The 10th Annual MLSP Competition: First Place

— Schizophrenia Classification Challenge —

Arno Solin Simo Särkkä
Department of Biomedical Engineering and Computational Science
Aalto University, Finland

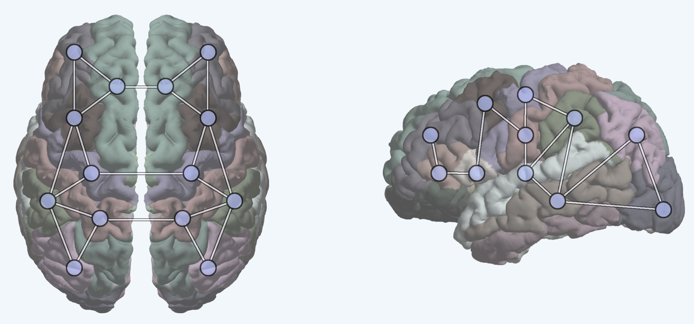
Introduction

- Schizophrenia is a chronic mental disorder that is often characterized by abnormal social behavior and odd interpretations of reality.
- Associated with small differences in brain structure and activity.
- Therefore brain imaging techniques provide data that can be used for assisting the diagnosis of schizophrenia.
- The goal was to automatically diagnose subjects with schizophrenia using multimodal features derived from their magnetic resonance imaging (MRI) brain scans.
- The winning proposition [1] was based on a Gaussian process (GP) classifier [2].

Materials

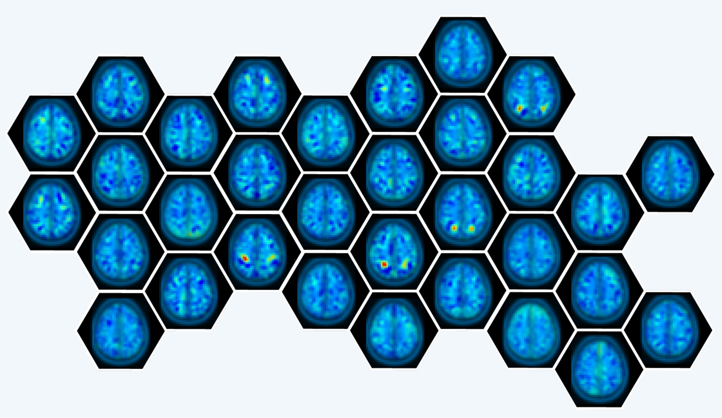
- The data consist of two sets of information collected by different medical imaging modalities: functional and structural data.

Functional data



- *Functional Network Connectivity* (FNC, [4]) derived from functional magnetic resonance imaging (fMRI) scans.
- Describes the subject's overall level of 'synchronicity' between brain areas.
- Synchrony measures between each of the 28 chosen areas (378 combinations altogether).

