Метод существенной выборки для оценивания границ доверительных интервалов в задачах логистической регрессии

Леушева Виктория Витальевна, гр. 522

Санкт-Петербургский государственный университет
Математико-механический факультет
Кафедра статистического моделирования

Научный руководитель: д. ф.-м. н., профессор М. С. Ермаков
Рецензент: к. ф.-м. н., доцент Ю. Н. Каштанов

Санкт-Петербург
2014г.

Введение

Достоинство логистической регрессии: можно оценить *вероятность* исхода какого — либо события, принимающего два возможных значения 1 или 0.

Цель работы:

- Оценить малые вероятности отклонения оценок параметров логистической регрессии и нейронной сети с помощью метода существенной выборки;
- Построить оценки точных значений границ доверительных интервалов для оценок параметров логистической регрессии и нейронной сети.

Постановка задачи №1

Проведем численное моделирование на примере модели логистической регресии с двумя весовыми коэффициентами $\beta = (\beta_0, \beta_1)$.

Независимый признак $X = (x_1, \ldots, x_n)^\mathsf{T}$.

Логистическая функция:

$$p(X;\beta) = \frac{1}{1 + \exp(-(\beta_0 + \beta_1 X))}.$$

Зависимый признак $Y = (y_1, \ldots, y_n)^\mathsf{T}$ моделируем с вероятностью $p(X;\beta)$:

$$P(y_i = 1|x_i) = p(x_i;\beta) \qquad \text{и} \qquad P(y_i = 0|x_i) = 1 - p(x_i;\beta).$$

Параметр β_1 будем считать неизвестным.

Задача оценивания вероятности отклонения оценки параметра от его истинного значения:

$$\mathrm{A} = P((\hat{\beta}_1 - \beta_1) > \Delta).$$

Метод максимального правдоподобия

Функция правдоподобия:

$$L(X;\beta) = \prod_{i=1}^{n} p(x_i;\beta)^{y_i}(1-p(x_i;\beta))^{1-y_i} \to \max.$$

Логарифмируем функцию правдоподобия и записываем в более удобном виде:

$$\ln L(X;\beta) = \sum_{i=1}^{n} y_i \ln \frac{p(x_i;\beta)}{1-p(x_i;\beta)} + \sum_{i=1}^{n} \ln(1-p(x_i;\beta)).$$

Используя свойство логистической модели $\ln \dfrac{p(x)}{1-p(x)} = \beta_0 + \beta_1 X$, получаем:

$$\ln L(X;\beta) = \sum_{i=1}^{n} y_i(\beta_0+\beta_1 x_i) - \sum_{i=1}^{n} \ln(1+\exp(\beta_0+\beta_1 x_i)).$$

Дифференцирование по неизвестному параметру β_1:

$$\frac{\partial \ln L(X;\beta)}{\partial \beta_1} = \sum_{i=1}^{n} y_i x_i - \sum_{i=1}^{n} \frac{\exp(\beta_0+\beta_1 x_i)}{1+\exp(\beta_0+\beta_1 x_i)} x_i = 0,$$

$$\sum_{i=1}^{n} y_i x_i = \sum_{i=1}^{n} p(x_i;\beta) x_i.$$

Итерационный численный метод Ньютона — Рафсона

Определим функционал $Q(\beta)$:

$$Q(\beta) = -\left(\sum_{i=1}^{n} y_i \ln p(x_i;\beta) + (1-y_i)\ln(1-p(x_i;\beta))\right).$$

Алгоритм метода Ньютона — Рафсона:

1. Задаем начальное приближение оценки неизвестного параметра β_1:

$$\beta_1^0 = \ln\frac{\bar{Y}}{(1-\bar{Y})}, \text{ где } \bar{Y} = \frac{1}{n}\sum_{i=1}^{n} y_i;$$

2. Находим последующие приближения в соответствии с рекуррентным соотношением:

$$\beta_1^k = \beta_1^{k-1} - \eta_{k-1}(Q''(\beta_1^{k-1}))^{-1}Q'(\beta_1^{k-1}), \text{ где } \eta - \text{ шаг ;}$$

3. Повторяем пункт 2, пока значения β_1^k не сойдутся.

