

Вероятностное программирование: наблюдения

- Большинство задач машинного обучения могут быть сформулированы в Байесовской постановке
- Такая постановка подразумевает задание гипотез распределения данных и параметров, а также метод определения оптимальных параметров (MLE, max posterior, ...)
- Например, классический метод PCA в Байесовской постановке записывается следующим образом:

$$p(\mathbf{x}_n | \mathbf{y}_n, \mathbf{W}, v) = N(\mathbf{x}_n | \mathbf{W}^T \mathbf{y}_n, v\mathbf{I}),$$
$$p(\mathbf{y}_n) = N(\mathbf{y}_n | \mathbf{0}, \mathbf{I}), \quad p([\mathbf{W}]_d) = N(\mathbf{w}_d | \mathbf{0}, \text{diag}([\alpha_1, \dots, \alpha_D])).$$

Оптимальные параметры определяются методом MLE,

$$\max(p(\mathbf{x}_n | \mathbf{y}_n, \mathbf{W}, v)), \quad \text{при } v \rightarrow 0$$

Вероятностное программирование: идея

- Вероятностное программирование предоставляет язык описания Байесовских моделей
- С помощью этого языка исследователь тратит основное время на обдумывание структуры модели, а не на разработку программного кода
- Вероятностное программирование компилирует сгенерированную модель и выполняет процедуры определения оптимальных параметров наиболее приемлемым методом

Вероятностное программирование: средства

- BUGS: Bayesian inference using Gibbs sampling
- Infer.NET: вероятностное программирование на платформе .NET
- PMTK: Probabilistic Modeling Toolkit for Matlab/Octave

Способы построения модели:

Creating a model

```
model = fooCreate(...) % manually specify parameters
```

```
model = fooFit(X, y, ...) % Compute ML or MAP estimate of params
```

```
model = fooFitBayes(X, y, ...) % Compute posterior of params
```

Using a model for prediction

```
[yhat, py] = fooPredict(model, Xtest) % plugin approximation
```

```
[yhat, py] = fooPredictBayes(model, Xtest) % posterior predictive
```

MLComp

- MLComp – система, служащая для сравнения алгоритмов машинного обучения на различных наборах данных (полигон)
- Позволяет
 - Выкладывать свои наборы данных и запускать на них имеющиеся в библиотеке алгоритмы машинного обучения
 - Выкладывать свои алгоритмы и запускать их на имеющихся наборах данных, сравниваясь с другими участниками
- Среди недостатков «Только одно разбиение задачи на обучение и контроль, в результате низкая статистическая надежность получаемых оценок» (Лисица, Воронцов и др., 2010)

РМТК и MLCOMP

- РМТК предоставляет ряд интерфейсов для загрузки своих моделей в MLComp, тестирования их на реальных данных и сравнения результатов
- В частности, функция `mlcompCompiler.m` генерирует программу на Octave нужного формата

```
mlcompCompiler(fitFn, predictFn, outputDir, fitOpts, predictOpts)
```

где

- `fitFn` – модель,
- `predictFn` – значение прогноза,
- `outputDir` – адрес временной директории для хранения промежуточных файлов
- `fitOpts` – массив аргументов для модели `fitFn`
- `predictOpts` – массив параметров предсказания `predictFn`