\theta_j^\star})$ which implies the implicit clustering property of the DP [7]. The first term in the brackets of this expression reflects the ability of the DP in creating new clusters, which is proportional to $\alpha$, while the second term reflects the fact that new samples join groups with large samples, namely with a probability proportional to $n_j$.

The clustering property makes the DP prior very attractive in the application field, especially in clustering data with mixture models. Here, the parameters of the mixture components are drawn from a DP prior. The nonparametric nature of DP translates a mixture model with a fixed number of components to a mixture model with countable infinite number of components. This model is widely known as DPM model. The marginal likelihood of a DPM model [6] can be written as

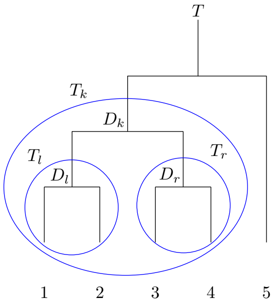
$$p(\mathbf{y}|\mathbf{X}, \alpha, \phi) = \sum_{\mathbf{z} \in Z} p(\mathbf{z}|\alpha) \prod_j p(\{\mathbf{y}_k : z_k = j\}_{k=1}^K | \{\mathbf{X}_k : z_k = j\}_{k=1}^K, \phi), \qquad (1)$$

where $z_k$ is the cluster assignment variable of task $k$, $p(\mathbf{z}|\alpha) = \int p(\mathbf{z}|\pi) p(\pi|\alpha) \, \mathrm{d}\pi$ is the standard Dirichlet integral and the last term is the marginal likelihood of data assigned to the $j$–th cluster. The sum over the exponential number latent partitions $Z$ makes exact Bayesian inference intractable. Instead of using MCMC sampling machinery which may be slow to converge, we apply the BHC algorithm to make approximate inference.

### 2.2 Bayesian Hierarchical Clustering Algorithm

The BHC algorithm [6] is similar to traditional agglomerative clustering, but with the distinction that it uses a Bayesian hypothesis test as merging criterion instead of an arbitrary distance measure. The BHC constructs a lower bound of Eq. 1, which approximates the sum over the latent partitions by summing over the exponential number of tree–consistent partitions, induced in a greedy manner by the agglomerative clustering procedure. Figure 1(a) illustrates the notion of tree–consistent partitions.

In the following we will explain the parts of the algorithm which are relevant for our multi–task learning model in Section 3. A detailed description of the algorithm can be found in [6]. The complete BHC algorithm is summarized in Figure 1(b).



```
input : data $D$, model $p(y|x,\theta)$, prior $p(\theta|\phi)$, and $\alpha$
while $P(D|T)$ does not significantly increase do
    (1) building tree structure:
    initialize: number of clusters $c = K$, and $\mathcal{B} = \{D_i\}_{i=1}^{K}$
    while $c > 1$ do
        Find the pair $D_l$ and $D_r$ with $\arg\max_k p(\mathcal{H}_k|D_k)$
        Merge $D_k \rightarrow D_l \cup D_r$,   $T_k \rightarrow (T_l, T_r)$
        Delete $\mathcal{B} \rightarrow \mathcal{B} \setminus \{D_r, D_l\}$, $c \rightarrow c - 1$

    (2) learning hyperparameters by maximizing $p(D|T)$:
    Optimize $\alpha$ by line search algorithm
    Optimize $\phi$ by taking gradient $\frac{\partial \log p(D|T)}{\partial \phi}$
```

*Fig. 1:* (a) Scheme of a cluster hierarchy of tasks, where $T_l$ and $T_r$ were merged into $T_k$ with the associated data set $D_k = D_l \cup D_r$. $T$ denotes the root node of the tree. For example, the partitioning $\{\{1,2\},\{3,4\},\{5\}\}$ and $\{\{1\},\{2\},\{3,4,5\}\}$ are tree–consistent, while $\{\{1,2,3\},\{4,5\}\}$ does not reflect a tree–consistent partitioning. (b) Pseudocode of the BHC algorithm.

