Multi-output Chain Models and their Application in Data Streams

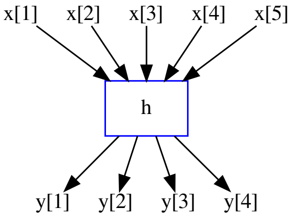
Jesse Read and Luca Martino

Outline

Multi-Output Learning (Multi-label Classification)

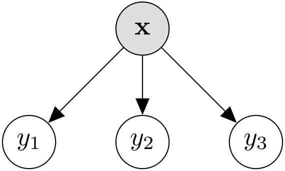
X_1	X_2	X_3	X_4	X_5	Y_1	Y_2	Y_3	Y_4
1	0.1	3	A	NO	0	1	1	0
0	0.9	1	C	YES	1	0	0	0
0	0.0	1	A	NO	0	1	0	0
1	0.8	2	B	YES	1	0	0	1
1	0.0	2	B	YES	0	0	0	1
0	0.0	3	A	YES	?	?	?	?

- Given: $\mathcal{D} = \{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^n$
- We want: model h such that $\widehat{\mathbf{y}} = h(\mathbf{x}) \approx \mathbf{y}$.



e.g., $\mathbf{x}$ is a text document; we want relevant labels ($\Leftrightarrow y_1 = 1$)

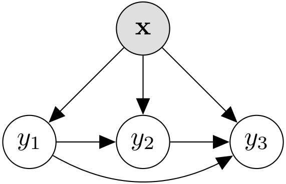
Use Independent Models?

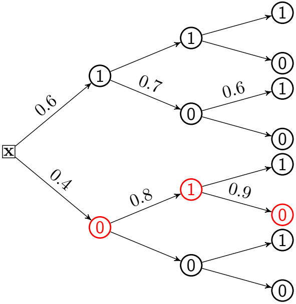


$$\widehat{\mathbf{y}} = [h_1(\mathbf{x}), h_2(\mathbf{x}), h_3(\mathbf{x})]$$

Why not?
Short answer: it works better modeling relationships among labels

Classifier Chains

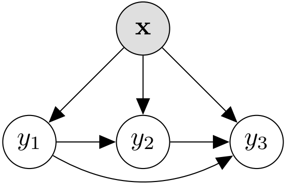


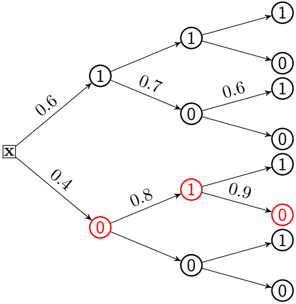


- Predictions cascade along a chain (as additional features)
- Use any suitable base classifier
- May suffer error propagation; probabilistic model can help: $\hat{\mathbf{y}} = \mathrm{argmax}_{\mathbf{y} \in \{0,1\}^L} \prod_{j=1}^{L} P(y_j | \mathbf{x}, y_1, \ldots, y_{j-1})$
- i.e., inference becomes a search e.g., Monte Carlo search[1]
- State of the art performance/benchmark method

[1] Read, Martino, and Luengo, Pat. Rec. 2014

Classifier Chains





- Predictions cascade along a chain (as additional features)
- Use any suitable base classifier
- May suffer error propagation; probabilistic model can help: $\hat{\mathbf{y}} = \text{argmax}_{\mathbf{y} \in \{0,1\}^L} \prod_{j=1}^{L} P(y_j | \mathbf{x}, y_1, \ldots, y_{j-1})$
- i.e., inference becomes a search e.g., Monte Carlo search[1]
- State of the art performance/benchmark method

And when outputs are continuous?

