|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ *Робототехники и комплексной автоматизации*

КАФЕДРА *Системы автоматизированного проектирования (РК-6)*

**ОТЧЕТ О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ**

по дисциплине: «Вычислительная математика»

|  |  |  |
| --- | --- | --- |
| Студент |  | Петраков Станислав Альбертович |
| Группа |  | РК6-56Б |
| Тип задания |  | лабораторная работа |
| Тема лабораторной работы |  | Интерполяция в условиях с измерений с неопределенностью (вариант 5) |

Студент **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_Петраков С.А.\_\_**

*подпись, дата фамилия, и.о.*

Преподаватель **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_Соколов А.П. \_\_**

*подпись, дата фамилия, и.о.*

*Москва, 2021 г.*

Оглавление

[Задание на лабораторную работу 3](#_Toc84187572)

[Цель выполнения лабораторной работы 5](#_Toc84187573)

[Выполненные задачи 5](#_Toc84187574)

[1. Разработать алгоритм вычисления коэффициентов кубического сплайна 6](#_Toc84187575)

[2. Вычисление значений кубического сплайна на промежутке и его первой производной 7](#_Toc84187576)

[3. Построение кубического сплайна 7](#_Toc84187577)

[4. Интерполяция полиномом Лагранжа 9](#_Toc84187578)

[5. Анализ влияния погрешности входных данных на интерполяционный полином Лагранжа 10](#_Toc84187579)

[6. Анализ влияния погрешности входных данных на кубический сплайн 14](#_Toc84187580)

[Заключение 17](#_Toc84187581)

[Список использованных источников 18](#_Toc84187582)

# Задание на лабораторную работу

Интерполяция, вероятно, является самым простым способом определения недостающих значений некоторой функции при условии, что известны соседние значения. Однако, за кадром зачастую остается вопрос о том, насколько точно мы знаем исходные данные для проведения интерполяции или любой другой аппроксимации. К примеру, исходные данные могут быть получены путем снятия показаний с датчиков, которые всегда обладают определенной погрешностью. В этом случае всегда возникает желание оценить влияние подобных погрешностей и неопределенностей на аппроксимацию. В этом задании на простейшем примере мы познакомимся с интерполяцией в целом (базовая часть) и проанализируем, как неопределенности влияют на ее предсказания (продвинутая часть).

Базовая часть:

1. Разработать функцию *qubic\_spline\_coeff(x\_nodes, y\_nodes),* которая посредством решения матричного уравнения вычисляет коэффициенты естественного кубического сплайна. Для простоты, решение матричного уравнения можно производить с помощью вычисления обратной матрицы с использованием функции *numpy.linalg.inv()*
2. Написать функции *qubic\_spline(x, qs\_coeff)* и *d\_qubic\_spline(x, qs\_coeff)*, которые вычисляют соответственно значение кубического сплайна и его производной в точке *x* (*qs\_coeff* обозначает матрицу коэффициентов).
3. Используя данные в таблице 1, требуется построить аппроксимацию зависимости уровня поверхности жидкости ℎ*(x)* от координаты *x.* C помощью кубического сплайна и продемонстрировать ее на графике вместе с исходными узлами.

Таблица 1

Значения уровня поверхности вязкой жидкости

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 | 1 |
|  | 3.37 | 3.95 | 3.73 | 3.59 | 3.15 | 3.15 | 3.05 | 3.86 | 3.60 | 3.70 | 3.02 |

Продвинутая часть:

