ROYAL INSTITUTE OF TECHNOLOGY

Aalto University
School of Science and Technology

Where we left off last time:

Variational inference

Martin Wainwright & Michael Jordan "Graphical Models, Exponential Families, and Variational Inference", *Foundations and Trends In Machine Learning* **1**: 1–305 (2008)

Particularly mean-field inference. In neuroscience called "naïve mean-field".

Mean-field approximation (statistics, Boolean version)

$$P(\mathbf{S}) = \frac{1}{Z(\mathbf{h},\mathbf{J})} \exp\left(\sum_i h_i S_i + \sum_{ij} J_{ij} S_i S_j \right)$$

$$m_i = \sum_{\mathbf{S}} P(\mathbf{S}) S_i$$

$$P^{\text{trial}}(\mathbf{S}) = \Pi_i \frac{\exp\left(h_i^{MF} S_i\right)}{2\cosh\left(h_i^{MF}\right)} = \Pi_i \tfrac{1}{2}\left(1 + m_i^{MF} S_i\right)$$

$$m_i^{MF} = \sum_{\mathbf{S}} P^{\text{trial}}(\mathbf{S}) S_i = \tanh h_i^{MF}$$

Minimize Kullback-Leibler divergence (relative entropy)

$$h_i^{MF} = \arg\min D_{KL}\left(P \,\middle\|\, P^{\text{trial}}\right) = \arg\min \sum_{\mathbf{S}} P(\mathbf{S}) \log \frac{P(\mathbf{S})}{P^{\text{trial}}(\mathbf{S})} \quad \Rightarrow \quad m_i^{MF} = m_i$$

This is what "mean-field" means in statistics. For estimation one can adjust the parameters (m_i^{MF}) of the trial distribution to match the sample averages (m_i). But we will also need m_i in terms of $(\mathbf{h}, \mathbf{J})$.

Variational formulation (1/2)

"maximum-entropy" (actually, "minimum free energy")

$$P(\mathbf{S}) = \frac{1}{Z(\mathbf{h},\mathbf{J})} \exp\left(\sum_i h_i S_i + \sum_{ij} J_{ij} S_i S_j \right) = \arg\min \left[\sum_S P(S) \left(\log P(S) - \sum_i h_i S_i - \sum_{ij} J_{ij} S_i S_j \right) \right]$$

Variation is over all probability distributions.

Proof is by variation using a Lagrange parameter for the normalization:

$$\frac{\delta}{\delta P(\boldsymbol{\sigma})} \left[\sum_{\mathbf{S}} P(\mathbf{S}) \left(\log P(\mathbf{S}) - \sum_i h_i S_i - \sum_{ij} J_{ij} S_i S_j \right) + \lambda \left(\sum_{\mathbf{S}} P(\mathbf{S}) - 1 \right) \right] = \log P(\boldsymbol{\sigma}) + 1 - \sum_i h_i \sigma_i - \sum_{ij} J_{ij} \sigma_i \sigma_j + \lambda$$

$$\frac{\delta}{\delta P(\boldsymbol{\sigma}) \delta P(\boldsymbol{\tau})} \left[\sum_{\mathbf{S}} P(\mathbf{S}) \left(\log P(\mathbf{S}) - \sum_i h_i S_i - \sum_{ij} J_{ij} S_i S_j \right) + \lambda \left(\sum_{\mathbf{S}} P(\mathbf{S}) - 1 \right) \right]_{\text{vanishing first variation}} = \frac{1}{P(\boldsymbol{\sigma})} \mathbf{1}_{\boldsymbol{\sigma},\boldsymbol{\tau}}$$

Related to reverse Kullback-Leibler

$$\sum_S P^{\text{trial}}(S) \left(\log P^{\text{trial}}(S) - \sum_i h_i S_i - \sum_{ij} J_{ij} S_i S_j \right) = -\log Z + D_{KL}\left(P^{\text{trial}} \| P \right)$$

Variational formulation (2/2)

In statistics and neuroscience this is "naïve mean-field"

$$P^{\mathrm{nMF}}(\mathbf{S}) = \prod_i \frac{\exp\left(h_i^{nMF} S_i\right)}{2\cosh\left(h_i^{nMF}\right)} = \arg\min\left[\sum_S P(S)\left(\log P(S) - \sum_i h_i S_i - \sum_{ij} J_{ij} S_i S_j\right)\right]$$

Variation is now only over all factorized distributions.

$$m_i^{nMF} = \tanh h_i^{nMF}$$

$$P^{\mathrm{trial}}(\mathbf{S}) = \Pi_i \tfrac{1}{2}\left(1 + m_i^{nMF} S_i\right)$$

$$H\left(\frac{1+m_i^{nMF}}{2}\right) = -\frac{1+m_i^{nMF}}{2}\log\frac{1+m_i^{nMF}}{2} - \frac{1-m_i^{nMF}}{2}\log\frac{1-m_i^{nMF}}{2}$$

$$F^{nMF}\left(\mathbf{m}^{nMF}\right) = \sum_i H\left(\frac{1+m_i^{nMF}}{2}\right) + h_i m_i^{nMF} + \sum_{ij} J_{ij} m_i^{nMF} m_j^{nMF}$$

$$\frac{\partial F^{nMF}\left(\mathbf{m}^{nMF}\right)}{\partial m_i^{nMF}} = 0 \Leftrightarrow \tanh^{-1} m_i^{nMF} = h_i + \sum_j J_{ij} m_j^{nMF}$$

Forward problem: use knowledge of **(h,J)** to predict $\mathbf{m}^{\mathrm{nMF}}$. Generally $\mathbf{m}^{nMF} \neq \mathbf{m}$. This is "mean-field" in Physics.

Inverse problem: use observed **m** to infer **(h,J)**, assuming $\mathbf{m} \approx \mathbf{m}^{nMF}$. So far under-determined.

Naïve mean-field inference

Combine naïve mean-field with a simple exact result for this family of probability distributions:

$$P(\mathbf{S}) = \frac{1}{Z(\mathbf{h},\mathbf{J})} \exp\left(\sum_i h_i S_i + \sum_{ij} J_{ij} S_i S_j \right)$$

$$m_i = \sum_{\mathbf{S}} P(\mathbf{S}) S_i = \partial_{h_i} \log Z(\mathbf{h},\mathbf{J})$$

$$c_{ij} = \sum_{\mathbf{S}} P(\mathbf{S}) S_i S_j - m_i m_j$$

$$\text{(response)} \quad \chi_{ij} = \frac{\partial m_i}{\partial h_j} = c_{ij} \quad \text{(correlation)}$$

$$\chi^{nMF}_{ij} = \frac{\partial m^{nMF}_i}{\partial h_j} \approx \frac{\partial m_i}{\partial h_j} = c_{ij}$$

$$\left(\chi^{nMF}\right)^{-1}_{ij} = \frac{\partial h_i}{\partial m^{nMF}_j} \approx \left(c^{-1}\right)_{ij}$$

$$h_i = \tanh^{-1} m^{nMF}_i - \sum_j J_{ij} m^{nMF}_i \Rightarrow \left(c^{-1}\right)_{ij} \approx \frac{1}{1-m_i^2} 1_{ij} - J_{ij} \Rightarrow J^{nMF}_{ij} = -\left(c^{-1}\right)_{ij}$$

To estimate pair-wise interactions in mean-field DCA one should hence score by the *inverse* correlation matrix!!!

