Causal Phenotype Discovery via Deep Networks

Dave Kale$^{1,2}$, Zhengping Che$^1$
M. Taha Bahadori$^1$, Wenzhe Li$^1$, Yan Liu$^1$
Randall Wetzel$^2$

$^1$ University of Southern California, Computer Science
$^2$ Laura P. and Leland K. Whittier VPICU, Children’s Hospital LA

November 20, 2015
Disclosures

- D. Kale, Z. Che, T. Bahadori, W. Li, and Y. Liu have no commercial or financial interests related to this work.
- R. Wetzel is CEO of Virtual PICU (VPS) Systems, LLC.

Funding

- D. Kale is funded by a Innovation in Engineering Fellowship from the Alfred E. Mann Institute at USC.
- The VPICU is funded by a grant from the Laura P. and Leland K. Whittier Foundation.
Outline

1. Background: why and how of computational phenotyping
   - Phenotypes: representations of illness
   - Computational phenotyping
   - Phenotyping as representation learning

2. Phenotyping clinical time series with deep learning
   - Deep learning for time series
   - Causal analysis of phenotypic representations

3. Experiments
   - Setup
   - Prediction results
   - Visualization of causal phenotypes

4. Conclusion

5. References
Outline

1. Background: why and how of computational phenotyping
   Phenotypes: representations of illness
   Computational phenotyping
   Phenotyping as representation learning

2. Phenotyping clinical time series with deep learning
   Deep learning for time series
   Causal analysis of phenotypic representations

3. Experiments
   Setup
   Prediction results
   Visualization of causal phenotypes

4. Conclusion

5. References
Electronic (or computational) phenotyping

**Rules/algorithms** that define diagnostic/inclusion criteria [PheKB].

![Algorithm for identifying T2DM cases in the EMR.](image-url)
Electronic (or computational) phenotyping

**Rules/algorithms** that define diagnostic/inclusion criteria [PheKB].

**Classifiers** that answer the question “does patient have X?” [AL14] [AP14]

Clusters of patients with similar symptoms/signs [MK12] [SWS15].

Latent factors/bases for diagnoses, procedures, etc. [HGS14] [ZW14].
Electronic (or computational) phenotyping

**Rules/algorithms** that define diagnostic/inclusion criteria [PheKB].

**Classifiers** that answer the question “does patient have X?” [AL14] [AP14]

**Clusters** of patients with similar symptoms/signs [MK12] [SWS15].

Figure 1: Algorithm for identifying T2DM cases in the EMR.
Electronic (or computational) phenotyping

**Rules/algorithms** that define diagnostic/inclusion criteria [PheKB].

**Classifiers** that answer the question “does patient have X?” [AL14] [AP14]

**Clusters** of patients with similar symptoms/signs [MK12] [SWS15].

**Latent factors/bases** for diagnoses, procedures, etc. [HGS14] [ZW14].
Computational phenotyping of critical illness

Our setting: learning critical illness phenotypes from multivariate PICU time series.


Subspace clustering [BK15]
Phenotyping as representation learning

**Medicine:** *phenotypes, biomarkers* [BD01]

1. Measurable attributes of patient/disease.
2. Independent of other biomarkers.
3. Separate patients into meaningful groups.
4. Improve outcome prediction, risk assessment.
5. Clinically plausible, interpretable.

**Machine learning:** features, representations [BCV13]

1. Measurable properties of objects.
2. Independent, disentangle factors of variation.
3. Form natural clusters.
4. Useful for discriminative, predictive tasks.
5. Interpretable, provide insight into problem.
Phenotyping as representation learning

**Medicine:** *phenotypes, biomarkers* [BD01]

1. Measurable attributes of patient/disease.
2. Independent of other biomarkers.
3. Separate patients into meaningful groups.
4. Improve outcome prediction, risk assessment.
5. Clinically plausible, interpretable.

**Machine learning:** *features, representations* [BCV13]

1. Measurable properties of objects.
2. Independent, disentangle factors of variation.
3. Form natural clusters.
4. Useful for discriminative, predictive tasks.
5. Interpretable, provide insight into problem.
Deep learning of representations

Representation learning: learn transformation of data useful for some task.
Deep learning of representations

Representation learning: learn transformation of data useful for some task.
Main tool: neural networks (feedforward nets, ConvNets, RNNs, etc.)

- Date back to 40s; abandoned in 90s.
- Revived as deep learning in 2000s. (new methods, big data, faster hardware)
- State-of-the-art in vision, speech, NLP
- Google, Apple, Microsoft, Facebook
- Biologically inspired (if not plausible).
- Maximally varying, nonlinear functions.
- Exploit labeled and unlabeled data.
- Layers yield increasing abstraction.
Deep learning of representations

Representation learning: learn transformation of data useful for some task. Main tool: neural networks (feedforward nets, ConvNets, RNNs, etc.)