The statistical test in the merging stage is based on comparing two different hypothesis. The null hypothesis $\mathcal{H}_k$ is that $D_k$ is i.i.d. from the same probabilistic model. This can be expressed simply by the marginal likelihood of the data $p(D_k|\mathcal{H}_k) = \int \prod_i^{n_k} p(y_{ki}|\mathbf{x}_{ki}, \theta) p(\theta|\phi)\, \mathrm{d}\theta$, where $p(\theta|\phi)$ is the prior over the latent parameters $\theta$ with hyperparameters $\phi$. The alternative hypothesis $\overline{\mathcal{H}}_k$ is that $D_k$ is generated by two or more clusters. Due to the restriction to tree–consistent partitions the distribution can be formulated by $p(D_k|\overline{\mathcal{H}}_k) = p(D_l|T_l)p(D_r|T_r)$. Thus, marginal likelihood of the BHC algorithm for a tree $T_k$ can be written as

$$p(D_k|T_k) = \pi_k p(D_k|\mathcal{H}_k) + (1 - \pi_k) p(D_l|T_l) p(D_r|T_r), \tag{2}$$

where $\pi_k \overset{\text{def}}{=} p(\mathcal{H}_k)$ is the prior that $D_k$ belongs to one cluster. Eq. 2 is a lower bound of Eq. 1 (see [6]), if and only if $\pi_k = \frac{\alpha\Gamma(n_k)}{d_k}$ with $d_k = \alpha\Gamma(n_k) + d_l d_r$, where $\alpha$ is the concentration parameter of the DPM and $\Gamma(\cdot)$ is the gamma function. By using the Bayes rule, the posterior of the merged hypothesis, which is also used as merging criterion in the BHC algorithm, is $p(\mathcal{H}_k|D_k) = \frac{\pi_k p(D_k|\mathcal{H}_k)}{p(D_k|T_k)}$.

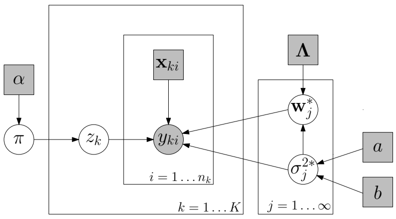
As shown in in Figure 1 (b) the set of hyperparameters is optimized by a line search and gradient descent algorithm after the tree was constructed. The gradient of Eq. 2 w.r.t. to model hyperparameters $\phi$ is (with $\omega \overset{\text{def}}{=} p(\mathcal{H}|D)$)

$$\frac{\partial \log p(D|T)}{\partial \phi} = \omega \frac{\partial \log p(D|\mathcal{H})}{\partial \phi} + (1-\omega) \left[ \frac{\partial \log p(D_l|T_l)}{\partial \phi} + \frac{\partial \log p(D_r|T_r)}{\partial \phi} \right]. \tag{3}$$

Finally, prediction can be made for an unseen sample $\mathbf{x}_*$ corresponding to task $k$ by summing over the predictive distribution of each node that includes the target tasks. Let $p(y_\star|x_\star, D_k) = \int p(y_\star|x_\star, \theta) p(\theta|D_k, \phi)\, \mathrm{d}\theta$ denote the predictive distribution of node $k$ and $\mathcal{A}_k$ denotes the set of nodes along the path from the root to the node $k$. So, the overall predictive distribution is defined by (see [3])

$$p(y_\star|x_\star, D) = \sum_{i \in \mathcal{A}_k} \frac{w_i}{\sum_{j \in \mathcal{A}_k} w_j} p(y_\star|x_\star, D_i), \tag{4}$$

where $w_k = p(\mathcal{H}_k|D_k) \prod_{i \in \mathcal{A}_k \setminus \{k\}} (1 - p(\mathcal{H}_i|D_i))$ is a weighting term on cluster $k$.