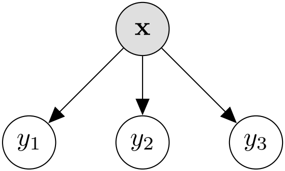
[1] Read, Martino, and Luengo, Pat. Rec. 2014

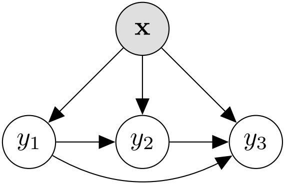
Multi-Output Regression

X_1	X_2	X_3	X_4	X_5	Y_1	Y_2	Y_3
1	0.1	3	A	NO	37.00	25	0.88
0	0.9	1	C	YES	-22.88	22	0.22
0	0.0	1	A	NO	19.21	12	0.25
1	0.8	2	B	YES	88.23	11	0.77
1	0.0	2	B	YES	0	0	0.08
1	0.0	2	B	YES	?	?	?

e.g., $\mathbf{x}$ is an image, $\widehat{\mathbf{y}} =$ time, temperature, date, . . . of image

- Individual regressors vs
- "Regressor Chains"?

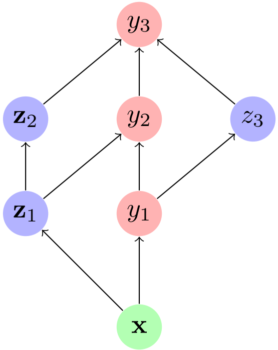




- greedy inference (single propagation),
  but may be pointless, or worse (divergence along the chain)
- probabilistic inference – not tractable, no tree to search
  and what are we optimizing?

It's all about Label Dependence?

Not really. If we unravel the chain, we get a "deep" neural network:



($\mathbf{z}_1 = \mathbf{z}_2 = \mathbf{x}$, and $z_3 = y_1$)

it's deep in the label space! Classification has a natural non-linearity, regression not necessarily!

What are we Optimizing?

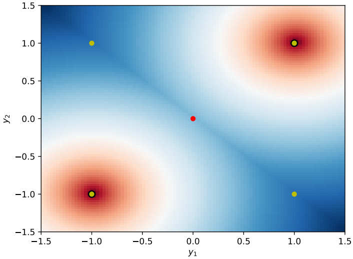


Figure: Left: A bimodal joint distribution over two labels (ground truth, given some x). Red $=$ MMSE estimator; yellow $=$ MMAE estimator; black $=$ MAP estimate (mode).

Outline

Regressor Chains

Motivation: A Regressor Chains models, where:

- Able to optimize different metrics (other than MSE)
- Outputs should serve as a non-linear representation for other outputs
- "Error propagation" (path degeneration) should be limited
- Able to offer interpretation/explainability

Sequential Monte Carlo Regressor Chains (SMCRC)

Given a test instance $\mathbf{x}$:

For $j = 1, \ldots, L$: $\triangleright$ Across the outputs

For $m = 1, \ldots, M$: $\triangleright$ For each particle

$$y_j^{(m)} \sim f(y_j | \tilde{y}_1^{(m)}, \ldots, \tilde{y}_{j-1}^{(m)}) \quad \triangleright \text{ Draw samples}$$

$$w_j^{(m)} = w_{j-1}^{(m)} \frac{\ell(y_j^{(m)} \mid \mathbf{x}, \tilde{y}_1^{(m)}, \ldots, \tilde{y}_{j-1}^{(m)})}{f(y_j^{(m)} \mid \tilde{y}_1^{(m)}, \ldots, \tilde{y}_{j-1}^{(m)})} \quad \triangleright \text{ Transition weights}$$

If $\widehat{ESS}(\bar{w}_j^{(1:M)}) \leq \eta M$:

$$\{\tilde{y}_j^{(1)}, \ldots, \tilde{y}_j^{(M)}\} \sim \{y_j^{(1)}, \ldots, y_j^{(M)}\} \quad \triangleright \text{ Resample} \propto \bar{w}_j^{(m)}$$

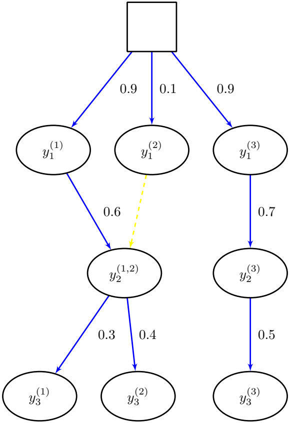
and (Optional) Apply K steps of MCMC or AIS.