Plefka expansion

An alternative route to mean-field approximations is to consider the **J** small, and then adjust **h** to match **m**.

$$P(\mathbf{S}) = \frac{1}{Z^0(\mathbf{h}^0) + \varepsilon Z^1(\mathbf{h}^0, \mathbf{h}^1, \mathbf{J}) + \cdots} \exp\left(\sum_i h_i^0 S_i + \varepsilon h_i^1 S_i + \cdots + \varepsilon \sum_{ij} J_{ij} S_i S_j \right)$$

$$\tanh h_i^0 = m_i^{MF} = m_i$$

$$\sum_{\mathbf{S}} P(\mathbf{S}) = 1 \Rightarrow Z^1(\mathbf{h}^0, \mathbf{h}^1, \mathbf{J}) = Z^0(\mathbf{h}^0) \left(\sum_i h_i^1 m_i + \sum_{ij} J_{ij} m_i m_j \right)$$

$$\sum_{\mathbf{S}} P(\mathbf{S})\, S_i = m_i \Rightarrow h_i^1 = -\sum_k J_{ik} m_k$$

$$\tanh\left(h_i^0 + \varepsilon h_i^1 + \varepsilon \sum_k J_{ik} m_k \right) = m_i + O(\varepsilon)$$

Expanding h_i to first order in ε hence gives the same relation as physical mean-field. Naïve mean-field inference can therefore be seen as a first order in ε approximation to mean-field in statistics sense.

Second-order Plefka expansion (aka "TAP")

$$P(\mathbf{S}) = \frac{1}{Z(\mathbf{h},\mathbf{J})} \exp\left(\sum_i h_i S_i + \sum_{ij} J_{ij} S_i S_j \right)$$

$$P^{\text{trial}}(s) = \Pi_i P_i(S_i)$$

$$F^{TAP} = \sum_i H\left(\frac{1+m_2}{s}\right) + \sum_i h_i m_i + \sum_{ij} J_{ij} m_i m_j + \frac{1}{2} J_{ij}^2 \left(1-m_i^2\right)\left(1-m_j^2\right)$$

$$\frac{\partial F^{TAP}}{\partial m_i} = 0 \quad \Longrightarrow \quad m_i = \tanh\left(h_i^{nMF} + \sum_j J_{ij}\left(m_j - J_{ij}\left(1-m_j^2\right)m_i\right)\right)$$

$$\frac{\partial h_i^{TAP}}{\partial m_j} \approx \left(c^{-1}\right)_{ij} \quad \Longrightarrow \quad \left(c^{-1}\right)_{ij} \approx \left(\frac{1}{1-m_i^2} + \sum_k J_{ik}^2 \left(1-m_k^2\right)\right) 1_{ij} - \left(J_{ij} + 2 m_i m_j J_{ij}^2\right)$$

L Onsager (1930ies), Thouless-Anderson-Palmer (1970ies)
Kappen & Rodriguez, 1998, Kappen & Spanjers, 2001, F Ricci-Tersenghi, 2013
Unfortunately does not give any improvement over naïve mean-field in DCA.

More variational methods

"Bethe-Peierls" ansatz or "Belief Propagation"

$$E(s) = \sum_i h_i S_i + \sum_{ij} J_{ij} S_i S_j \quad P^{\text{trial}}(s) = \Pi_i P_i(S_i) \Pi_{ij} \frac{P_{ij}(S_i, S_j)}{P_i(S_i) P_j(S_j)}$$

$$F^{BA} = \sum_{ij} S_{ij}(m_i, m_j, c_{ij}) - (d_i - 1) \sum_i S_i + \sum_i h_i m_i + \sum_{ij} J_{ij}(m_i m_j + c_{ij})$$

$$S(m_i, m_j, c_{ij}) = H\left(\frac{(1-m_i)(1-m_j)+c_{ij}}{4}\right) + H\left(\frac{(1+m_i)(1+m_j)+c_{ij}}{4}\right) + H\left(\frac{(1-m_i)(1+m_j)-c_{ij}}{4}\right) + H\left(\frac{(1+m_i)(1-m_j)-c_{ij}}{4}\right)$$

$$\frac{\partial F^{BA}}{\partial c_{ij}} = 0$$

A somewhat complicated expression involving m_i and m_j and t_{ij}=tanh J_{ij}, *see* F Ricci-Tersenghi, 2012

$$m_i = \tanh\left(h_i^{BA} + \sum_j \tanh^{-1}\left(t_{ij} f(m_i, m_j, t_{ij})\right)\right)$$

…with an auxiliary function f…

$$\frac{\partial h_i^{BA}}{\partial m_j} \approx \left(c^{-1}\right)_{ij}$$

A rather complicated expression. This theoretically beautiful method unfortunately seems to do worse than the mean-field in DCA (nobody really knows why).

2nd main method: pseudo-likelihood maximization

Maximum likelihood $P(\mathbf{S}) = \frac{1}{Z(\mathbf{h},\mathbf{J})} \exp\left(\sum_i h_i S_i + \sum_{ij} J_{ij} S_i S_j \right)$

$$\Pr(\mathbf{S}^{(1)},\ldots,\mathbf{S}^{(n)};\mathbf{h},\mathbf{J}) = P(\mathbf{S}^{(1)};\mathbf{h},\mathbf{J}) \cdots P(\mathbf{S}^{(n)};\mathbf{h},\mathbf{J})$$

$$\mathbf{h}^*, \mathbf{J}^* \in \arg\max \left[\sum_{ij} h_i \frac{1}{n} \sum_{s=1}^{n} S_i^{(s)} + \sum_{ij} J_{ij} \frac{1}{n} \sum_{s=1}^{n} S_i^{(s)} S_j^{(s)} - \log Z(\mathbf{h},\mathbf{J}) \right]$$

Pseudo-maximum likelihood (avoids computing Z):

$$P(S_r \mid S_{\setminus r}) = \exp\left(h_r S_r + \sum_l J_{rl} S_r S_l \right) \Big/ \sum_y \exp\left(h_r y + \sum_l J_{rl} y S_l \right)$$

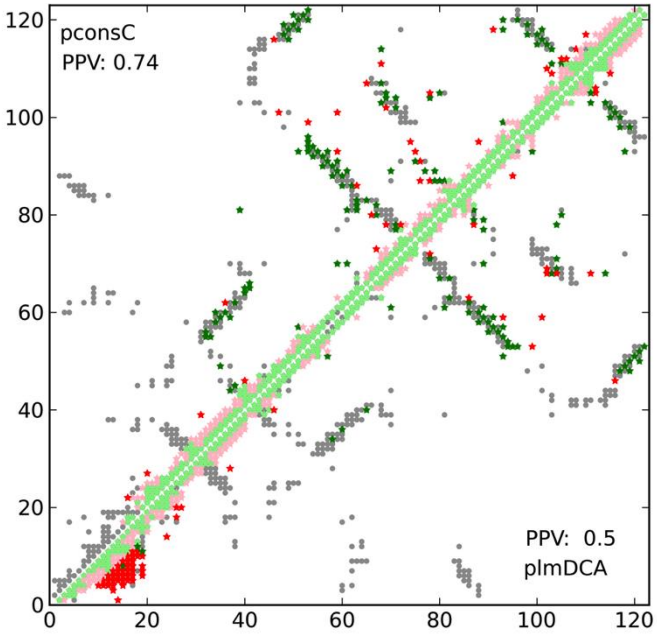
$$h_r^{plm}, J_{rl}^{plm} \in \arg\max \left[\sum_{ij} h_i \frac{1}{n} \sum_{s=1}^{n} S_i^{(s)} + \sum_{ij} J_{ij} \frac{1}{n} \sum_{s=1}^{n} S_i^{(s)} S_j^{(s)} - f(h_r, J_{rl}, S_{\setminus r}) \right]$$

Besag (1974), Wainwright-Ravikumar-Lafferty (2010)