$$
\begin{aligned}
\pi &\sim GEM(\alpha) \\
z_k \mid \pi &\sim Mult(\pi) \\
\sigma_j^{2*} \mid a, b &\sim G^{-1}(a, b) \\
\mathbf{w}_j^* \mid \sigma_j^{2*}, \Lambda &\sim N(\mathbf{0}, \sigma_j^{2*}\mathbf{\Lambda}^{-1}) \\
y_{ki} \mid \mathbf{x}_{ki}, z_k, \{(\mathbf{w}_j^*, \sigma_j^{2*})\}_{j=1}^{\infty} &\sim N(\mathbf{w}_{z_k}^{*\top}\mathbf{x}_{ki}, \sigma_{z_k}^{2*})
\end{aligned}
$$

*Fig. 2:* A graphical model representation of our multi–task linear model using stick–breaking construction [7] with the corresponding probability distributions of the parameters. $GEM(\alpha)$ is the stick–breaking distribution over $\pi$ and each $z_k$ is drawn from a multinomial distribution.

## 3 Multi–Task Linear Model

In this section we propose a parametric linear model for multi–task learning, which jointly learns its parameters by coupling these with a DP prior. The model implies a clustering of related tasks, and therefore permits negative information transfer. The presented BHC algorithm learns the posterior distribution of the latent parameters, which are used to make Bayesian inference. For each task $k$ the $i$–th sample $y_{ki}$ is generated from a linear function $f(\mathbf{x}_{ki}) = \mathbf{w}_k^\top \mathbf{x}_{ki} + \epsilon_{ki}$, where $\epsilon_k \sim N(0, \sigma_k^2)$. If we assume that $y_{ki}$ are drawn i.i.d. from the underlying distribution of task $k$, then the model likelihood is given by $p(\mathbf{y}_k|\mathbf{X}_k, \theta_k) = N(\mathbf{X}_k^\top \mathbf{w}_k, \sigma_k^2 \mathbf{I})$ with $\theta_k = \{\mathbf{w}_k, \sigma_k^2\}$.

The aim is to learn the functions $f_k$ of each task $k$ jointly to share information across the tasks. In our proposed model we assume that similar tasks should share the model weights $\mathbf{w}_k$ and variances $\sigma_k^2$. This can be done by coupling the parameters $\theta_k$ by a DP prior, which implies a clustering of the linear models. The base distribution of the DP is specified by a normal inverse–Gamma distribution, which is the conjugate prior for the model, that is, we can analytically integrate out the latent parameters $\theta_k$. The parameters are generated by $\mathbf{w}_k \sim N(\mathbf{0}, \sigma_k^2 \mathbf{\Lambda}^{-1})$ and $\sigma_k^2 \sim G^{-1}(a, b)$ with the hyperparameters $\phi = \{\mathbf{\Lambda}, a, b\}$. From this, we see that the prior comprises a Normal prior on the coefficients given the noise term, with the assumption that their mass lies around zero and they are uncorrelated, i.e. $\mathbf{\Lambda} = \mathbf{I}$, and vague inverse–Gamma prior on the noise term, i.e. its parameters are initialized by $a = b = 10^{-3}$. The complete generative model is shown in Figure 2.

To learn the DPM model with linear function estimator using the BHC algorithm enabling for making Bayesian inference, we have to specify the posterior distribution of the parameters. By applying Bayes rule the posteriors are $p(\mathbf{w}_k, \sigma_k^2|D_k) = N(\mathbf{m}_N, \mathbf{\Lambda}_N) G^{-1}(a_N, b_N)$ with the posterior parameters $\mathbf{m}_N = \mathbf{\Lambda}_N^{-1} \mathbf{X}_k^\top \mathbf{X}_k \mathbf{y}_k$, $\mathbf{\Lambda}_N = \mathbf{\Lambda} + \mathbf{X}_k^\top \mathbf{X}_k$, $a_N = a + n_k/2$ and $b_N = b + \frac{1}{2}[\mathbf{y}_k^\top \mathbf{y}_k - \mathbf{m}_N^\top \mathbf{\Lambda}_N \mathbf{m}_N]$. Thus, the marginal likelihood of $D_k$ by integrating $\mathbf{w}_k$ and $\sigma_k^2$ out is given by

