Help

```
#ifndef _HESTON_H
#define _HESTON_H

#include "
href../../../../common/math/mcam/src/DupireModel_h_src.pdfModel.hpp"
#include "pnl/pnl_matrix.h"
#include "pnl/pnl_random.h"

namespace mcam {

/**
 * Heston Model
 *
 *
 * \ f[ dS^i_t = interest S^i_t dt + \ sqrt{\ sigma_t^i} S^i_t dB^i_t \ f]
 * \ f[ d\ sigma^i_t = \ kappa^i (a^i - \ sigma_t^i) dt + \ nu^i_t \ sqrt{\ sigm
 * where B and \ f$\ tilde B\ f$ are independent and
 * \ f[ d\ langle B \ rangle_t = \ Gamma_S dt = (\ rho + diag(1 - \ rho)) dt \ f
 * \ f[ d\ langle \ tilde B \ rangle_t = \ Gamma_\ sigma dt = (\ xi + diag(1 - \
 *
 * This can be equivalently rewritten
 * \ f[ dS^i_t = interest S^i_t dt + \ sqrt{\ sigma_t^i} S^i_t dB^i_t \ f]
 * \ f[ d\ sigma^i_t = \ kappa^i (a^i - \ sigma_t^i) dt + \ nu^i_t \ sqrt{\ sigm
 * where
 * \ f[ d\ langle B \ rangle_t = \ Gamma_S dt = (\ rho + diag(1 - \ rho)) dt \ f
 * \ f[ d\ langle B, W \ rangle_t = \ gamma \ Gamma_S dt \ f]
 * \ f[ d\ langle W \ rangle_t = (\ gamma^2 \ Gamma_S + (1 - \ gamma^2) \ Gamma_
 */
class HestonModel : public Model
{
private:
    double gamma;            /*!< Correlation between the vol and asset brownian
    double rho;              /*!< Correlation between the asset brownian motions
    PnlVect* sigma0;         /*!< Initial volatilities */
    PnlVect* voVol;          /*!< Volatility of Volatility (nu) */
    PnlVect* sigmaVector;    /*!< Instantaneous volatility vector */
    PnlVect* reverting_rate; /*!< Rate of mean revertion (kappa) */
    PnlVect* long_run_var;   /*!< mean level (a) */
    PnlMat* covChol;
```

```
public:
    HestonModel();
    HestonModel(const Param& P);
    ~HestonModel();
    int computeCovChol();
    void print() const;
    virtual void path(const PnlMat *G);
    virtual PnlVect* getMin() const;
    virtual PnlVect* getMax() const;
    virtual PnlMat* getRegressor() const;
};

}
#endif /* _HESTON_H */
```