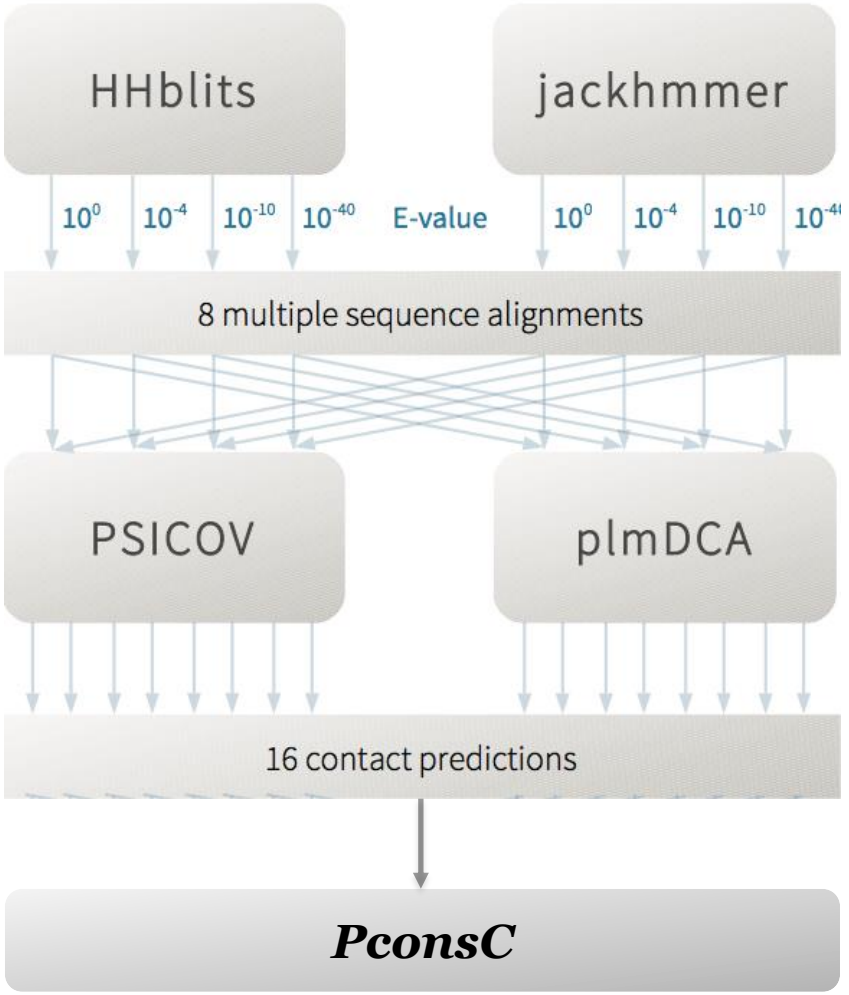
Ekeberg et al *Phys. Rev. E* (2013) + github.com/magnusekeberg/plmDCA

3rd main method: machine-learning by ”pooling” predictions...

Machine learning methods to combine different alignment sources and inference schemes

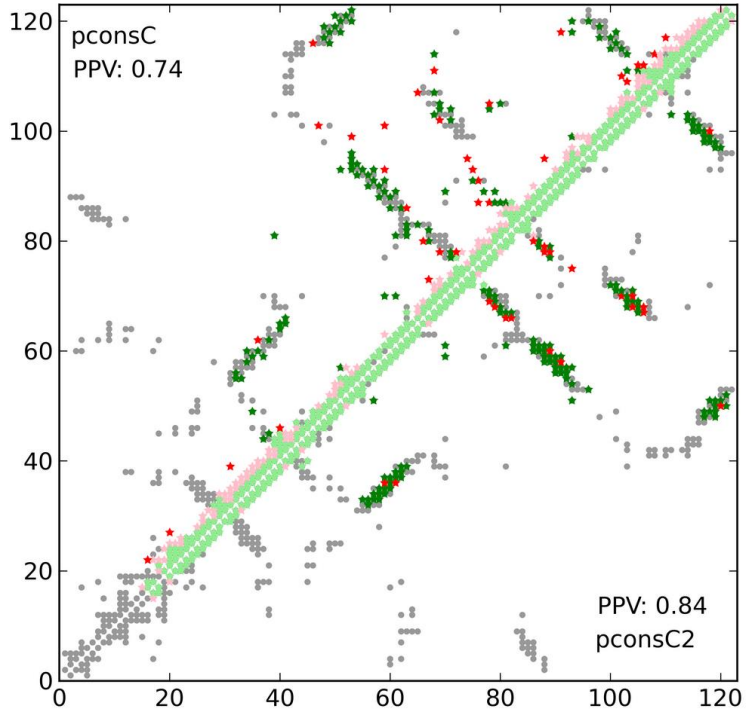


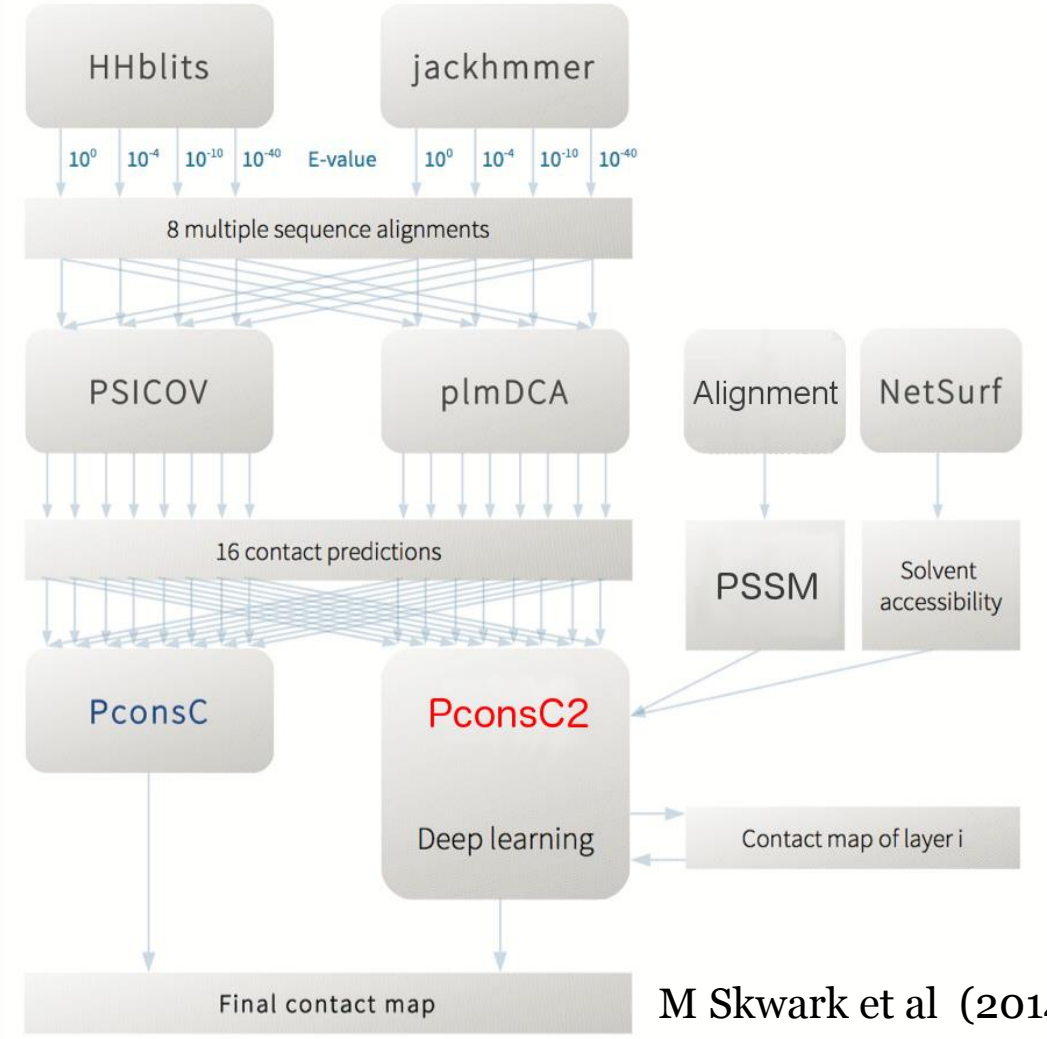
M Skwark et al, Bioinformatics (2013)



...ongoing method development...