$$
p(D_k) = \frac{|\mathbf{\Lambda}|^{\frac{1}{2}} b^a \Gamma(a_N)}{|\mathbf{\Lambda}_N|^{\frac{1}{2}} (b_N)^{a_N} \Gamma(a) (2\pi)^{\frac{2}{n}}}, \tag{5}
$$

which can be inserted into Eq. 2 to build the cluster hierarchy. In the next step, for optimizing the hyperparameters it is required to specify the gradients of the log marginal likelihood (Eq. 5) wrt. to the parameters $\phi$. The gradients can then be placed into Eq. 3. Note that the precision matrix $\mathbf{\Lambda}$ is a positive–semidefinite symmetric matrix and can be decomposed into $\mathbf{\Lambda} = \mathbf{L}\mathbf{L}^\top$. In order to keep this property of $\mathbf{\Lambda}$ by the gradient ascent procedure we have to update the lower triangular matrix $\mathbf{L}$. The gradients wrt. $a$, $b$ and $\mathbf{L}$ are

$$\frac{\partial \log p(D_k)}{\partial a} = \log a - \log b_N + \Psi(a_N) - \Psi(a), \qquad \frac{\partial \log p(D_k)}{\partial b} = \frac{a}{b} - \frac{a_N}{b_N},$$

$$\frac{\partial \log p(D_k)}{\partial \mathbf{L}} = \left\{ \left[\mathbf{\Lambda}^{-1} - \mathbf{\Lambda}_N^{-1}\right] + \left[\left(\frac{a}{b} - \frac{a_N}{b_N}\right)\mathbf{\Lambda}_N^{-1}\mathbf{B}\mathbf{\Lambda}_N^{-1}\right]\right\}\mathbf{L}, \tag{6}$$

where $\mathbf{B} = \mathbf{X}_k\mathbf{y}_k\mathbf{y}_k^\top\mathbf{X}_k^\top$ and $\Psi(x) = \frac{\partial \log \Gamma(x)}{\partial x}$ is the digamma function. Finally the predictive distribution follows a Student–t distribution, which has the following form

$$p(y_*|\mathbf{x}_*, D_k) = St(\mathbf{x}_*^\top \mathbf{m}_N, b_N(1 + \mathbf{x}_*^\top \mathbf{\Lambda}_N^{-1}\mathbf{x}_*), b_N), \tag{7}$$

where the mode $\mathbf{x}_*^\top \mathbf{m}_N$ of the distribution is used as predictive value. The overall predictive distribution of the BHC model is determined by placing Eq. 7 into Eq. 4.

## 4 Experimental Results

We evaluate our multi–task learning approach on two real world problems: (i) exam score prediction and (ii) prediction of ground–motion intensity parameters. For comparison, we have also applied a Bayesian linear regression model, which is learned for each task separately (STL) and on the complete data set (CPL). The model weights are MAP estimates learned via the EM algorithm. The performance measure employed is the mean squared error. Table 1 shows that our approach (BHC MTL) outperforms the baseline methods on all data sets.

| | CPL | STL | BHC MTL |
|---|---|---|---|
| School | 0.6853 (0.0161) | 0.6580 (0.0164) | 0.6355 (0.0125) |
| NGA | 0.4197 (0.1646) | 0.5412 (0.4235) | 0.3857 (0.1868) |
| Allen & Wald | 0.2792 (0.0507) | 0.2739 (0.0691) | 0.2689 (0.0691) |

*Table 1:* Mean squared error for the different algorithms on the school and two ground–motion data sets. The figures in brackets are standard errors.

The **school data** has been used to study the effectiveness of schools.[1] It is a 50% sample of examination records from 139 secondary schools in years 1987–1989 containing 15362 records. Each task in our setting is defined by the prediction of exam scores for students of a specific school. For comparison to previous studies [1, 2] we have used the same 10 random splits of 75% training data and 25% test data and follow the same preproccesing steps. In our experiments we have discarded the school–dependent features, by the reason that they may differ for the same school in different years. Previous studies [1, 2] reported their results in terms of explained variance. For comparison the explained variance of our proposed model is 37.07% which is an increase compared to the best value of 33.08% found in Bonilla et al. [2]. Furthermore the experiments has been conducted on two **ground–motion data sets**: (i) the Next Generation of Attenuation (NGA) [8] data set and (ii) the data set of Allen & Wald [9]. We refer to Kuehn et al. [10] to get a detailed review in the problem of ground–motion