1. Разработать функцию *l\_i(i, x, x\_nodes),* которая возвращает значение *i*-го базисного полинома Лагранжа, заданного на узлах с абсциссами *x\_nodes*, в точке *x.*
2. Написать функцию *L(x, x\_nodes, y\_nodes),* которая возвращает значение интерполяционного полинома Лагранжа, заданного на узлах с абсциссами *x\_nodes* и ординатами *y\_nodes*, в точке *x*.
3. Известно, что при измерении координаты *x* всегда возникает погрешность, которая моделируется случайной величиной с нормальным распределением с нулевым математическим ожиданием и стандартным отклонением 0,01. Требуется провести следующий анализ, позволяющий выявить влияние этой погрешности на интерполяцию:
4. Сгенерировать 1000 векторов значений *[ x1z , … , x11z ]T*, предполагая, что *xiz = xi + Z*, где *xi* соответствует значению в таблице 1, и *Z* является случайной величиной с нормальным распределением с нулевым математическим ожиданием и стандартным отклонением 0,01.
5. Для каждого из полученных векторов построить интерполянт Лагранжа, предполагая, что в качестве абсцисс узлов используются значения *xiz*, а ординат – ℎ*i*  из таблицы 1. В результате вы должны иметь 1000 различных интерполянтов.
6. Предполагая, что все интерполянты представляют собой равновероятные события, построить такие функции *hl(x)* и *hu(x),* где *hl(x)* < *hu(x),* для любого *x* *∈ [0; 1]*, что вероятность того, что значение интерполянта в точке будет лежать в интервале *[hl(x)*; *hu(x)]*, равна 0.9.
7. Отобразить на едином графике функции *hl(x)* и *hu(x)*, усредненный интерполянт и узлы из таблицы 1.
8. Какие участки интерполянта и почему являются наиболее чувствительными к погрешностям?
9. Повторить анализ, описанный в предыдущем пункте, в предположении, что координаты вам известны точно, в то время как измерения уровня поверхности ℎ имеют ту же погрешность, что и в предыдущем пункте. Изменились ли выводы вашего анализа?
10. Повторить два предыдущие пункта для случая интерполяции кубическим сплайном. Какие выводы вы можете сделать, сравнив результаты анализа для интерполяции Лагранжа и интерполяции кубическим сплайном?

# Цель выполнения лабораторной работы

**Цель выполнения лабораторной работы** – написать алгоритмы интерполяции полиномом Лагранжа и кубическим сплайном, попытаться оценить устойчивость интерполяции при присутствии погрешности во входных данных.

# Выполненные задачи

1. Разработать алгоритм вычисления коэффициентов кубического сплайна по входным значениям узлов
2. Вычисление значений кубического сплайна на промежутке и его первой производной
3. Построение кубического сплайна
4. Интерполяция полиномом Лагранжа
5. Анализ влияния погрешности входных данных на интерполяционный полином Лагранжа
6. Анализ влияния погрешности входных данных на кубический сплайн

# Разработать алгоритм вычисления коэффициентов кубического сплайна

Из курса лекций мы знаем, что кубический сплайн имеет вид:

Где коэффициенты , , выражаются в следующем виде, где :

Самым «сложным» для вычислений является коэффициент , который вычисляется решение матричного уравнения вида:

Так как это уравнение вида , его решением является . Для получения обратной матрицы воспользовался функцией *np.linalg.inv*. А для вычисления скалярного произведения *numpy .dot().* После вычисления коэффициента нахождение остальных не является проблемой. В конце функции объединяю все коэффициенты в матрицу.

# Вычисление значений кубического сплайна на промежутке и его первой производной

Так как коэффициенты кубического сплайна будут использоваться только для получения значений функции, я принял решение изменить входные параметры на массив известных точек и массив точек интерполируемого промежутка.

Для вычисления значений кубического сплайна нам необходимо знать какой сплайн нам брать, для этого была создана функция *IndexForCubicSpline(x, xNodes)* которой на вход мы подаем значения точки *x* и массив известных точек по оси абсцисс *xNodes*, как результат мы получаем индекс нужного сплайна.

Теперь мы знаем всё для вычисления и вычисляем его по формуле [1.1](#Формула_1_1).

В программной реализации она представлена функцией *CubicSpline* в модуле *BasePart*

Вычислим первую производную функции:

В программной реализации она представлена функцией D*CubicSpline* в модуле *BasePart.*

# Построение кубического сплайна

Для его построения необходимо получить значения сплайна на промежутке. Я подал на вход функции *CubicSpline* из модуля *BasePart* значения из таблицы [1](#Таблица_1) и значения по оси абсцисс на промежутке с шагом . По полученным значения был построен график, представленный на рисунке [3.1](#Рисунок_3_1). Также был построен график первой производной кубического сплайна, представленный на рисунке [3.2](#Рисунок_3_2).