We need to learn f (e.g., kernel density estimate), and ℓ (e.g., Bayesian regression) from training data

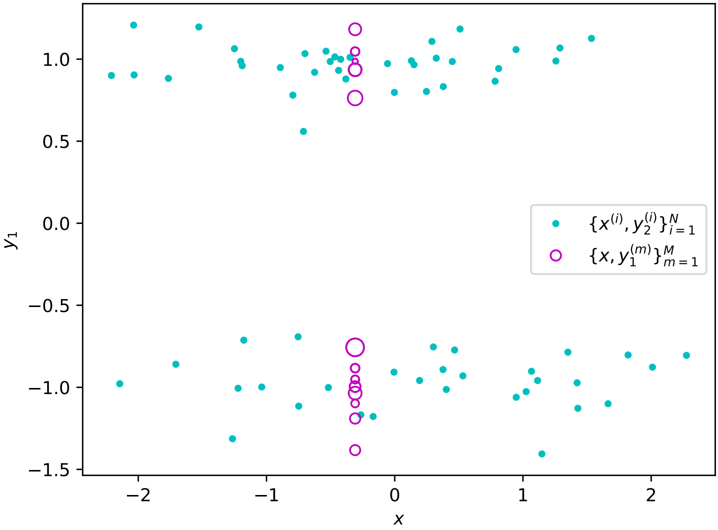
Output:

$$\widehat{\mathbf{y}}^{\text{MAP}} = \widehat{\mathbf{y}}^{(m^*)} \quad \text{where} \quad m^* = \operatorname*{argmax}_m w^{(m)}$$

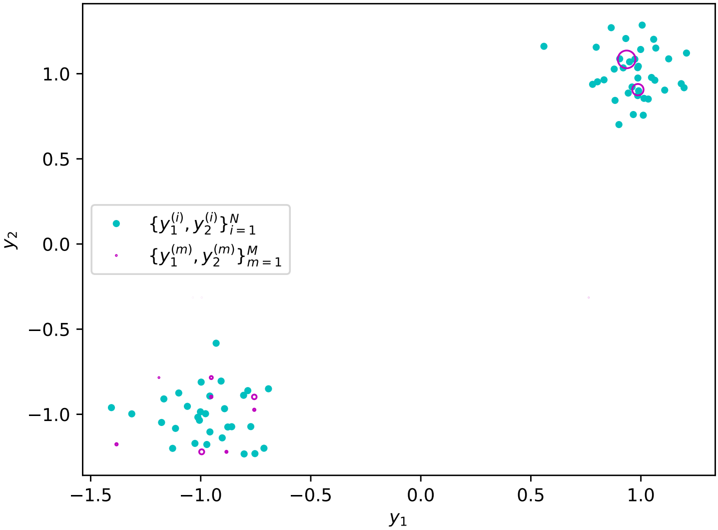
$$\widehat{\mathbf{y}}^{\text{MSE}} = \sum_{m=1}^{M} \mathbf{y}^{(m)}\, \bar{w}^{(m)}$$



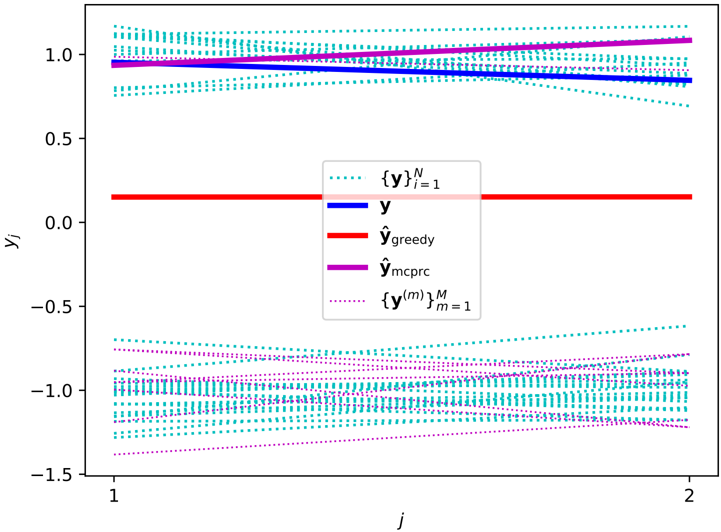
$$w^{(m)} := w_1^{(m)} \cdot w_2^{(m)} \cdot \cdots \cdot w_L^{(m)}$$