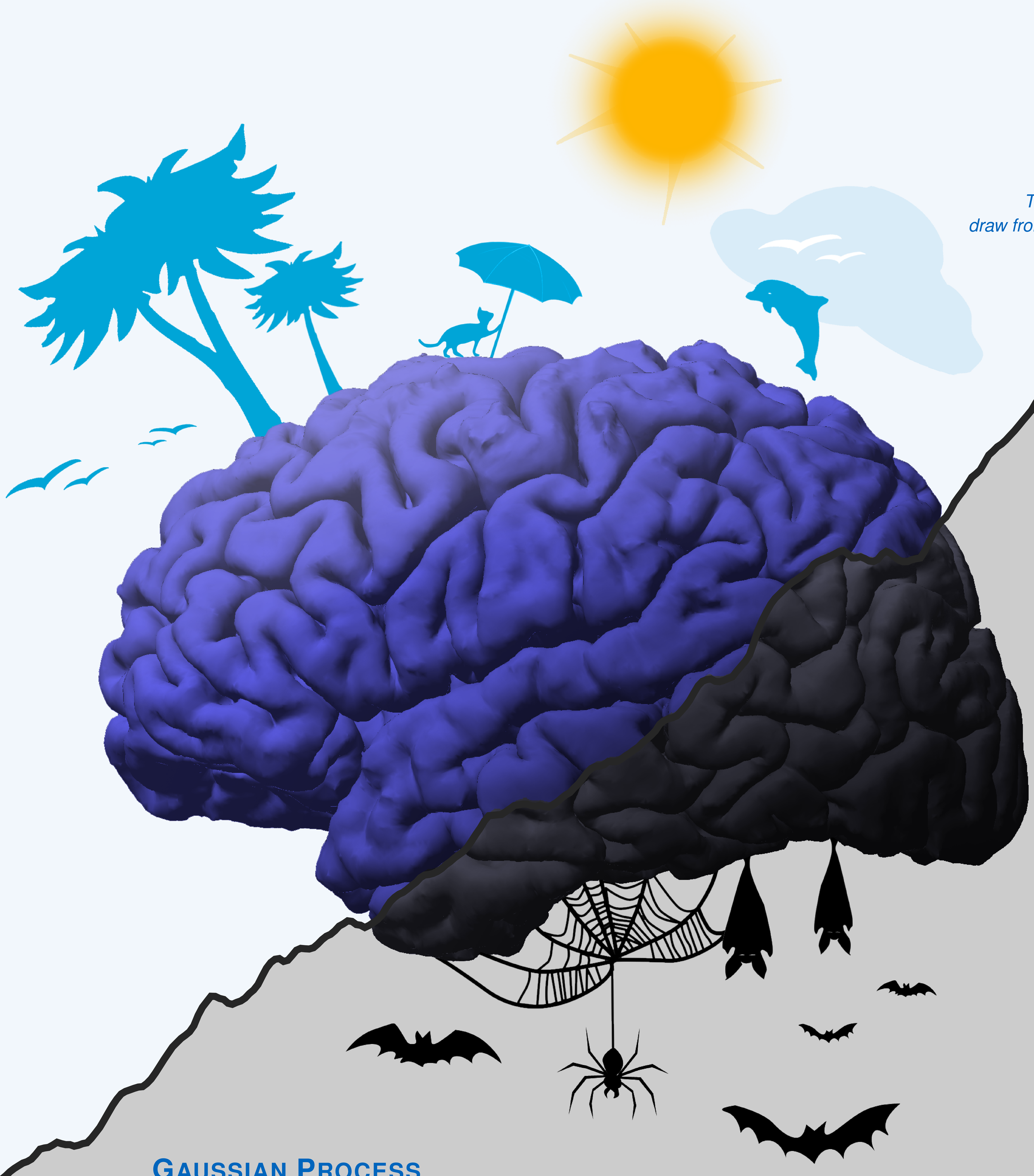
Structural data



- *Source-Based Morphometry* (SBM, [3]) loadings. SBM loadings are derived from structural MRI scans.
- Indicates the concentration of grey matter in different regions of the subject's brain.
- A number of 32 feature regions altogether.

Data and normalization

- The training data $\mathcal{D}$ consist of $n = 86$ subjects, where $\mathbf{x}_i \in \mathbb{R}^{410}$ (378 from the FNC and 32 from the SBM).
- The test data $\mathcal{D}_* = \{(\mathbf{x}_{*,i}, y_{*,i})\}_{i=1}^{n_*}$ consists of $n_* = 119{,}748$ subjects (artificially inflated to prevent hand labeling) with unknown labels y_*.
- Each dimension was normalized in the inputs $\mathbf{x}_i$ and $\mathbf{x}_{*,i}$ by dividing them by the standard deviations from training inputs.



Classification is about separation. This separating line demonstrates a random draw from the latent GP model in one dimension.

Gaussian Process Classification

- The winning model was based on Gaussian process classification [2], which is a Bayesian machine learning method.
- In binary GP classification with observations $y_i \in \{-1, 1\}$, $i = 1, \ldots, n$, associated with inputs $\{\mathbf{x}\}_{i=1}^n$, the observations are considered to be drawn from a Bernoulli distribution with a success probability $p(y_i = 1 \mid \mathbf{x}_i)$.
- The probability is related to a latent function via a sigmoid function that transforms it to a unit interval.
- We use a probit transformation that defines the likelihood model
$$p(y_i \mid f(\mathbf{x}_i)) = \Phi(y_i f(\mathbf{x}_i)) = \int_{-\infty}^{y_i f(\mathbf{x}_i)} \mathcal{N}(z \mid 0, 1)\,\mathrm{d}z,$$
where $\Phi(\cdot)$ is the Gaussian cumulative distribution function.
- A Gaussian process defines the prior distribution over the latent functions:
$$f \sim \mathcal{GP}(0, k(\mathbf{x}, \mathbf{x}')).$$
- The latent Gaussian process model is characterized by its covariance function (kernel) $k(\cdot, \cdot)$.
- We want to account for any linear structure plus some additional short-scale non-linearities in the latent space.
- Therefore we set up the covariance function as a linear combination of three separate covariance functions:
$$k(\mathbf{x}, \mathbf{x}') = k_{\text{const.}}(\mathbf{x}, \mathbf{x}') + k_{\text{linear}}(\mathbf{x}, \mathbf{x}') + k_{\text{Matérn}}^{\nu=5/2}(\mathbf{x}, \mathbf{x}').$$
- The first two covariance function components (a constant and linear covariance function) define an affine model:
$$k_{\text{const.}}(\mathbf{x}, \mathbf{x}') = \theta_1 \quad \text{and} \quad k_{\text{linear}}(\mathbf{x}, \mathbf{x}') = \theta_2\, \mathbf{x}^\mathsf{T} \mathbf{x}'.$$
- The last covariance function gives the model flexibility to adopt to some non-linearities:
$$k_{\text{Matérn}}^{\nu=5/2}(\mathbf{x}, \mathbf{x}') = \theta_3 \left(1 + \frac{\sqrt{5}r}{\theta_4} + \frac{5r}{3\theta_4^2}\right) \exp\left(-\frac{\sqrt{5}r}{\theta_4}\right),$$
where $r = \|\mathbf{x} - \mathbf{x}'\|$.
- This particular Matérn covariance function holds the assumption of the model functions being continuous and rather smooth (twice differentiable).
- The hyperparameters $\boldsymbol{\theta} = \{\theta_1, \theta_2, \theta_3, \theta_4\}$ were given the following hyper-priors: $\theta_1, \theta_2, \theta_3 \sim$ Log-Uniform, and $\theta_4 \sim t_4(0, 1)$. The hyperparameters were initialized as $\boldsymbol{\theta} = \{1, 1, 1, 0.01\}$.
- The training was started by running a Laplace approximation scheme on the model until convergence.
- Final training was performed by sampling (1000 samples, 91 after removing burn-in and thinning).
- We used *Elliptical Slice Sampling* [5] for the latent functions, and the *Surrogate Slice Sampler* [6] for the hyperparameters.
- Class label probabilities $p(y_{*,i} = 1 \mid \mathcal{D}, \mathbf{x}_{*,i})$ for the test set were predicted by the trained model by integrating over the latent functions.