[1] Data is available at `http://www.cmm.bristol.ac.uk/learning-training/multilevel-m-support/datasets.shtml`

ESANN 2012 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 25-27 April 2012, i6doc.com publ., ISBN 978-2-87419-049-0. Available from http://www.i6doc.com/en/livre/?GCOI=28001100967420.

prediction. In this experiment the aim is the prediction of ground–motion intensity parameters given earthquake– and site–related parameters in which each region represents a task. The records were mapped to 10 geographical regions as shown in [10]. Further, we have discarded regions with less than 2 earthquakes and less than 5 samples. Both data sets were preprocessed by using a binary representation of categorical features and standardizing numerical features, also samples with missing values were removed. After preprocessing the NGA data consists of 2641 samples over 5 regions and the data of Allen & Wald consists of 14542 over 8 regions.[2] For the experiments we have performed a 10–fold cross validation. To guarantee that our algorithm predicts intensity parameters well for future earthquakes at specific site in a region, we take into account that no two records of the same earthquake might occur in both training and test data.

## 5 Conclusion and Future Work

We presented a hierarchical modelling approach for learning related linear function estimators in the context of multi–task learning. A DP prior was used to model the relatedness of different tasks. The results show that clustering of linear functions outperforms the models in which no information is shared. In future work we hope to further improve our model by replacing the linear model by Gaussian Processes. Furthermore, we plan to investigate how we can make prediction for novel tasks, in which no training data is available. We can deal with this by extending our model with an extra Gaussian component over task specific features, which are also coupled by the DP prior. This Gaussian component can be considered as gating function, that determines the responsibility for each task with respect to the novel task. A problem arises in the context of seismological data, in which the relation between earthquakes and different sites are nested, i.e. the i.i.d. assumption is violated. Hence, we will extend the model with a hierarchical DP prior to capture the relatedness of earthquakes in different regions.

## References

[1] B. Bakker and T. Heskes. Task Clustering and Gating for Bayesian Multitask Learning. *Journal of Machine Learning Research*, 4:83–99, 2003.

[2] E.V. Bonilla, K. Ming, A. Chai, and C. K. I. Williams. Multi-task Gaussian process prediction. *Advances in Neural Information Processing Systems*, 20:153–160, 2008.

[3] D.M. Roy and L.P. Kaelbling. Efficient Bayesian task-level transfer learning. In *Proceedings of the 20th Joint Conference on Artificial Intelligence*, 2007.

[4] Y. Xue, X. Liao, L. Carin, and B. Krishnapuram. Multi-task learning for classification with Dirichlet process priors. *The Journal of Machine Learning Research*, 8:35–63, 2007.

[5] K. Yu, V. Tresp, and A. Schwaighofer. Learning Gaussian processes from multiple tasks. In *Proceedings of 22nd International Conference on Machine Learning (ICML)*, 1012–1019, 2005.

[6] K.A. Heller and Z. Ghahramani. Bayesian hierarchical clustering. In *22th ICML*, pages 297–304, 2005.

[7] Y.W. Teh. Dirichlet Process. *Submitted to Encyclopedia of Machine Learning*, 2007.

[8] B. Chiou, R. Darragh, N. Gregor, and W. Silva. NGA Project Strong-Motion Database. *Earthquake Spectra*, 24:23–44, 2008.

[9] T. I. Allen and D. J. Wald. Evaluation of Ground-Motion Modeling Techniques for Use in Global ShakeMap. *USGS Open–File Report*, 2009–1047.

[10] N.M. Kuehn, C. Riggelsen, F. Scherbaum, and T. I. Allen. A Bayesian Hierarchical Global Ground-Motion Model to Take into Account Regional Difference. *submitted to Bull. Seism. Soc. Am.*, 2010.

[2] To obtain the preprocessed data sets please contact: `hermkes@geo.uni-potsdam.de`