Где:

$$Q'(\beta) = \frac{\partial Q(\beta)}{\partial \beta_1} = -\sum_{i=1}^{n}(y_i - p(x_i;\beta))x_i,$$

$$Q''(\beta) = \frac{\partial^2 Q(\beta)}{\partial \beta_1^2} = \sum_{i=1}^{n}(1-p(x_i;\beta))p(x_i;\beta)x_i^2.$$

Метод существенной выборки

Наша задача состоит в вычислении вероятности:

$$\mathrm{A} = P((\hat{\beta}_1 - \beta_1) > \Delta).$$

- Исходная мера $p(X;\beta_1) \Longleftrightarrow$ Новая мера $p(X;\beta_1+\Delta)$, где $\Delta = \dfrac{b\sqrt{\mathrm{D}\hat{\beta}_1}}{\sqrt{n}}$.
- Моделируем k независимых величин $Y^{(j)} = (y_1^{(j)},\ldots,y_n^{(j)}), j = \overline{1,k}$ с вероятностью $p(X;\beta_1+\Delta)$.
- Считаем k оценок неизвестного параметра: $\hat{\beta}_1{}^{\Delta}$.
- В качестве оценки вероятности берем:

$$\hat{\mathrm{A}} = \frac{1}{k}\sum_{j=1}^{k} \mathcal{X}_{\{\hat{\beta}_{1j}^{\Delta} > \beta_1+\Delta\}} \prod_{i=1}^{n} \frac{p(x_i;\beta_1)}{p(x_i;\beta_1+\Delta)}, \text{ где}$$

$$p(x_i;\beta_1) = \frac{1}{1+\exp(-\beta_0-\beta_1 x_i)},$$

$$p(x_i;\beta_1+\Delta) = \frac{1}{1+\exp(-\beta_0-(\beta_1+\Delta)x_i)}.$$

Результаты численного моделирования. Зависимость от n

Зависимость результата при различных значениях параметра n.