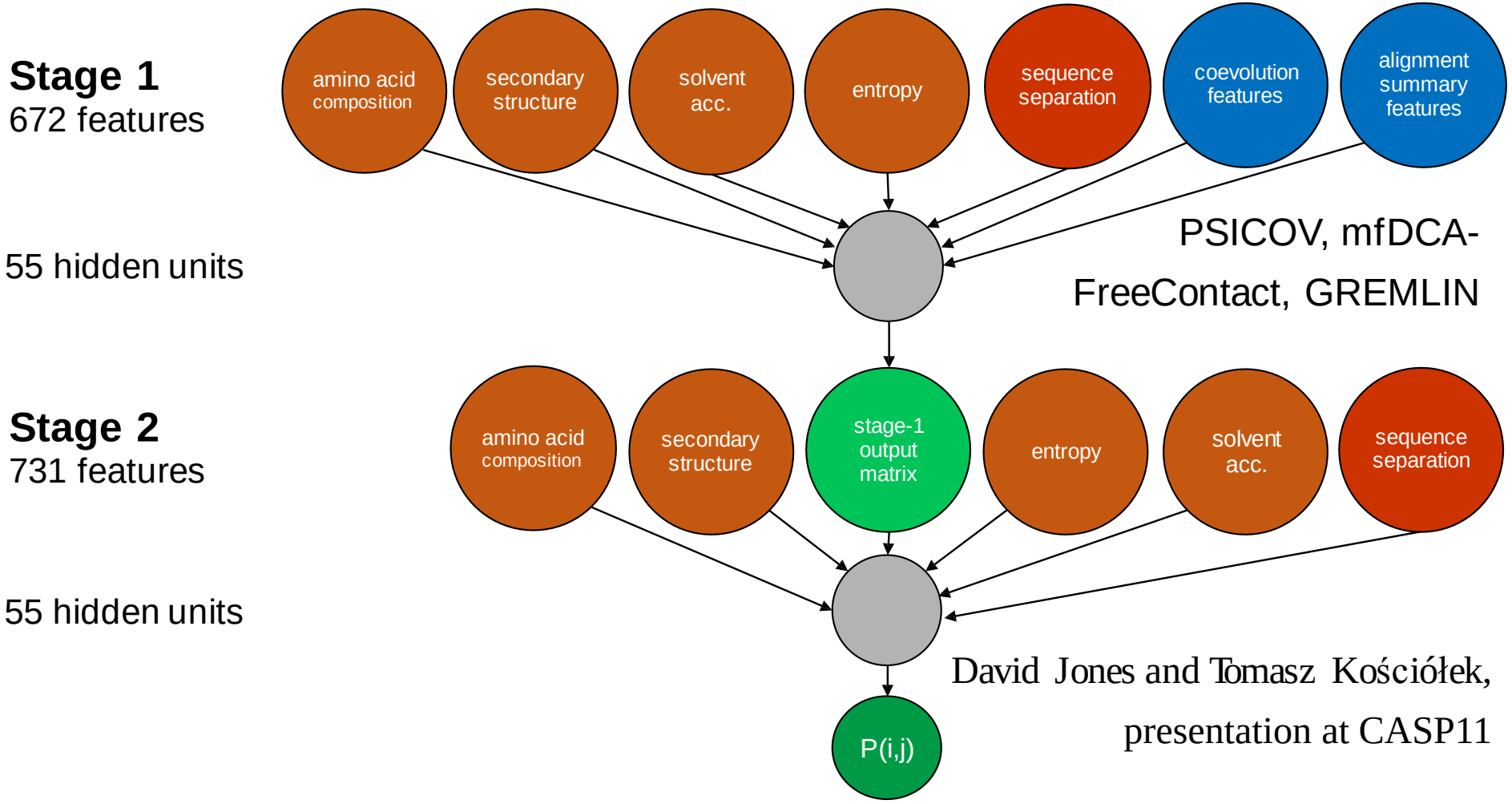
There is an ongoing effort to better pool predictions + using also domain knowledge such as solvent accessibility and protein secondary structure





M Skwark et al (2014)

...CONSIP2 / MetaPSICOV, which won contact prediction at CASP11



DCA has become a tool

For the protein structure prediction problem ”pure DCA” is no longer state of the art. The best methods at e.g. CASP combine DCA with other information by machine learning methods. The best group pursuing this in Sweden ─ and one of the best world-wide ─ is Arne Elofsson at SciLifeLab.

I’ll instead talk about another application where versions of ”pure DCA” are so far the only ones feasible.

ROYAL INSTITUTE OF TECHNOLOGY

Aalto University
School of Science and Technology

Genome-scale Direct Coupling Analysis

SF2935 Modern Methods of Statistical Learning

Guest lecture: Erik Aurell

December 12, 2017

Skwark *et al*, "Interacting networks of resistance, virulence and core machinery genes identified by genome-wide epistasis analysis" *PLoS Genetics* 2017
Puranen *et al*, " SuperDCA for genome-wide epistasis analysis" [biorxiv:182527]
高辰毅 Chen-Yi Gao, 周海军 Hai-Jun Zhou, E.A. [arXiv:1710.04819]
Latest code at: github.com/gaochenyi/CC-PLM

DCA as epistasis analysis

DCA as a hypothesis in evolutionary biology: epistasis within one gene (protein) is reflected by the co-variation of allele frequencies as described by a models in an exponential family (Potts model).

Translation: epistasis means non-additive effects of gene variants on fitness.

> *"One of the best systems for rigorously testing the functional and evolutionary consequences of epistasis is in the within-locus interactions that characterize protein folding and activity"*
>
> P. Phillips, *Nat Rev Genet.* 2008 November ; 9(11): 855–867.

It is not clear when DCA as a hypothesis is true. Studies on DCA for protein structures prediction however give (strong) empirical support.

Suitable whole-genome bacterial sequencing data

The Maela Acute Respiratory Infections study

BBC News 2007 9/17

….included a large-scale whole-genome sequencing project to better understand the genetic characteristics of colonizing strains of *Streptococcus pneumoniae*

…about 3,000 pneumococcal carriage isolates were sequenced

…after various filtrations about 100,000 loci of variability remain.

…data used to learn a Potts model in Skwark *et al* (2017) and in Puranen *et al* (2017), and by Chen-Yi Gao's PLM-DCA.

DCA and inference methods

Maximum likelihood and Bayesian point estimates are clearly unfeasible for these data sizes (even for the protein problems)

Variational inference would be possible, but has not yet been tried on the whole-genome bacterial data.

