Multi-Task Learning: The Bayesian Way

Tom Heskes

Radboud University Nijmegen

July 12, 2006

Contents

Newspaper sales

De Telegraaf

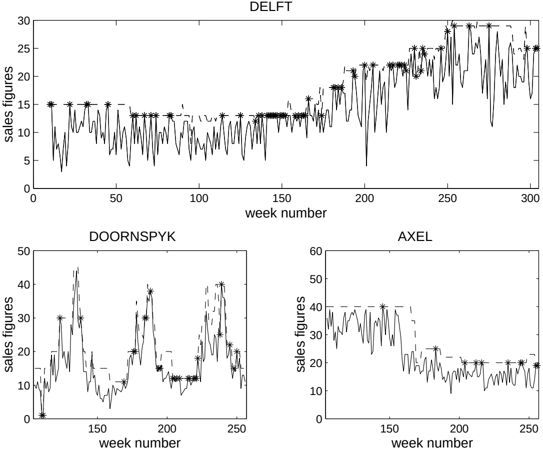
- Major Dutch newspaper (circulation over 1 million).
- 15.000 outlets.
- 7 days a week.

"Right of return"

- History of roughly 5 years.
- Delay of 4 weeks between most recent sales figure and delivery.


Telesport ● DROOM HUILENDE AGASSI SPAT UITEEN TEGEN NADAL
De Telegraaf ZONDAG
HUSHOVD PAKT EERSTE GELE TRUI
Lubbers informateur
Fransen onttronen Brazilië
AFZIEN op de Cauberg
Muntschat inzet van OORLOG
TURKIJE, maar dan anders
Ook Portugezen naar halve finale
KOZIJN VERVANGEN?
VANDAAG GEGARANDEERD 30°C
Het roer moet om
ENGELSEN 'VERBETEREN' ORANJE
dat is slim
thuis bellen zonder abonnements-kosten

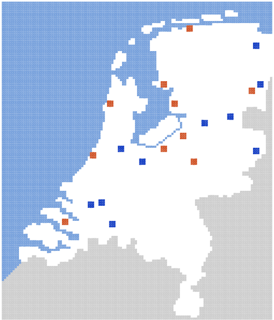
Data



Low signal-to-noise ratio.

Explanatory variables

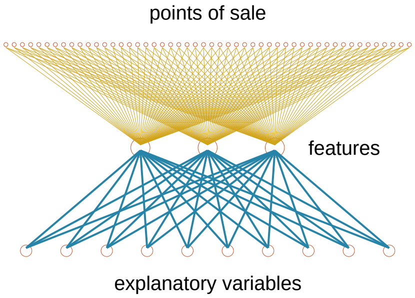
- recent sales (and sellouts)
- last year's sales
- season
- holidays
- weather
- news content
- ...



- more sales with nice weather
- less sales with nice weather

Many (possible) explanatory variables.

"Classical" multi-task learning



Model for the sales $y_i(t)$ of point of sale i in week number t given explanatory variables $x(i,t)$.

Hidden units:

$$f_k(i,t) = \tanh\left(\sum_j \Psi_{kj} x_j(i,t)\right) .$$

Output units:

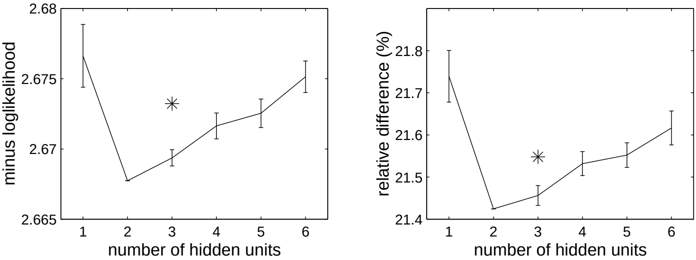
$$y_i(t) = \sum_k \theta_{ik} f_k(i,t) .$$

All points of sale combined into one big network, sharing the first layer.

Caruana, Machine Learning, 1997; Thrun & Pratt, "Learning to Learn", 1997.

Does it help?

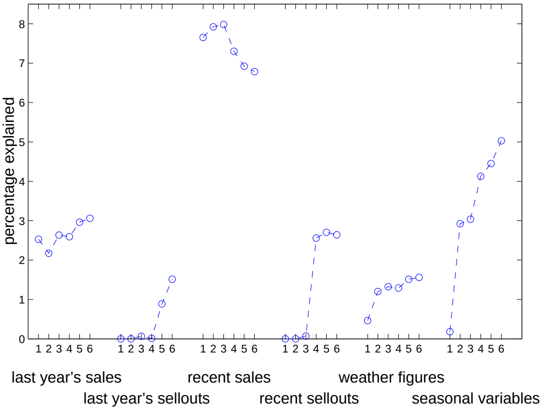
Comparison with carefully handcrafted features



Overfitting starts with three hidden units...