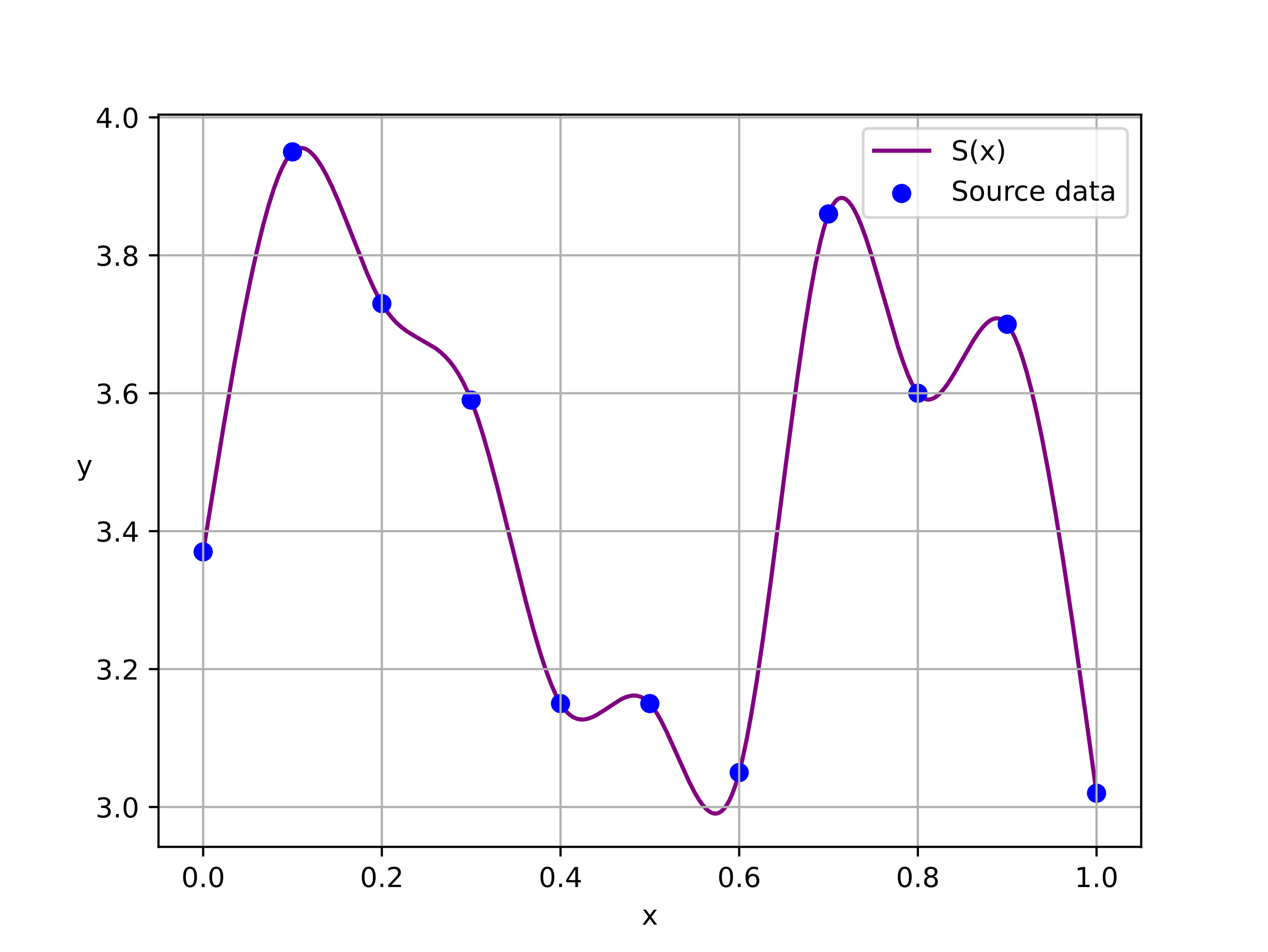


Рис. 3.1 – Естественный кубический сплайн, построенный на 11 точках из таблицы [1](#Таблица_1)