Output: \( \hat{y} = g(h_L W_{out} + b_{out}) \)
- sigmoid for binary classification
- softmax for multiclass classification
- identity for regression

Hidden: \( \hat{h}_\ell = h(h_{\ell-1} W_\ell + b_\ell) \)
- sigmoid or \( \tanh \) traditional
- rectified linear \((h(a) = \max(0, a))\) popular

Input: \( \hat{h}_0 = x \)
**Deep learning** of representations

**Representation learning**: learn transformation of data useful for some task.

**Main tool**: *neural networks* (feedforward nets, ConvNets, RNNs, etc.)

Train using *gradient descent*.

**Cost**: \( C(y, x; \{ W_\ell, b_\ell \}) \) (denote \( C \))

**Update**: \( W_\ell(i, j) = W_\ell(i, j) - \alpha \frac{\partial C}{\partial W_\ell(i, j)} \)

Computing the gradients via backpropagation:

\[
\frac{\partial C}{\partial W_\ell(i, j)} = \frac{\partial C}{\partial h_\ell(j)} \frac{\partial h_\ell(j)}{\partial a_\ell(j)} \frac{\partial a_\ell(j)}{\partial W_\ell(i, j)} \quad \text{where}
\]

\[
\frac{\partial h_\ell(j)}{\partial a_\ell(j)} = g'(a_\ell(j)) \quad \frac{\partial a_\ell(j)}{\partial W_\ell(i, j)} = h_{\ell-1}(i)
\]

\[
\frac{\partial C}{\partial h_\ell(j)} = \sum_k W_{\ell+1}(j, k) \frac{\partial C}{h_{\ell+1}(k)}
\]

\[
a_\ell(j) = h_{\ell-1} W_\ell(:, j) + b_j
\]
Neural nets combine different views of CP
Neural nets combine different views of CP

Output layer: classifier

---

Kale/Che (USC/VPICU)  Learning Causal Phenotypes  November 20, 2015
Neural nets combine different views of CP

Output layer: classifier

Hidden layers:
Latent factors/bases
Neural nets combine different views of CP

Output layer: classifier

Hidden layers:
Latent factors/bases

Multiclustering [BC13]
Major challenge of neural nets: interpretation

No predefined semantics
(vs. graphical model)

Learned bases not guaranteed to be uncorrelated or independent
(vs. PCA, ICA)

Information contained in distributed activations, so interpreting individual features unreliable [SZ14]
Background: why and how of computational phenotyping
Phenotypes: representations of illness
Computational phenotyping
Phenotyping as representation learning

Phenotyping clinical time series with deep learning
Deep learning for time series
Causal analysis of phenotypic representations

Experiments
Setup
Prediction results
Visualization of causal phenotypes

Conclusion

References
Deep learning for time series: window-based approach

- Apply neural net (NNet) to fixed-size windows (subsequences).
- Classification, feature extraction.
- Correlations across variables, time.
- Relatively few, weak model assumptions.
- Can learn to detect smooth, trajectory-like patterns.
Deep learning for time series: window-based approach

Can also be applied in sliding window fashion to longer time series.

\[ \hat{y} = \max_y \frac{1}{N} \sum_i P(y | X_t) \]

Full Time Series Classification

Classification

Feature extraction
Causal analysis of learned phenotypic features

- Now have set of latent factors $\{h_i\}_{i=1}^D$, response $y$.
- Analyze causal relationship between each factor, response.
- Choose causal direction of each edge: $h_i \rightarrow y$ or $h_i \leftarrow y$.
- Use only causal factors ($h_i \rightarrow y$) in further analysis.
- Note: for predictive tasks, use original network.
Causal analysis of learned phenotypic features

- Now have set of $D$ latent factors $\{h_i\}_{i=1}^D$, response $y$.
Causal analysis of learned phenotypic features

• Now have set of $D$ latent factors $\{h_i\}_{i=1}^D$, response $y$.
• Analyze of causal relationship between each factor, response.
Causal analysis of learned phenotypic features

- Now have set of $D$ latent factors $\{h_i\}_{i=1}^D$, response $y$.
- Analyze of causal relationship between each factor, response.
- Choose causal direction of each edge: $h_i \rightarrow y$ or $h_i \leftarrow y$. 
Causal analysis of learned phenotypic features

- Now have set of $D$ latent factors $\{h_i\}_{i=1}^D$, response $y$.
- Analyze the causal relationship between each factor and response.
- Choose the causal direction of each edge: $h_i \rightarrow y$ or $h_i \leftarrow y$. 