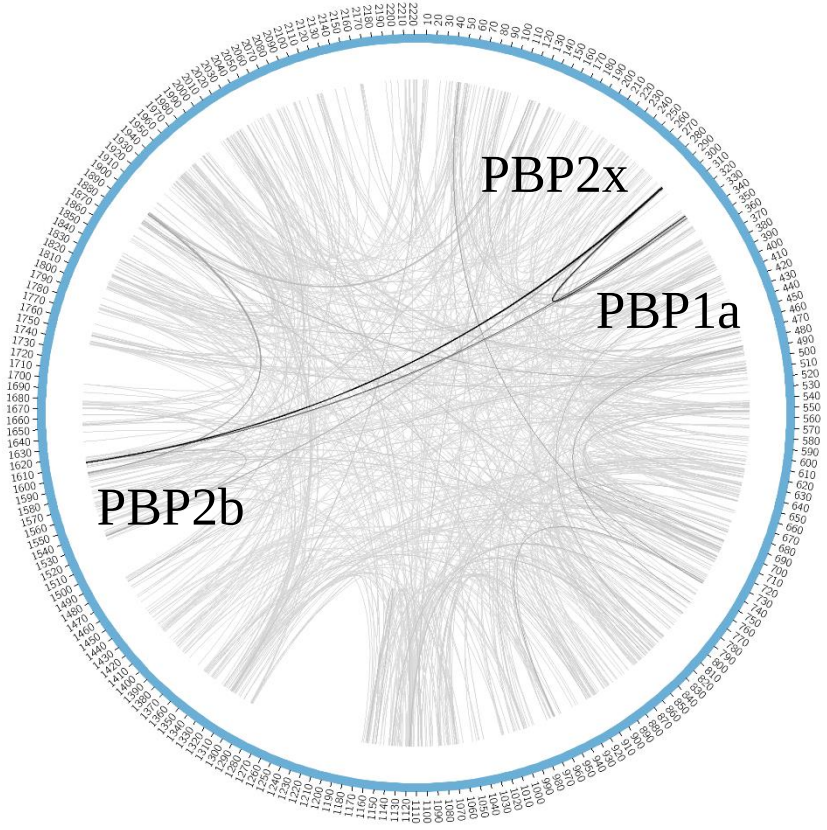
Pseudo-likelihood maximization (PLM) is based on maximizing *conditional probability* of a model in an exponential family:

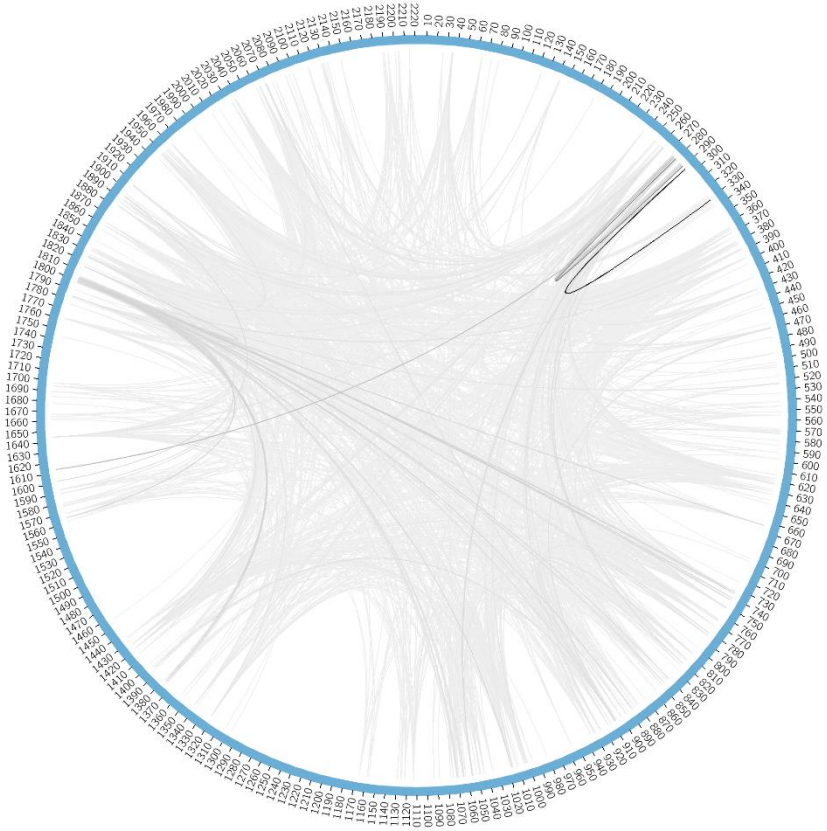
$$P(x_r \mid x_{\setminus r}) = \frac{1}{z(x_{\setminus r}, h_r, J_{rl})} \exp\left(h_r(x_r) + \sum_{l \in \setminus r} J_{rl}(x_r, x_l)\right)$$

$$h_r^*, J_{rl}^* \in \arg\max \frac{1}{n} \sum_{s=1}^{n} \left(h_r(x_r^{(s)}) + \sum_{l \in \setminus r} J_{rl}(x_r^{(s)}, x_l^{(s)}) - \log z(x_{\setminus r}^{(s)})\right) - \lambda_h |h|_2 - \lambda_J |J|_2$$

Besag (1975); Ravikumar, Wainwright & Lafferty (2011); Ekeberg et al (2013) (plmDCA); Kamisetty et al 2013 (GREMLIN); Nguyen, Berg & Zecchina (2017)

DCA vs correlations on Maela

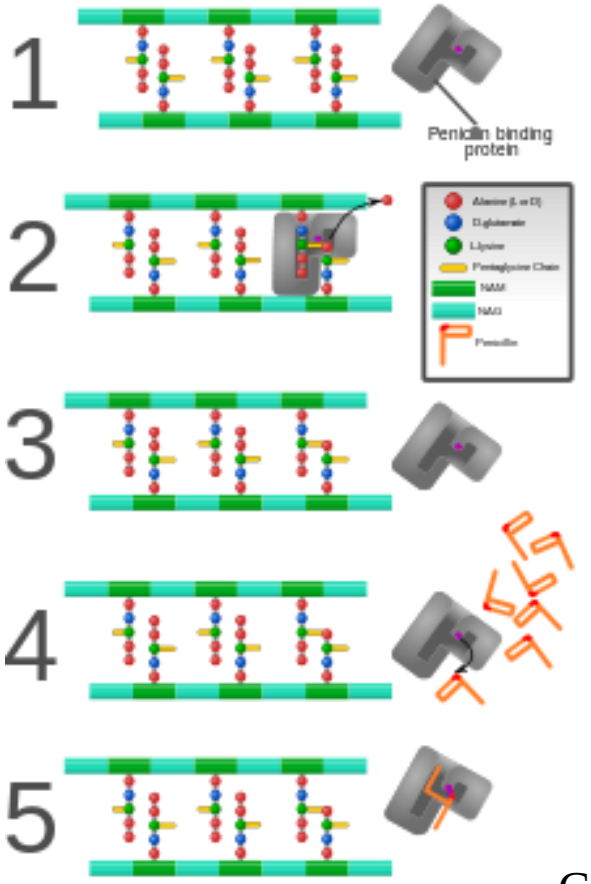




Epistatic interactions between three genes in the PBP family. Skwark *et al*, *PLoS Genetics* 2017 (new visualization)

5199 locus-locus correlation matrices scored by mutal information. Finds presumably some indirect couplings.

β-lactam (penicillin) resistance



@Mcstrother
Wikimedia Commons

PBPs (Penicillin-binding proteins)

B. Spratt, *Eur. J. Biochem.* (1977)

PASTA (PF03793)

Penicillin-binding protein **a**nd **s**erine/**t**hreonine kinase **a**ssociated domain [..] binds beta-lactam antibiotics and their peptidoglycan analogues […] describe this previously uncharacterized domain and infer that it binds beta-lactam antibiotics and their peptidoglycan analogues.

C. Yeats, RD Finn, A. Bateman, *Trends Biochem Sci.* (2002)
"The PASTA domain: a beta-lactam-binding domain".

PLM is quite compute-intensive on genome scale

Theoretical scaling is L^3 with a large pre-factor; L for the number of loci and L^2 for the maximization (conjugate gradient).