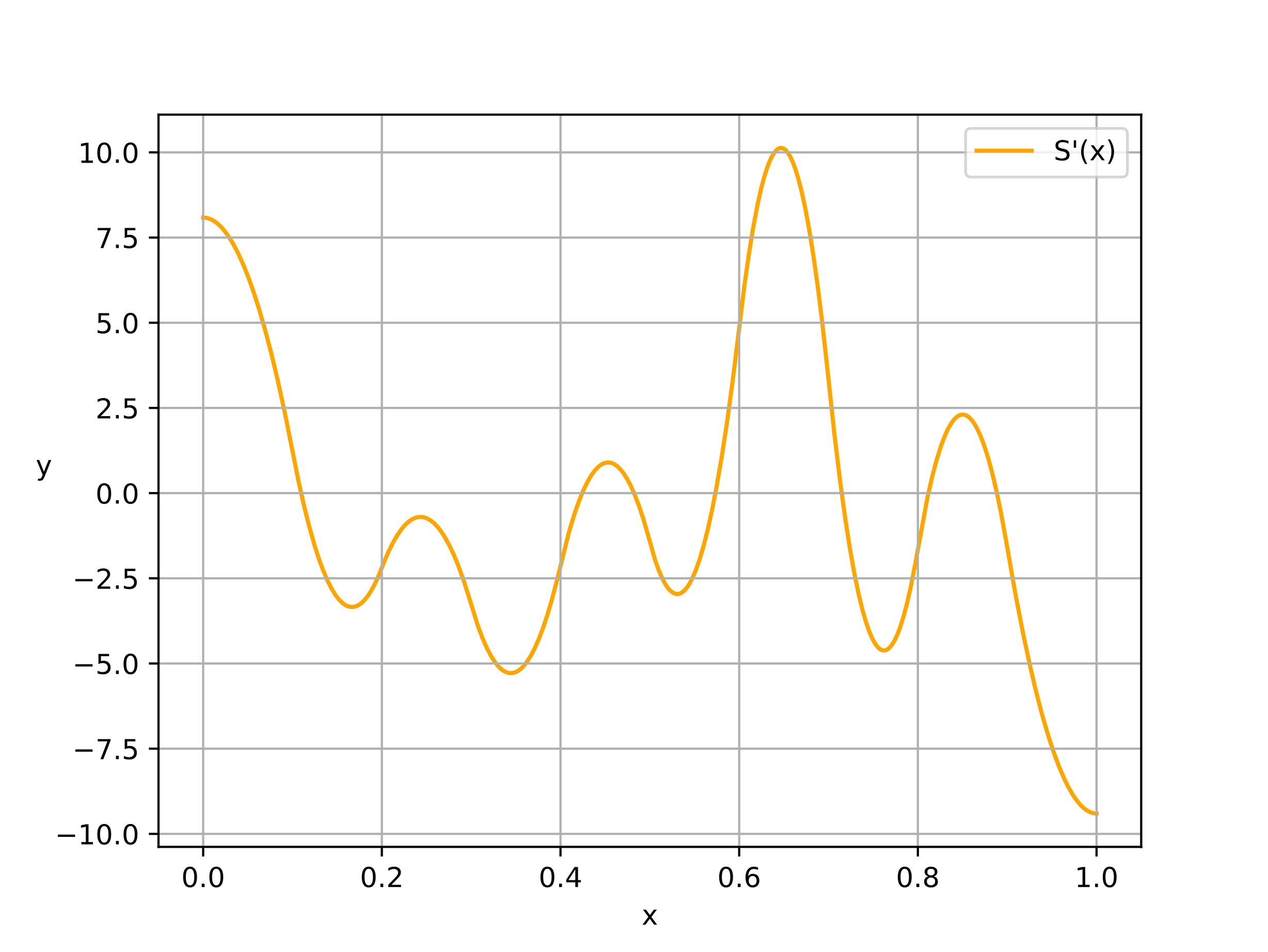


Рис 3.2. – Первая производная кубического сплайна, построенного на 11 точках из таблицы [1](#Таблица_1)

# Интерполяция полиномом Лагранжа

Интерполяционный многочлен Лагранжа представляет собой сумму произведений, представленных в виде:

В данной формуле являются базисными полиномами Лагранжа и представлены в виде:

Для вычисления базисных полиномов Лагранжа ([4.2](#Формула_4_2)) была создана функция *BasicLagrangePolynomial* в модуле *AdvancedPart.* Далее на её основе была создана функция *LagrangePolynomial* в том же модуле, которая на основе интерполируемого промежутка и значений исходных данных выдает массив значений полинома Лагранжа на интерполируемом промежутке.

График полинома Лагранжа, построенный по исходным данным таблицы [1](#Таблица_1) представлен на рисунке [4.1](#Рисунок_4_1). Можно наблюдать ярко выраженные осцилляции к концам отрезка.

