

## 7\_1

May 2, 2016

```
In [154]: import numpy as np
import matplotlib.pyplot as plt
import scipy.stats as stats
%matplotlib inline
```

### 0.1 Оценка мат. ожидания

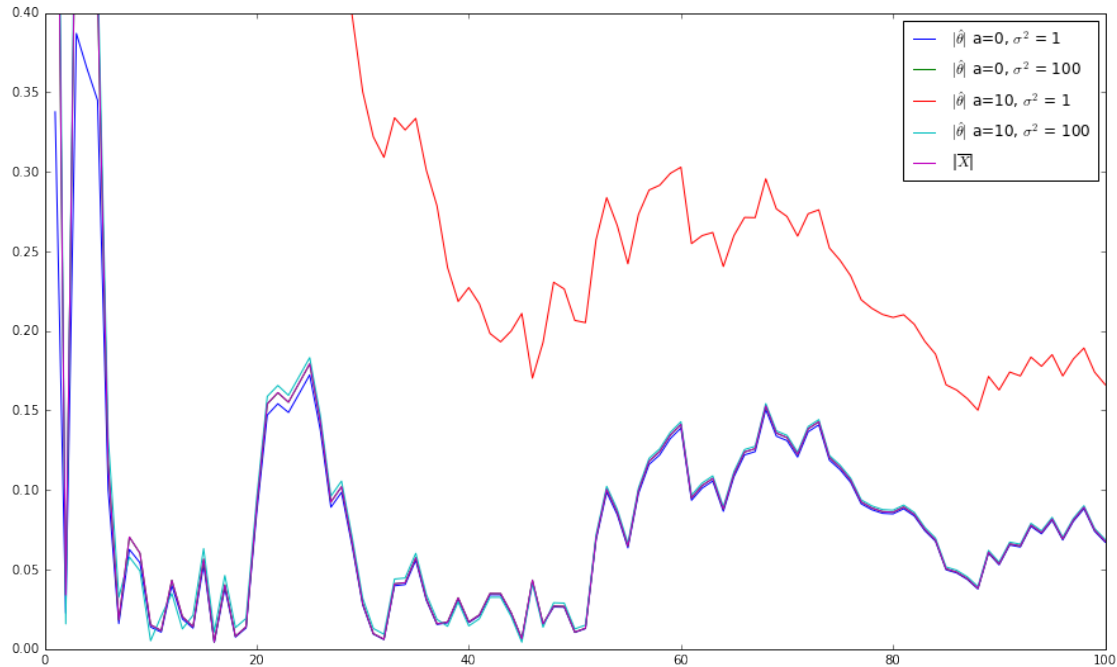
Сопряженное к нормальному  $N(\theta, 1)$  - нормальное  $N(a, \sigma^2)$ . Байесовская оценка мат. ожидания  $\hat{\theta}_n = \frac{\sigma^2 \cdot n \cdot \bar{X} + a}{\sigma^2 \cdot n + 1}$ . Оценка методом макс. правдоподобия равна  $\bar{X}$

```
In [155]: params = ((0, 1), (0, 100), (10, 1), (10, 100))
```

```
In [156]: def bayers_expectation_est(sample, a, sigmaSq):
    n = len(sample)
    avg = np.average(sample)
    return (sigmaSq * n * avg + a) / (sigmaSq * n + 1)
```

```
In [157]: N = 100
def draw_plot():
    sample = stats.norm.rvs(size=N)
    grid = np.arange(1, N+1, 1)
    plt.figure(figsize=(15, 9))
    for param in params:
        plt.plot(grid,
                 np.abs([bayers_expectation_est(sample[:n], param[0], param[1]) for n in grid]),
                 label=r'$\vert \hat{\theta} \vert$' + r' a={:}, $\sigma^2$ = {}'.format(param[0], param[1]))
    plt.plot(grid, np.abs([np.average(sample[:n]) for n in grid]), label=r'$\overline{X}$')
    plt.legend(loc='best')
    plt.ylim(0, 0.4)
    plt.show()
```

```
In [161]: draw_plot()
```



**Вывод.** Из графиков видно, что байесовские оценки также хороши, как и оценка методом максимального правдоподобия. Все байесовские оценки из предложенных, кроме оценки с априорным распределением  $N(10, 1)$ , показывают такой же результат, как и оценка методом максимального правдоподобия.