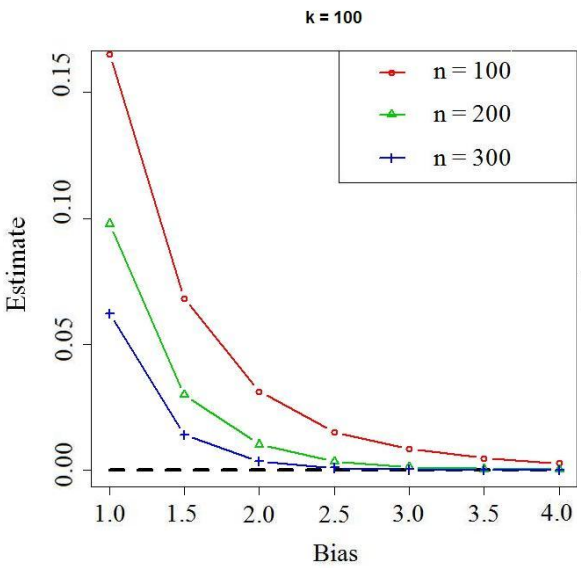

Рис. 1: Оценки вероятности при $k = 100$, $n = 100, 200, 300$

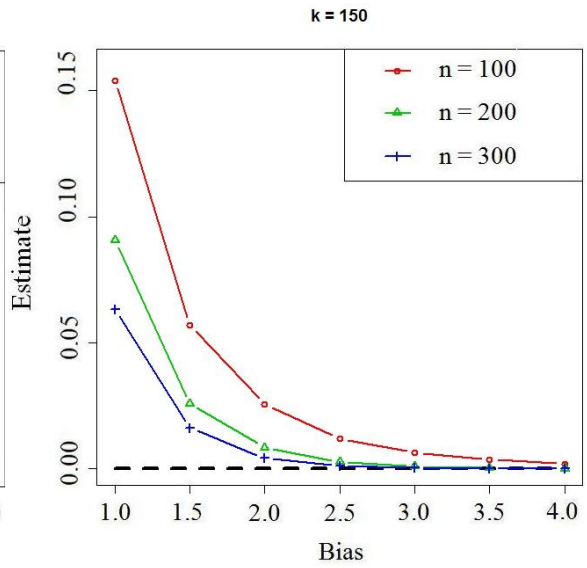

Рис. 2: Оценки вероятности при $k = 150$, $n = 100, 200, 300$

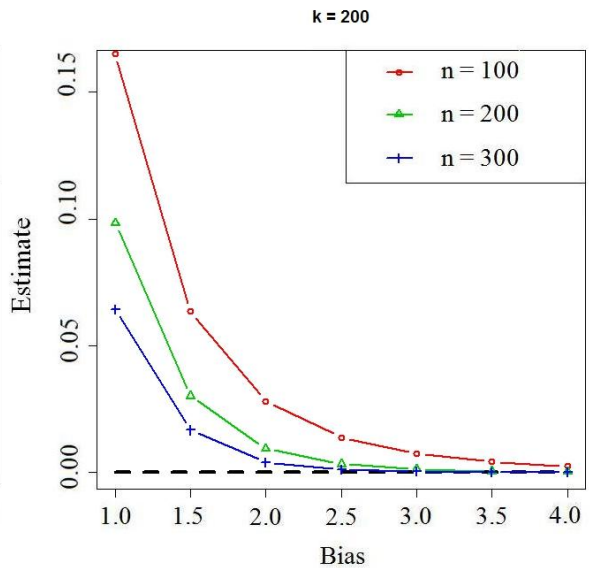

Рис. 3: Оценки вероятности при $k = 200$, $n = 100, 200, 300$

Результаты численного моделирования. Зависимость от k

Зависимость результата при различных значениях параметра k.