Note: for predictive tasks, use the original network.
Causal analysis of learned phenotypic features

• Now have set of $D$ latent factors $\{h_i\}_{i=1}^D$, response $y$.
• Analyze of causal relationship between each factor, response.
• Choose causal direction of each edge: $h_i \rightarrow y$ or $h_i \leftarrow y$.
• Use only causal factors ($h_i \rightarrow y$) in further analysis.
Causal analysis of learned phenotypic features

- Now have set of $D$ latent factors $\{h_i\}_{i=1}^{D}$, response $y$.
- Analyze of causal relationship between each factor, response.
- Choose causal direction of each edge: $h_i \rightarrow y$ or $h_i \leftarrow y$.
- Use only causal factors ($h_i \rightarrow y$) in further analysis.
- **Note:** for predictive tasks, use original network.
Causal analysis with pairwise likelihood ratios [HS13]

For two variables $h$ and $y$, want to distinguish between two causal models:

$h \rightarrow y : y = \rho h + d$

$h \leftarrow y : h = \rho y + e$

$h, y$ are non-Gaussian. Noise $d (e)$ is independent of $x (y)$.

Model log-likelihood:

$$\log L(h \rightarrow y) = \log p_h(h) + \log p_d \left( \frac{y - \rho h}{\sqrt{1 - \rho^2}} \right) - \log(1 - \rho^2).$$

Sign of likelihood ratio determines direction of causal edge:

$$R = \log L(h \rightarrow y) - \log L(h \leftarrow y)$$

$$\begin{cases} R > 0 & \text{if } h \rightarrow y \\ R < 0 & \text{if } h \leftarrow y \end{cases}$$

**Important note:** makes no statement about strength of edge. Use in combination with feature selection!
Outline

1 Background: why and how of computational phenotyping
   Phenotypes: representations of illness
   Computational phenotyping
   Phenotyping as representation learning

2 Phenotyping clinical time series with deep learning
   Deep learning for time series
   Causal analysis of phenotypic representations

3 Experiments
   Setup
   Prediction results
   Visualization of causal phenotypes

4 Conclusion

5 References
Clinical data sets

8500 multivariate time series from CHLA PICU (PICU) [7]:

- All > 24 hours long.
- Sampled once per hour (after preprocessing*).
- 13 variables: vitals, labs, outputs, assessments.
- Phenotype labels: 67 groups of ICD-9 codes, 19 standard ICD-9 categories.

* Age correction (PICU only), resampling, imputation, rescaling, etc.
Clinical data sets

8500 multivariate time series from CHLA PICU (PICU) [7]:

- All > 24 hours long.
- Sampled once per hour \((after\ preprocessing^\ast\)).
- 13 variables: vitals, labs, outputs, assessments.
- Phenotype labels: 67 groups of ICD-9 codes, 19 standard ICD-9 categories.

8000 multivariate time series from PhysioNet Challenge 2012\(^\dagger\) (PC2012):

- 48 hours long (not full episodes in all cases).
- Sampled once per hour \((after\ preprocessing^\ast\)).
- 33 variables: vitals, labs, outputs, assessments.
- Label: in-hospital mortality

\(^\ast\) Age correction (PICU only), resampling, imputation, rescaling, etc.
\(^\dagger\) http://physionet.org/challenge/2012/
General experimental setup

1 Data preparation
   • Generate 5-10 random training/validation/test splits of episodes.
   • Train on fixed-size windows of time series:
     • PC2012: full 48 hour time series.
     • PICU: 12 hour windows extracted in sliding window fashion.

2 Model architecture, training details
   • 3 hidden layers, fully connected, sigmoid activation.
   • Unsupervised pretraining with stochastic denoising autoencoders.
   • Supervised training (with early stopping) as multilayer perceptron.

3 Evaluation
   • Quantitative: area under ROC curve (AUROC), area under precision-recall curve (AUPRC), precision at 90% recall.
   • Qualitative: causal feature analysis + visualization.
First 48-hour mortality prediction (*PC2012*)