Implementation

- To implement GP classification we used the GPstuff toolbox [7] for Mathworks Matlab (and Octave):
  `http://becs.aalto.fi/en/research/bayes/gpstuff`
- Codes for replicating the winning submission are available online:
  `http://github.com/asolin/MLSP2014-kaggle-challenge`

Conclusions

- Our GP classifier received a final private leaderboard AUC score of 0.92821 on Kaggle, and hence winning the competition.
- This particular GP classifier model was chosen by trying out a couple of models and comparing their performance by leave-one-out cross-validation (LOOCV).
- This model did show promising performance using LOOCV, but the score (AUC) on the public leaderboard (calculated on approximately 52% of the data) on Kaggle was only 0.70536.
- This sort of discrepancy is not uncommon in fields of study, where data is scarce.
- The limited size of the test data set did clearly affect the coherence of the public and private leaderboard scores, making it difficult to predict the true performance of the method based on the public score.
- One evident choice of improvement would be to consider two separate length-scale hyperparameters for the FNC and SBM loadings.
- It is also generally well-known that in GP classification MCMC is more accurate than approximative inference methods such as Expectation propagation (EP) or the Laplace approximation.
- However, the inference times line up in the opposite order. Therefore, for example EP could be a viable option to speed up the inference.

References

[1] Arno Solin and Simo Särkkä "The 10th annual MLSP competition: First place," in *Proceedings of IEEE International Workshop on Machine Learning for Signal Processing*, 2014.

[2] Carl Edward Rasmussen and Christopher K. I. Williams, *Gaussian Processes for Machine Learning*, The MIT Press, 2006.

[3] Judith M. Segall, Elena A. Allen, Rex E. Jung, Erik B. Erhardt, Sunil Kumar Arja, Kent A. Kiehl, and Vince D. Calhoun, "Correspondence between structure and function in the human brain at rest," *Frontiers in Neuroinformatics*, vol. 6, no. 10, 2012.

[4] Elena A. Allen, Eswar Damaraju, Sergey M. Plis, Erik B. Erhardt, Tom Eichele, and Vince D. Calhoun, "Tracking whole-brain connectivity dynamics in the resting state," *Cerebral Cortex*, pp. 663–676, 2012.

[5] Iain Murray, Ryan P. Adams, and David Mackay, "Elliptical slice sampling," in *International Conference on Artificial Intelligence and Statistics*, 2010, pp. 541–548.

[6] Iain Murray and Ryan P. Adams, "Slice sampling covariance hyperparameters of latent Gaussian models," in *Advances in Neural Information Processing Systems*, 2010, pp. 1732–1740.

[7] Jarno Vanhatalo, Jaakko Riihimäki, Jouni Hartikainen, Pasi Jylänki, Ville Tolvanen, and Aki Vehtari, "GPstuff: Bayesian modeling with Gaussian processes," *Journal of Machine Learning Research*, vol. 14, no. 1, pp. 1175–1179, 2013.

Contact details: arno.solin@aalto.fi, web address: http://arno.solin.fi. This work was presented at MLSP 2014, Reims, France.