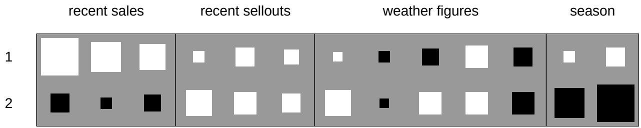
Heskes, ICML, 2000.

Does it make sense (1)?



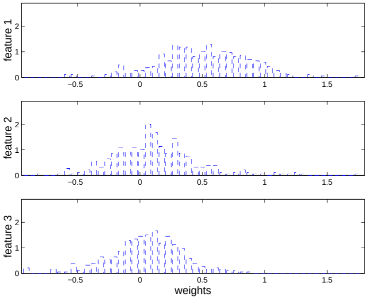
Different aspects enter with the addition of each hidden unit.

Does it make sense (2)?



First hidden unit mainly represents recent sales (short term effects); second hidden unit mainly seasonal and weather aspects.

The Bayesian way



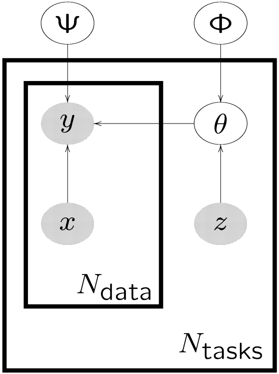
Empirical distribution of maximum-likelihood solutions for the task-specific parameters θ_i, maximizing[a] $P(y_i|\theta_i)$.

[a] For notational conveniences, we consider the inputs x and weights Ψ fixed and given.

Suggests:

- Treat the task-specific parameters as random variables. . .
- . . . for which we can define priors . . .
- . . . and then compute posteriors using Bayes' rule.

Summary of the model



- x: inputs, e.g., explanatory variables in a particular week for a specific outlet;
- y: outputs, e.g., sales in a particular week for a specific outlet;
- z: task-specific properties, e.g., location of an outlet;
- θ: task-specific parameters, e.g., hidden-to-output weights in MLP;
- Φ: hyperparameters specifying the prior on θ;
- Ψ: other shared parameters, e.g., input-to-hidden weights in MLP.

Priors on the task-specific parameters

Obvious choice:

$$P(\theta_i|\Psi) = \mathcal{N}(\theta_i; m, V) \; ,$$

a Gaussian with mean m and covariance matrix V, i.e., $\Psi = \{m, V\}$.

Other choices:

- a mixture of Gaussians (task-clustering):

$$P(\theta_i|\Psi) = \sum_\alpha \pi_\alpha \mathcal{N}(\theta_i; m_\alpha, V_\alpha) \; ;$$

- a "mixture-of-experts" prior (task-gating):

$$P(\theta_i|\Psi) = \sum_\alpha \pi_\alpha(z_i) \mathcal{N}(\theta_i; m_\alpha, V_\alpha) \; ,$$

  for example with

$$\pi_\alpha(z_i) = \frac{\exp \sum_l \gamma_{\alpha l} z_{il}}{\sum_{\alpha'} \exp \sum_l \gamma_{\alpha' l} z_{il}} \; .$$

Empirical Bayes

Maximize the loglikelihood

$$\log P(y|\Phi) = \sum_i \log \int d\theta P(y_i|\theta) P(\theta|\Phi) \,,$$

with respect to Φ.

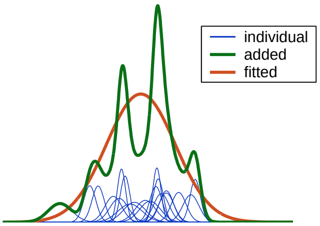
- Called empirical Bayes or type-II maximum likelihood procedure.
- Motivated as an approximation to hierarchical Bayes: since we can use all data to infer the hyperparameters, we do not have to integrate them out.

Expectation Maximization

Expectation-Maximization algorithm.