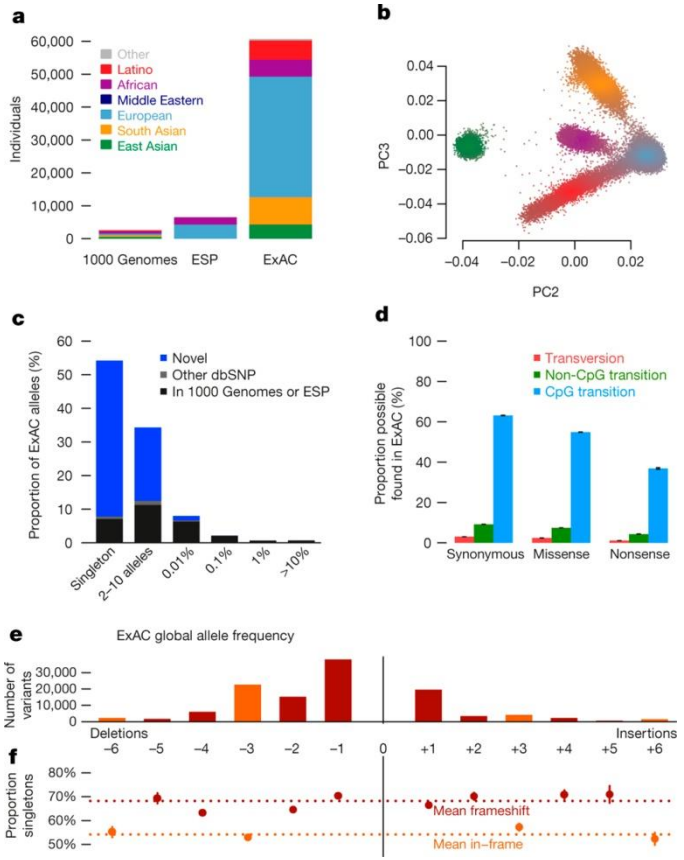
Magnus Ekeberg's plmDCA does not run out of the box on data size L=10^5 — on the computer resources we have had access to.

The results published in Skwark et al *PLoS Genetics* (2017) were based on sampling and running Ekeberg's plmDCA on a smaller data sample; it took on the order of 500,000 core-hours.

A recent re-implementation by Chen-Yi Gao takes about 18,000 core-hours (PLM-DCA, github.com/gaochenyi/CC-PLM).

The more advanced optimization by Santeri Puranen took about 3,300 core-hours (SuperDCA, biorxiv:182527).

DCA human genome scale?



The Exome Aggregation Consortium (ExAC) last year published a catalogue of human genetic diversity containing an average of one variant every eight bases of the exome (protein-coding regions), or 7,404,909 loci of variability (for some filtering criteria).

M Lek *et al.* "Analysis of protein-coding genetic variation in 60,706 humans" *Nature* **536,** 285–291 (2016)

DCA on human *exome*: 10^7 loci

DCA on human *genome*: 10^9 loci

Clearly DCA on the *human exome* is, in practice, not possible. It would also be of interest to have a DCA method with faster turn-around time for *bacterial genomes*.

ROYAL INSTITUTE OF TECHNOLOGY

Aalto University
School of Science and Technology

CC-DCA

Correlation-Compressed Direct Coupling Analysis

The idea: to look for strong DCA couplings only among pairs of loci that are strongly correlated. That is, to take "Direct" in the acronym DCA seriously.

$$|C_{ij}| > |C_{kl}| > |C_{np}| > \cdots$$

Gao, Zhou, Aurell (2017)
arXiv:1710.04819, Fig 1

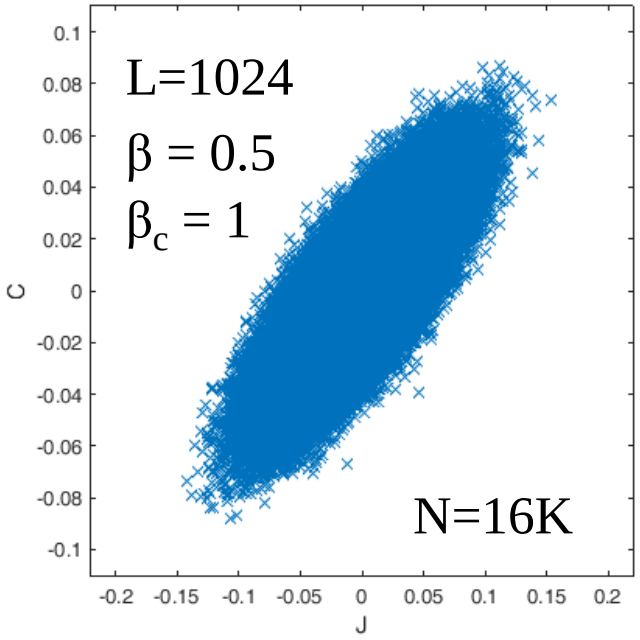
The computational bottle-neck moves to the easier and much more standard task of computing correlations.

$\tilde{\sigma}^{(1)} =$ GTTCAG
$\tilde{\sigma}^{(2)} =$ TGTAGC
$\tilde{\sigma}^{(N)} =$ GTGCTG

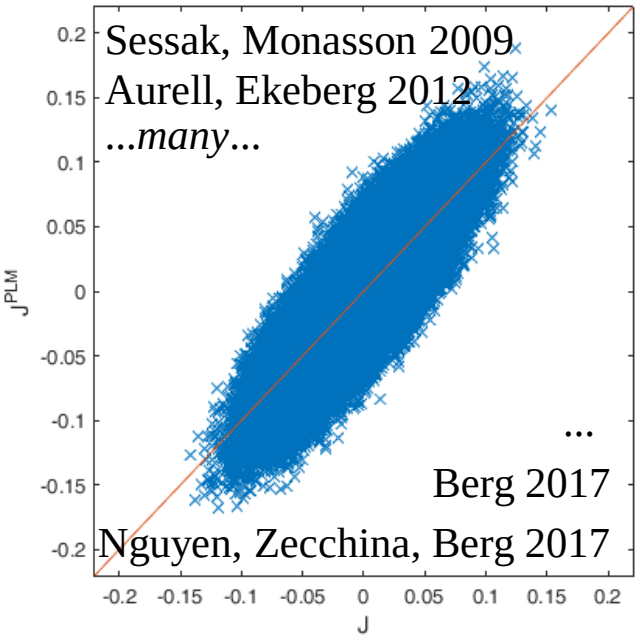
Naïvely: NL^2 (what we used)
P. Kaski: $L^{1.62}$ (very large L)

Does CC-DCA work? That depends on the data…

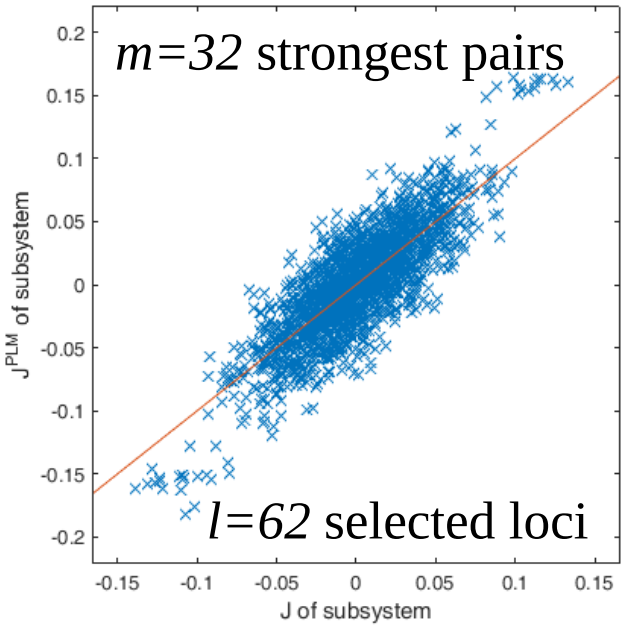
A well-studied model is the Sherrington-Kirkpatrick spin glass. All couplings are then *i.i.d.* Gaussian random variable distributed as $N(0,\beta/L)$ where β is a parameter. When $\beta \geq 1$ it is hard to sample from this model, while when $\beta \ll 1$ the signal is weak, many samples are needed. CC-DCA does reasonably compared to PLM.



