# **BNPlib** for density estimation:

A nonparametric C++ library (part 2)

Bruno Guindani Elena Zazzetti

January 8<sup>th</sup>, 2020



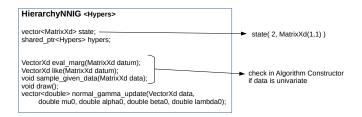
https://github.com/poliprojects/BNPlib

# Algoritmhs

## Algorithm <Hierarchy<Hypers>.Mixture> unsigned int maxiter = 20000; unsigned int burnin = 5000; int num clusters; MatrixXd data; vector<unsigned int> allocations; vector<Hierarchy<Hypers>> unique values; Neal2 <Hierarchy<Hypers>.Mixture> void initialize() override: void sample allocations() override; void sample unique values() override; Neal8 <Hierarchy<Hypers>.Mixture> unsigned int n\_aux = 3; vector<Hierarchy<Hypers>> aux unique values;

void sample allocations() override;

## Hierarchies



# HierarchyDummy <Hypers> vector<MatrixXd> state; shared\_ptr<Hypers> hypers; VectorXd eval\_marg(MatrixXd datum); VectorXd like(MatrixXd datum); void sample\_given\_data(MatrixXd data); void draw(); std:vector<Einen::MatrixXd> dummy\_update(MatrixXd data.

VectorXd mu0. MatrixXd lambda0):

# **Hypers**

#### HypersFixedNNIG

double mu0, lambda, alpha0, beta0;

#### HypersDummy

VectorXd mu0; MatrixXd lambda0;

## **Mixtures**

#### DirichletMixture

double totalmass:

double const prob\_existing\_cluster( int card, unsigned int n) double const prob\_new\_cluster( unsigned int n, unsigned int n\_unique)

#### PitYorMixture

double strength; double discount:

double const prob\_existing\_cluster( int card, unsigned int n) double const prob\_new\_cluster( unsigned int n, unsigned int n\_unique)  $\,$ 

# Algorithms

factories input

```
class Factory{
private:
    std::map<std::string, AlgoBuilderType> storage;
public:
    static Factory& Instance();
    void add_builder(const std::string &name, const AlgoBuilder auto create_algorithm(const std::string &name) const {};
//[...]
```

## Multivariate Proto

```
message Par_Col {
        repeated double elems = 1;
message Param {
        repeated Par_Col par_cols= 1;
message UniqueValues {
        repeated Param params= 1;
    }
message IterationOutput {
    repeated int32 allocations = 1;
    repeated UniqueValues uniquevalues = 2;
}
message ChainOutput {
    repeated IterationOutput chain = 1;
```

### Collectors