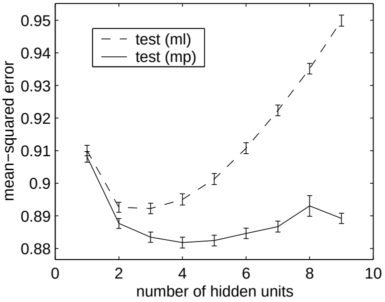
- E-step: compute $P(\theta|y_i, \Phi_{\rm old})$ for all tasks i.
- M-step: update $\Phi_{\rm old}$ to $\Phi_{\rm new}$ maximizing

$$\sum_i \int d\theta P(\theta|y_i, \Phi_{\rm old}) \log P(\theta|\Phi) \,.$$



See also: Schwaighofer, Yu, & Tresp, NIPS 17, 2005.

Does it help (1)?



Best performance requires both a bottleneck ("feature extraction") and the Bayesian part ("regularization")

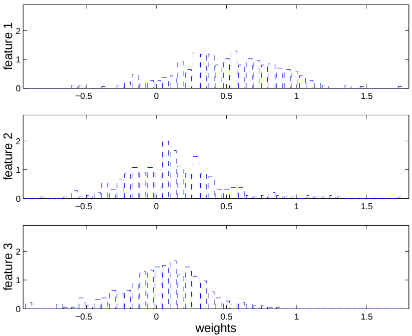
Heskes, ICML, 2000.

Does it help (2)?

Commercial product, consistently outperforms competitors.

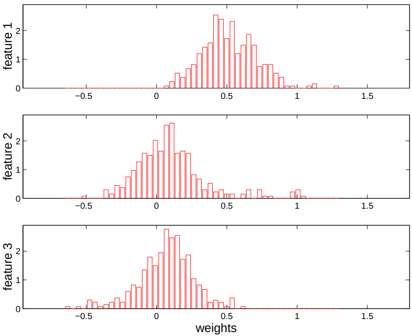
Heskes et al., Neural Computing and Applications, 2004

Does it make sense (1)?



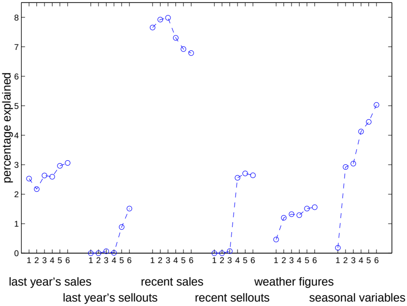
Maximum-likelihood solutions

$\Longrightarrow$



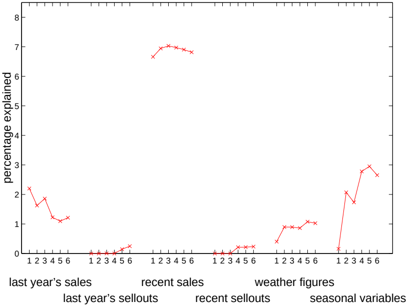
Maximum-a-posteriori solutions

Does it make sense (2)?



Maximum-likelihood solutions

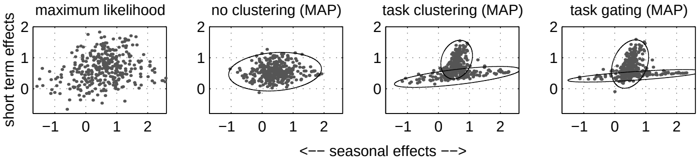
$\Longrightarrow$



Maximum-a-posteriori solutions

How about different priors?

Technically hardly more complicated. Results make sense. . .



. . . but do not improve performance by any significant amount (in this case).

Bakker & Heskes, JMLR, 2003.

Outlook

Multitask learning lends itself nicely for a Bayesian approach.

- Direct interpretation of the prior (even if your not a Bayesian).
- Empirical Bayes for learning the hyperparameters as well as other parameters shared between the tasks.
- Good performance.

A lot of work to do:

- appropriate models, e.g., for time series analysis;
- approximate inference.

But once translated not so different from other Bayesian technology...

Comparison with kernel approaches

Multi-task linear models with Gaussian priors are Gaussian processes.

- What does the kernel for a bottleneck MLP look like (probably easy)?
- . . . and for task-clustering and gating priors?
- What are we better at: to come up with appropriate kernels or with appropriate models and priors?
- Which approach is most sensitive to a suboptimal choice?
- Which is the most efficient approach?

E.g., Schwaighofer et al., NIPS 17, 2005; Evgeniou et al., JMLR, 2005.

Technicalities

We (often) seem to "assume"

- same inputs for all tasks;
- same noise variance (σ) for all tasks.

Procedures and analysis get quite complicated if we do not. Any "easy" solutions?

Any benchmark data sets for multitask learning?