Correlations vs true *J*'s



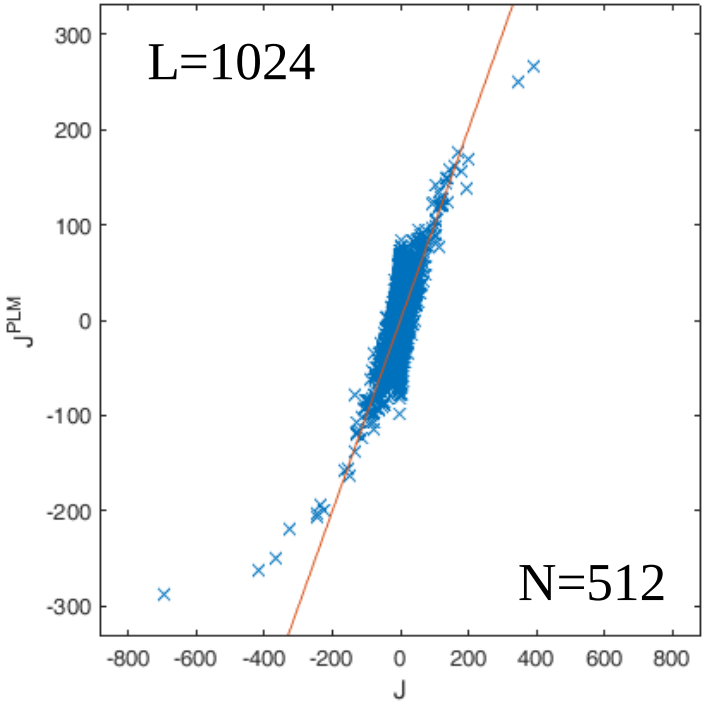
Inferred vs true *J*'s (plmDCA)



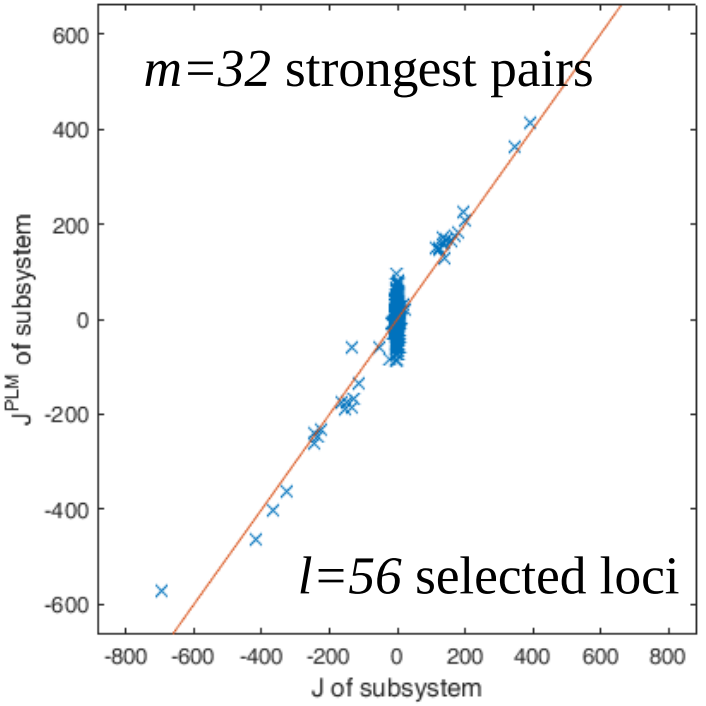
Same for CC-DCA

A more realistic scenario: Random Power-Law Model

In spin-glass RPLM all couplings are of random sign, but their sizes i.i.d. random variables with distribution $P(x) \sim x^{-\alpha}$. I here consider $\alpha = 3$. Some couplings will then be much larger than the others (mean is finite but variance unbounded).



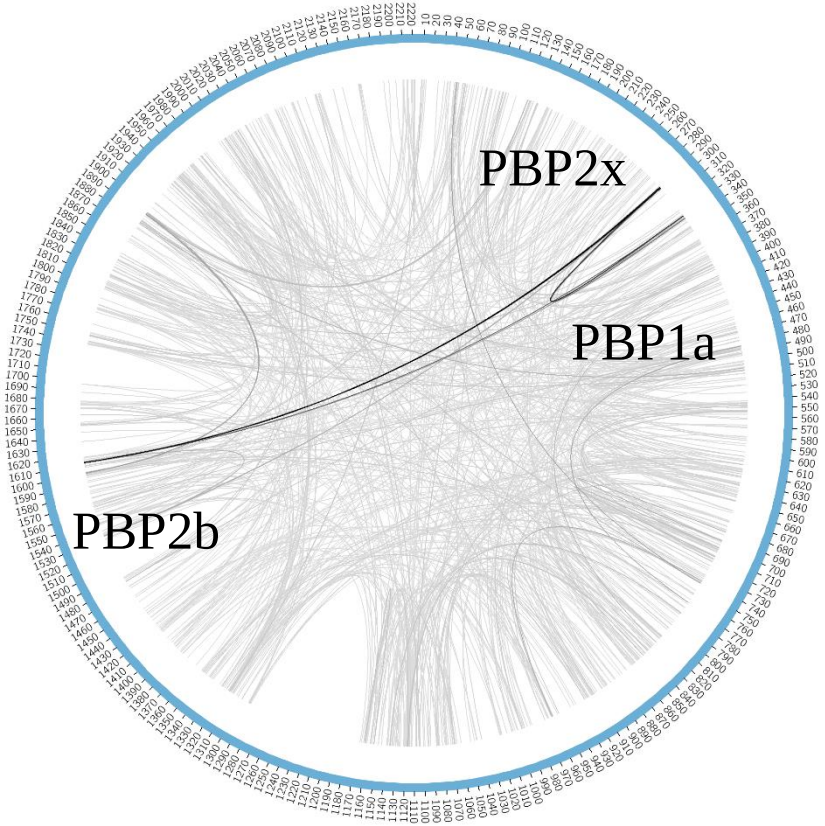
Inferred vs true J's (plmDCA)



Inferred vs true J's (CC-DCA)

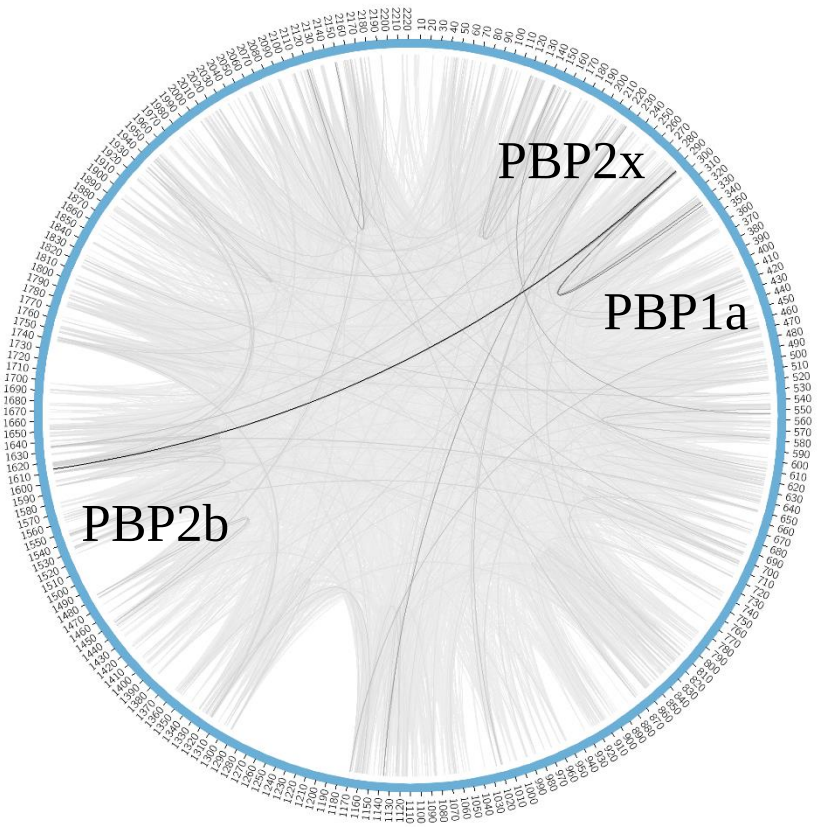
DCA vs CC-DCA on Maela

≈ 500,000 core·h



The 5,199 strong couplings identified in Skwark *et al*, *PLoS Genetics* 2017 (new visualization)