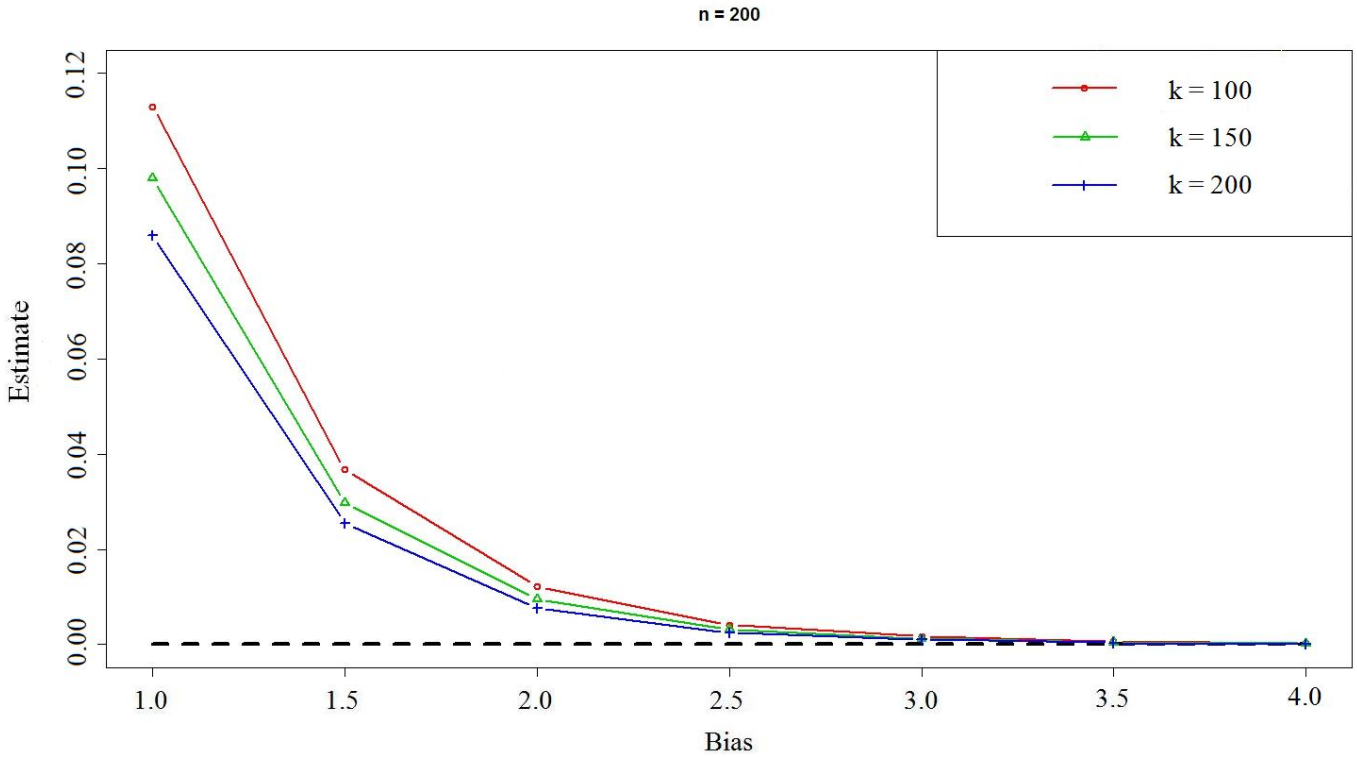

Рис. 4: Оценки вероятности при $n = 200, k = 100, 150, 200$

Оценка имеет вид:

$$\hat{\mathrm{A}} = \frac{1}{k}\sum_{j=1}^{k} \mathcal{X}_{\{\hat{\beta}_{1j}^{\Delta} > \beta_1 + \Delta\}} \prod_{i=1}^{n} \frac{p(x_i;\beta_1)}{p(x_i;\beta_1+\Delta)}.$$

Для построения доверительных интервалов, считаем искомые оценки m раз, находим среднее значение $\mathrm{mean}(\hat{\mathrm{A}})$ и стандартное отклонение оценки s.

Доверительный интервал имеет вид:

$$\left(\mathrm{mean}(\hat{\mathrm{A}}) - c_\gamma \frac{s}{\sqrt{n}}, \mathrm{mean}(\hat{\mathrm{A}}) + c_\gamma \frac{s}{\sqrt{n}}\right),$$

где $c_\gamma = \Phi^{-1}\left(\frac{1+\gamma}{2}\right)$.

Доверительные интервалы для оценок малых вероятностей.

Таблица 1: $k = 100, n = 100$

Смещение	1	1.5	2	2.5	3	3.5
Средние значение	0.16106	0.06459	0.02810	0.01374	0.00733	0.00420
Нижняя граница	0.15930	0.06376	0.02774	0.01354	0.00724	0.00415
Верхняя граница	0.16282	0.06542	0.02846	0.01394	0.00742	0.00425

Таблица 2: $k = 100, n = 200$

Смещение	1	1.5	2	2.5	3	3.5
Средние значение	0.09913	0.02953	0.00959	0.00337	0.00126	0.00050
Нижняя граница	0.09792	0.02920	0.00947	0.00332	0.00124	0.00049
Верхняя граница	0.10034	0.02986	0.00971	0.00342	0.00128	0.00051

Таблица 3: $k = 100, n = 300$

Смещение	1	1.5	2	2.5	3	3.5
Средние значение	0.07479	0.01872	0.00504	0.00146	0.00045	0.000148
Нижняя граница	0.07391	0.01849	0.00497	0.00144	0.00044	0.000146
Верхняя граница	0.07567	0.01895	0.00511	0.00147	0.00046	0.000150