$y_1^{(m)} \sim f_1(\cdot|x)$ ▷ for some test instance x



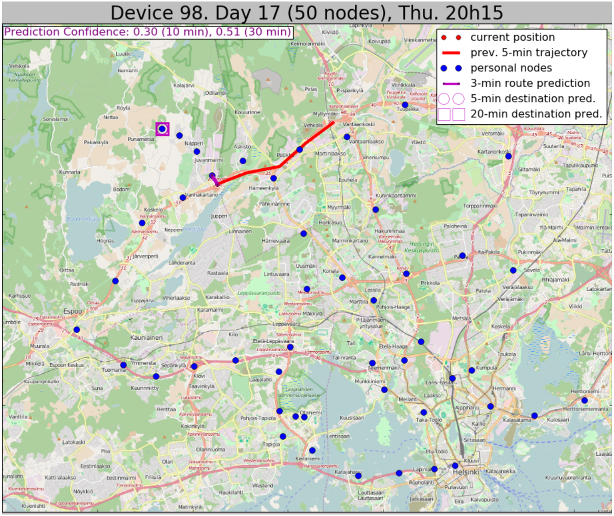
$$y_2^{(m)} \sim f_2(\cdot|x, y_1^{(m)})$$



Greedy chains vs SMCRC (MAP estimate)

Outline

Route Forecasting

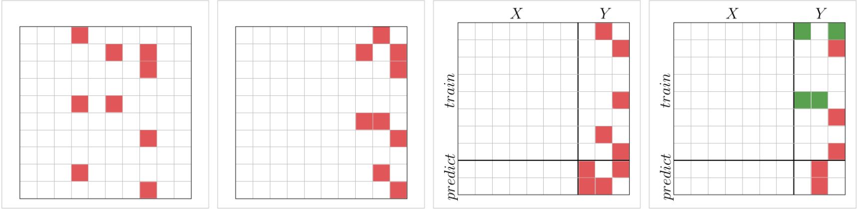


Personal nodes of a traveller and a predicted trajectory

Read, Martino, and Hollmén, Pat. Rec. 2017

Missing-Value Imputation

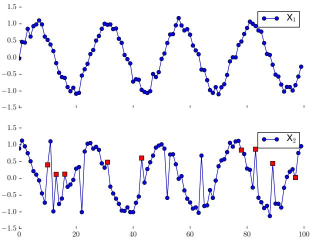
- Treat missing values as [unknown] labels to *impute* (i.e., predict).



A set/stream of data transformed into a multi-output prediction problem.

Montiel et al., PAKDD 2018, and manuscript under review

Anomaly Detection and Interpretation



- Create chains across feature space and through time
- Can generate likely paths over the 'gap'
  (expand the number of samples if necessary)

Song et al., ICDM demo 2018, and Elvira et al., manuscript in preparation

Summary

1. Multi-Output Prediction
2. Sequential Monte Carlo Regressor Chains
3. Applications and Related Methods

Conclusions

- Application of 'chain models' to the multi-output regression case is not straightforward;
    - Off-the-shelf application can be useless, or worse on account of
    - Error propagation
- Sequential Monte Carlo Regressor Chains
    - Weighted samples through the output space
    - Resampling avoids error propagation
    - Able to obtain a MAP estimate
    - Competitive, especially on multi-modal data
    - Useful for real applications requiring interpretation
- Related to many other methods (such as state space models, GPs, ResNets, . . .)
- Many other details (e.g., chain order, . . .) to deal with

Multi-output Chain Models and their Application in Data Streams

Jesse Read and Luca Martino

http://www.lix.polytechnique.fr/~jread/