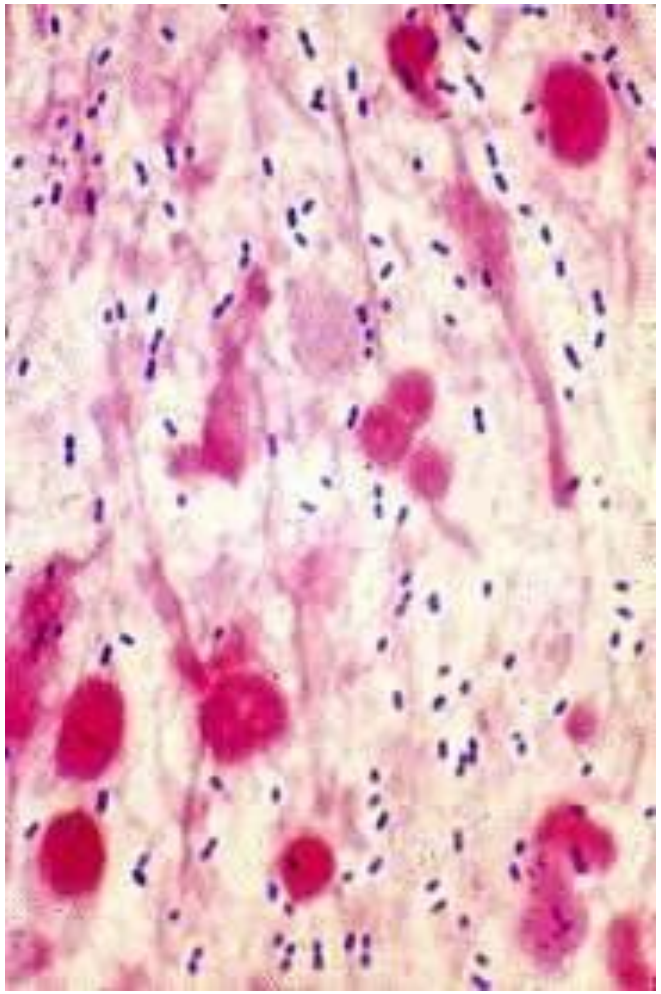
≈ 15.8 core·h



All 15,717 long-range couplings between l=9,300 loci that appear in 30,000 strongest correlated pairs.

Conclusions

DCA and CC-DCA on bacterial genome scale:

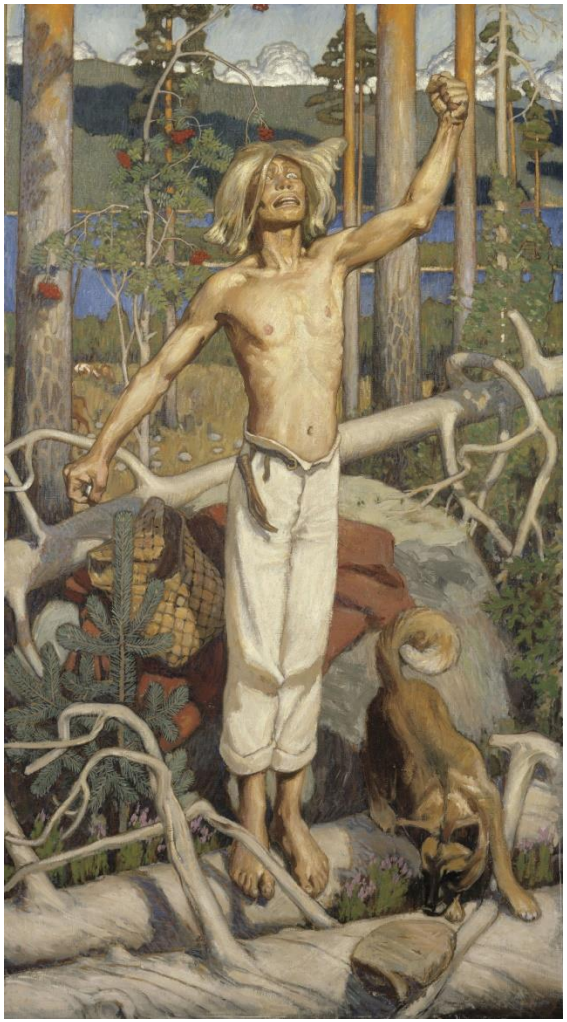


Prior state-of-the-art: DCA works on bacterial genome scale, though with a turn-around time of days to weeks.

Here: CC-DCA brings down turn-around to 16h on a single 2.2 GHz processor of less than 8GB of memory. Results are closer to DCA than to bare correlations.

Ultimate goal: to find things useful to fight pneumococcal disease. That however requires collaborating biologists and/or medical professionals, so for another time.

CC-DCA on human scale?



By the naïve algorithm it should take about $20 \cdot 9^2 \approx 1{,}600$ times longer to compute all the correlations in ExAC, compared to doing so on the Maela data set, about 25,000 core-hours. A bit long, but certainly doable.

Sub-sampling of loci or humans in the ExAC set would be a possible strategy. Fast ways to compute a set of most correlated pairs would be very interesting. NB, also by itself, without considerations of DCA or CC-DCA.

Suppose that all $3 \cdot 10^9$ positions on the human genome harbor genetic variants and all 10^{10} living humans are sequenced. These numbers could be increased further by considering also the variation of the human microbiome. For problems on that scale fast ways to compute most correlated pairs would be mandatory.

Discrete and linear traits

Traits with complicated inheritance are often assumed to depend linearly on the variants at many loci. For Boolean genetic variants and Boolean traits thus:

$$P(t \mid x) = \frac{e^{\sum_{a,i} A_{a,i} t_a x_i}}{\Pi_a (2\cosh \sum_i A_{a,i} x_i)}$$

Argued in many papers by Michael Lässig and his group

cf. N. Riedel et al "Multiple-line inference of selection on quantitative traits" *Genetics* **201,** 305-322 (2015)

The parameters of such a combined model could be learnt as in DCA. Quantitative traits are however often continuous and depend on both genetic and environmental factors. Obesity for instance depends positively on over-eating.

Karl X Gustav. Abraham Wuchters (1610-1682) [wikipedia]

Acknowledgements

高辰毅 Chen-Yi Gao

周海军 Hai-Jun Zhou

Try it out!
github.com/gaochenyi/CC-PLM

plmDCA

Magnus Ekeberg
Martin Weigt
Cecilia Lövkvist
Yueheng Lan
Tuomo Hartonen
Andrea Pagnani
Christoph Feinauer
Marcin Skwark

genomeDCA

Marcin Skwark
Jukka Corander
Santeri Puranen
Maiju Pesonen
Yingying Xu
Maela collaboration