Нейронная сеть. Математическая модель

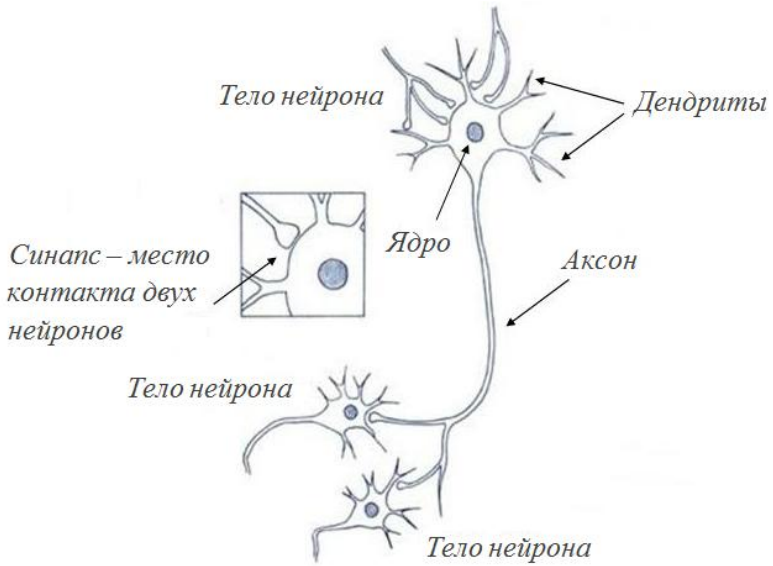

Рис. 5: Биологическая модель

Нейронные сети возникли из попыток воспроизвести способность биологических нервных систем обучаться и исправлять ошибки, моделируя низкоуровневую структуру мозга.

Структурная схема нейрона:

- Входные сигналы $X_1, \ldots, X_n$;
- Весовые коэффициенты $\beta_0, \ldots, \beta_n$;
- Сумматор $\sum = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n X_n$;
- Функция активации нейрона $\sigma(\sum) = \dfrac{1}{1 + \exp(-\sum)}$;
- Выход Y.

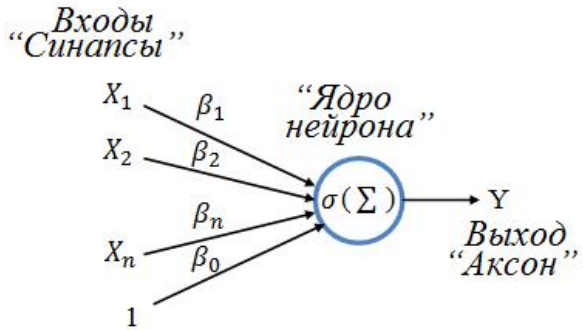

Рис. 6: Математическая модель

Постановка задачи №2

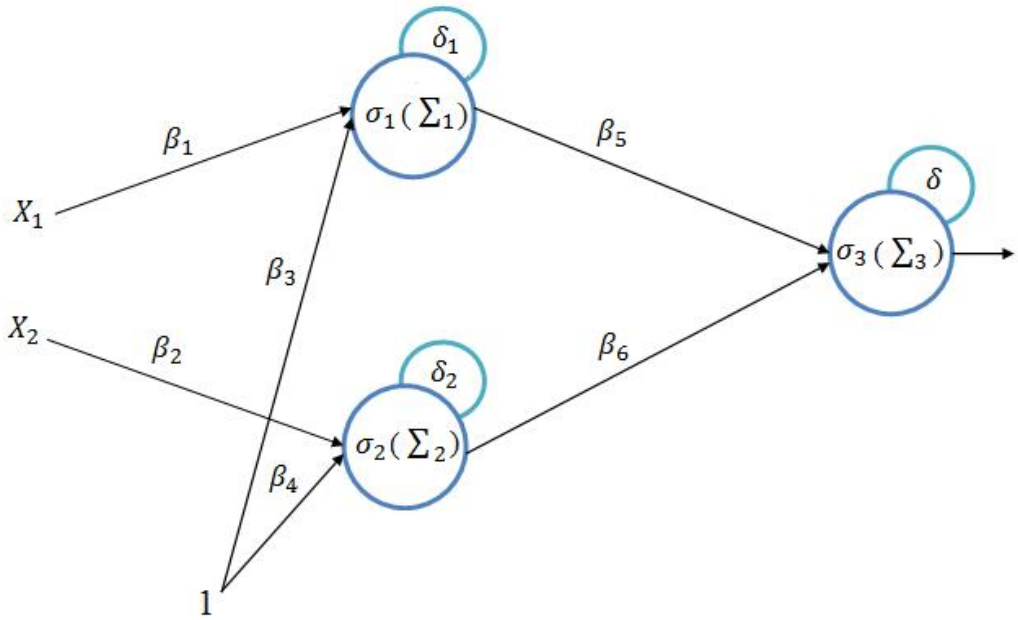

Рис. 7: Нейронная сеть

Дано:

- Входной слой X_1, X_2;
- Скрытый слой $\sigma_1(\sum_1) = \dfrac{1}{1+\exp(-(\beta_3+\beta_1 X_1))}$,
$\sigma_2(\sum_2) = \dfrac{1}{1+\exp(-(\beta_4+\beta_2 X_2))}$;
- Весовые коэффициенты $\beta_1, \ldots, \beta_6$;
- Выходной слой $\sigma_3(\sum_3) = \dfrac{1}{1+\exp(-(\beta_5\sigma_1(\sum_1)+\beta_6\sigma_2(\sum_2)))}$.

Метод обратного распространения ошибки

- Инициализируем веса: $(\beta_1, \ldots, \beta_6)$.
- **Прямой ход.**
  Находим выходные значения сети на скрытом слое:
  $$\sigma_1(\sum_1) = \frac{1}{1+\exp(-(\beta_3+\beta_1 X_1))}, \qquad \sigma_2(\sum_2) = \frac{1}{1+\exp(-(\beta_4+\beta_2 X_2))}.$$
  Находим выходное значение сети на выходном слое:
  $$\sigma_3(\sum_3) = \frac{1}{1+\exp(-(\beta_5\sigma_1(\sum_1)+\beta_6\sigma_2(\sum_2)))}.$$
  Пусть $Y_{\text{иск}} = Y_{\text{вых}} + \xi$, где $\xi \sim N(0, s^2)$. Ошибка для выходного слоя: $\delta = \xi$.
- **Обратный ход.**
  Находим производные функции активации:
  $\sigma_1'(\sum_1) = \sigma_1(\sum_1)(1-\sigma_1(\sum_1))$, $\sigma_2'(\sum_2) = \sigma_2(\sum_2)(1-\sigma_2(\sum_2))$,
  $\sigma_3'(\sum_3) = \sigma_3(\sum_3)(1-\sigma_3(\sum_3))$.
  Ошибки для скрытого слоя: $\delta_1 = \delta\sigma_3'(\sum_3)\beta_5$ и $\delta_2 = \delta\sigma_3'(\sum_3)\beta_6$.
- **Градиентный шаг** (η — темп обучения)**:**
  $\beta_1 = \beta_1 - \eta\delta_1\sigma_1'(\sum_1)X_1$, $\beta_2 = \beta_2 - \eta\delta_2\sigma_2'(\sum_2)X_2$,
  $\beta_3 = \beta_3 - \eta\delta_1\sigma_1'(\sum_1)$, $\beta_4 = \beta_4 - \eta\delta_2\sigma_2'(\sum_2)$,
  $\beta_5 = \beta_5 - \eta\delta\sigma_3'(\sum_3)\sigma_1(\sum_1)$, $\beta_6 = \beta_6 - \eta\delta\sigma_3'(\sum_3)\sigma_2(\sum_2)$.

Алгоритм решения

Пусть мы знаем истинные значения весовых коэффициентов $\beta_1, \ldots, \beta_6$. Будем оценивать параметр β_1.

Вспомним задачу оценивания вероятности отклонения оценки параметра от его истинного значения:

$$\mathrm{A} = P((\hat{\beta}_1 - \beta_1) > \Delta).$$

Согласно методу существенной выборки в качестве оценки вероятности берем:

$$\hat{\mathrm{A}} = \frac{1}{k} \sum_{j=1}^{k} \mathcal{X}_{\{\hat{\beta}_{1j}^{\Delta} > \beta_1 + \Delta\}} \prod_{i=1}^{n} \frac{\frac{1}{\sqrt{2\pi}} \exp(-(Y_{\text{вых}} - \sigma_3(\sum_3; \beta_1))/2s^2)}{\frac{1}{\sqrt{2\pi}} \exp(-(Y_{\text{вых}} - \sigma_3(\sum_3; \beta_1 + \Delta))/2s^2)},$$

где s^2 — стандартное отклонение ошибки δ.