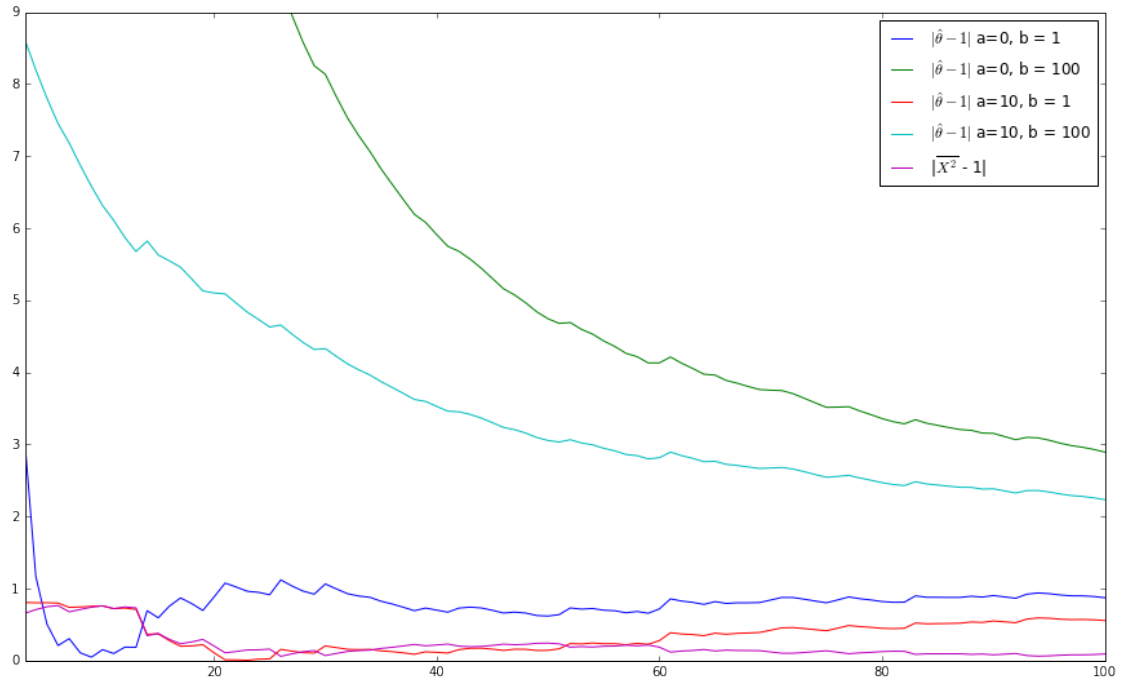
## 0.2 Оценка дисперсии

Сопряженное распределение для  $N(0, \theta)$  - гамма распределение  $Gamma(a, b)$ , для которого байесовская оценка равна  $\hat{\theta} = \frac{2(b+n\bar{X}^2)}{2\cdot a+n-2}$  Оценка методом максимального правдоподобия равна  $\hat{\theta} = \bar{X}^2$

```
In [151]: def bayers_dispersion_est(sample, a, b):
            n = len(sample)
            s = np.sum(sample**2)
            return 2*(s + b) / (2 * a + n - 2)

In [152]: def draw_dispersions_plot():
            sample = stats.norm.rvs(size=N)
            grid = np.arange(3, N+1, 1)
            plt.figure(figsize=(15, 9))
            for param in params:
                plt.plot(grid,
                         np.abs(np.array([bayers_dispersion_est(sample[:n], param[0], param[1]) for n
                                         label=r'$\vert \hat{\theta} - 1 \vert$' + r' a={:}, b = {:}'].format(param[0],
            plt.plot(grid, np.abs(np.array([np.average(sample[:n] ** 2) for n in grid]) - 1), label=r'$\bar{X}^2$')
            plt.legend(loc='best')
            plt.ylim(0, 9)
            plt.xlim(3, 100)
            plt.show()

In [153]: draw_dispersions_plot()
```



**Вывод.** Из графиков видно, что оценка методом максимального правдоподобия лучше байесовских. При оценки дисперсии для  $N(0, \theta)$  среди байесовских с априори сопряженным распределением из представленных лучше всего ведет себя оценка  $\Gamma(10, 1)$ .