<table>
<thead>
<tr>
<th></th>
<th>AUROC</th>
<th>AUPRC</th>
<th>Prec@90%Rec</th>
</tr>
</thead>
<tbody>
<tr>
<td>Raw (R)</td>
<td>0.787 ± 0.0290</td>
<td>0.407 ± 0.0429</td>
<td>0.221 ± 0.0171</td>
</tr>
<tr>
<td>HandDesigned (H)</td>
<td>0.829 ± 0.0211</td>
<td><strong>0.468 ± 0.0479</strong></td>
<td>0.259 ± 0.0494</td>
</tr>
<tr>
<td>NNet(R,3)</td>
<td>0.821 ± 0.0210</td>
<td>0.444 ± 0.0324</td>
<td>0.256 ± 0.0303</td>
</tr>
<tr>
<td>NNet(H,3)</td>
<td><strong>0.832 ± 0.0162</strong></td>
<td>0.462 ± 0.0480</td>
<td><strong>0.271 ± 0.0260</strong></td>
</tr>
<tr>
<td>H+R</td>
<td>0.823 ± 0.0183</td>
<td>0.438 ± 0.0354</td>
<td>0.256 ± 0.0319</td>
</tr>
<tr>
<td>H+NNet(R,3)</td>
<td><strong>0.845 ± 0.0165</strong></td>
<td><strong>0.487 ± 0.0473</strong></td>
<td><strong>0.291 ± 0.0335</strong></td>
</tr>
</tbody>
</table>

Mean performance with standard deviation (10 folds); classifier: linear SVM + $L_1$ penalty.

**NeuralNet:** Layer 3 hidden unit activations of neural net  
(3 layer neural net, unsupervised + supervised training)

**HandDesigned:** extremes, central tendencies, variance, trends

PICU classification results: Che, Kale, Li, Bahadori, and Liu, SIGKDD 2015 [*CK15*]
Phenotype for *septic shock* (ICD-9: 990-995)

- Very irregular physiology, known symptoms of sepsis.
- Low Glasgow coma score indicates patient is unconscious.
Phenotype for circulatory disease (ICD-9: 390-459)

- Elevated blood pressure and heart rate, depressed pH.
- Evidence of ventilation (elevated FIO2).
- Note elevated urine output; also correlated with urinary disorders.
Outline

1. Background: why and how of computational phenotyping
   - Phenotypes: representations of illness
   - Computational phenotyping
   - Phenotyping as representation learning

2. Phenotyping clinical time series with deep learning
   - Deep learning for time series
   - Causal analysis of phenotypic representations

3. Experiments
   - Setup
   - Prediction results
   - Visualization of causal phenotypes

4. Conclusion

5. References
Conclusion

We have presented

- a conceptual framework for discovery of causal phenotypic representations.
- empirical results showing it can discover relevant phenotypes.


Future work:
- Think deeply about what we mean by causality in this setting.
- Further empirical investigation of learned representations.
- Combine causal analysis, representation learning. See [CP15] for example.
- Take into account temporality, treatment effects.

Dave Kale: http://www-scf.usc.edu/~dkale/
Zhengping Che: http://www-scf.usc.edu/~zche/
Yan Liu: http://www-bcf.usc.edu/~liu32/

Thank you and fight on!

Kale/Che (USC/VPICU)
Conclusion

We have presented

- a conceptual framework for discovery of causal phenotypic representations.
- empirical results showing it can discover relevant phenotypes.

Conclusion

We have presented

- a conceptual framework for discovery of causal phenotypic representations.
- empirical results showing it can discover relevant phenotypes.


Future work:

- Think deeply about what we mean by *causality* in this setting.
- Further empirical investigation of learned representations.
Conclusion

We have presented

- a conceptual framework for discovery of causal phenotypic representations.
- empirical results showing it can discover relevant phenotypes.


Future work:

- Think deeply about what we mean by *causality* in this setting.
- Further empirical investigation of learned representations.
- Combine causal analysis, representation learning. See [CP15] for example.
Conclusion

We have presented

- a conceptual framework for discovery of causal phenotypic representations.
- empirical results showing it can discover relevant phenotypes.


Future work:

- Think deeply about what we mean by *causality* in this setting.
- Further empirical investigation of learned representations.
- Combine causal analysis, representation learning. See [CP15] for example.
- Take into account temporality, treatment effects.
Conclusion

We have presented

- a conceptual framework for discovery of causal phenotypic representations.
- empirical results showing it can discover relevant phenotypes.


Future work:

- Think deeply about what we mean by causality in this setting.
- Further empirical investigation of learned representations.
- Combine causal analysis, representation learning. See [CP15] for example.
- Take into account temporality, treatment effects.

Dave Kale: http://www-scf.usc.edu/~dkale/
Zhengping Che: http://www-scf.usc.edu/~zche/
Yan Liu: http://www-bcf.usc.edu/~liu32/

Thank you and fight on!
Outline

1 Background: why and how of computational phenotyping
   Phenotypes: representations of illness
   Computational phenotyping
   Phenotyping as representation learning

2 Phenotyping clinical time series with deep learning
   Deep learning for time series
   Causal analysis of phenotypic representations

3 Experiments
   Setup
   Prediction results
   Visualization of causal phenotypes

4 Conclusion

5 References


[PheKB] Phenotyping KnowledgeBase project: https://phekb.org/


[PheKB] TODO.