Численное моделирование. Нейронная сеть. Зависимость от n

Зависимость результата при различных значениях параметра n.

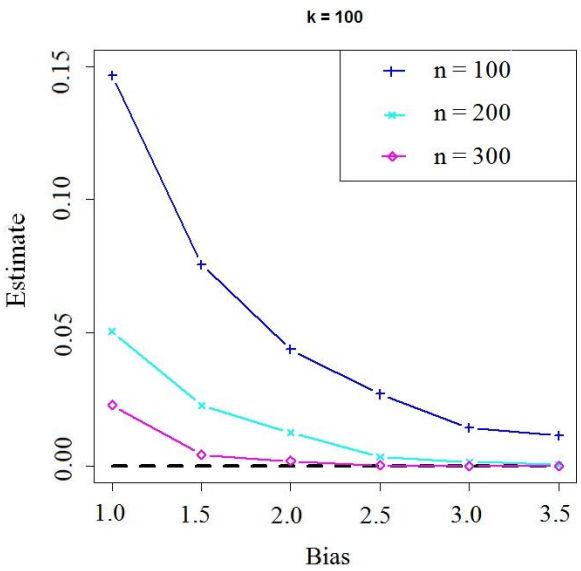

Рис. 8: Оценки вероятности при $k = 100$, $n = 100, 200, 300$

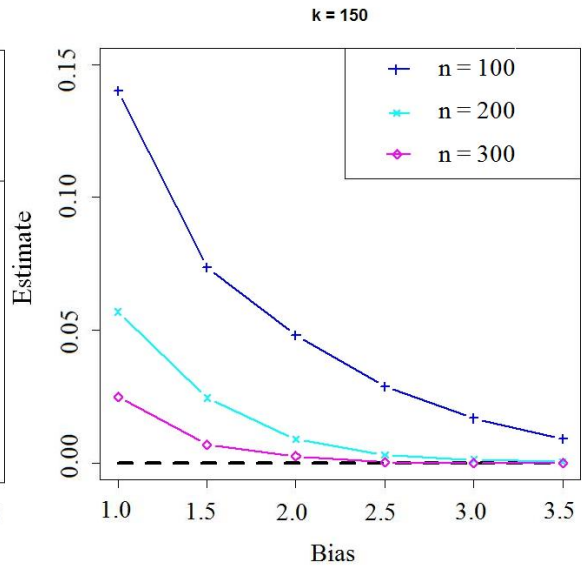

Рис. 9: Оценки вероятности при $k = 150$, $n = 100, 200, 300$

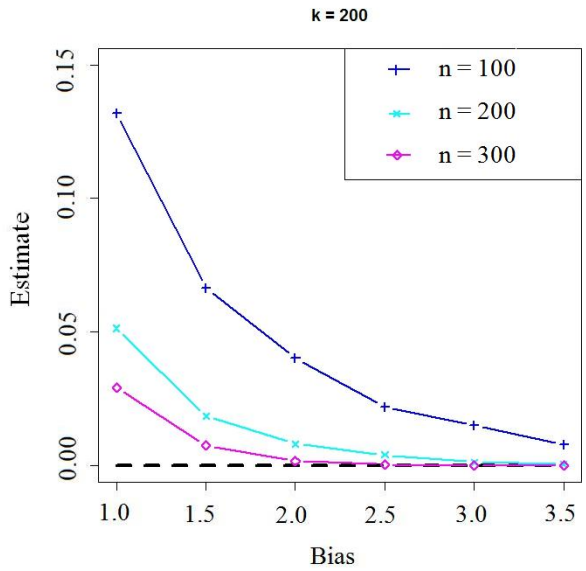

Рис. 10: Оценки вероятности при $k = 200$, $n = 100, 200, 300$

Доверительные интервалы для оценок малых вероятностей.

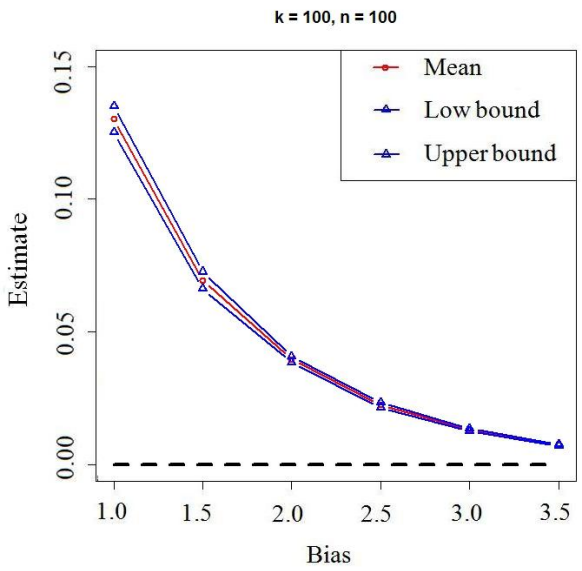

Рис. 11: Доверительные интервалы при $k = 100$, $n = 100$

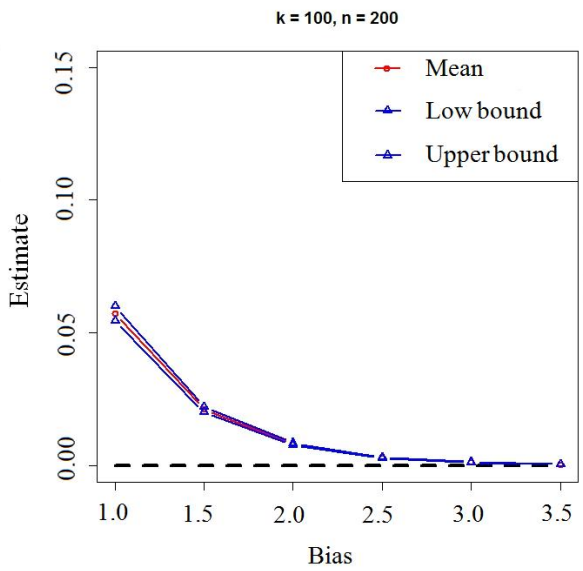

Рис. 12: Доверительные интервалы при $k = 100$, $n = 200$

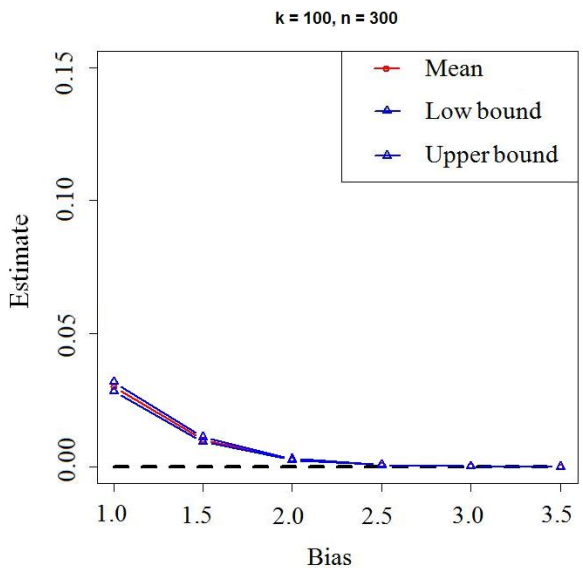

Рис. 13: Доверительные интервалы при $k = 100$, $n = 300$

Заключение

1. Рассмотрена задача вычисления малых вероятностей отклонения оценок параметров с помощью метода существенной выборки для
   - Логистической регрессии;
   - Нейронной сети.
2. Полученные результаты могут быть применены для построения доверительных интервалов оценок.
3. Построенные доверительные интервалы для оценок малых вероятностей распределений параметров логистической регрессии и нейронной сети показывают высокую точность предложенной процедуры моделирования.
4. Высокое качество предложенной процедуры моделирования позволяет предположить, что она будет хорошо работать для более сложных моделей нейронных сетей.