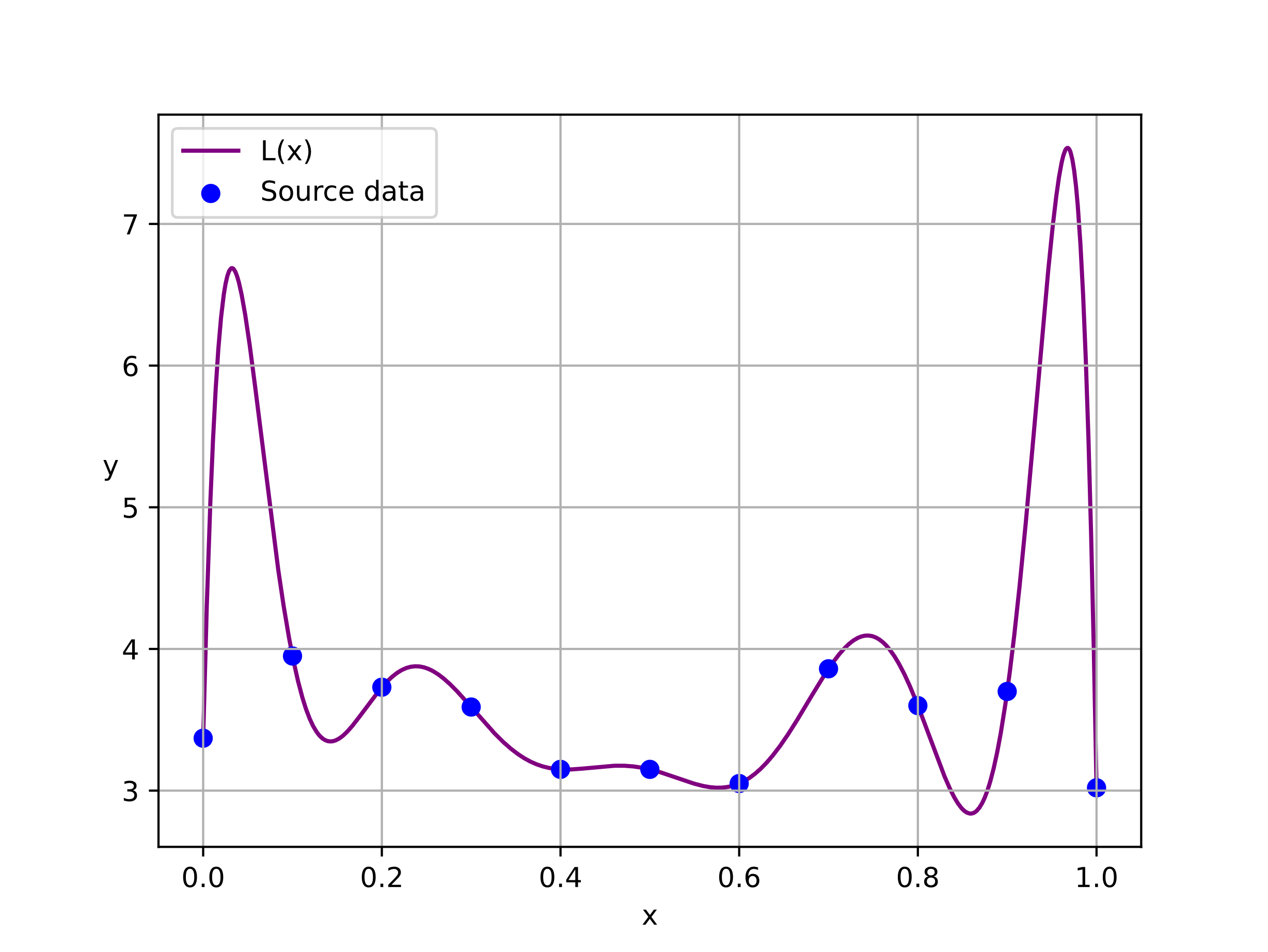


Рис. 4.1 – Полином Лагранжа 10 степени для исходных 11 точек таблицы [1](#Таблица_1)

# Анализ влияния погрешности входных данных на интерполяционный полином Лагранжа

По заданию необходимо сформировать 1000 векторов значений , где , где значения взяты из таблицы 1, а Z является нормально распределенной случайной величиной с нулевым математическим ожиданием и стандартным отклонением . Аналогичные вектора значений необходимо сделать для значений по оси ординат.

Для формирования данных векторов значений была реализована функция *GenerateSetOfErroneousData* в модуле *AdvancedPart*, которой на вход подается вектор значений, которому необходимо добавить погрешность, количество векторов для формирования, математическое ожидание и стандартное отклонение.

Для вычисления полиномов воспользовался функцией *LagrangePolynomial* в модуле *AdvancedPart*.

После вычисления было построено 1000 полиномов Лагранжа с ошибкой по оси абсцисс и ординат, представленных на рисунке [5.1](#Рисунок_5_1) и [5.2](#Рисунок_5_2).

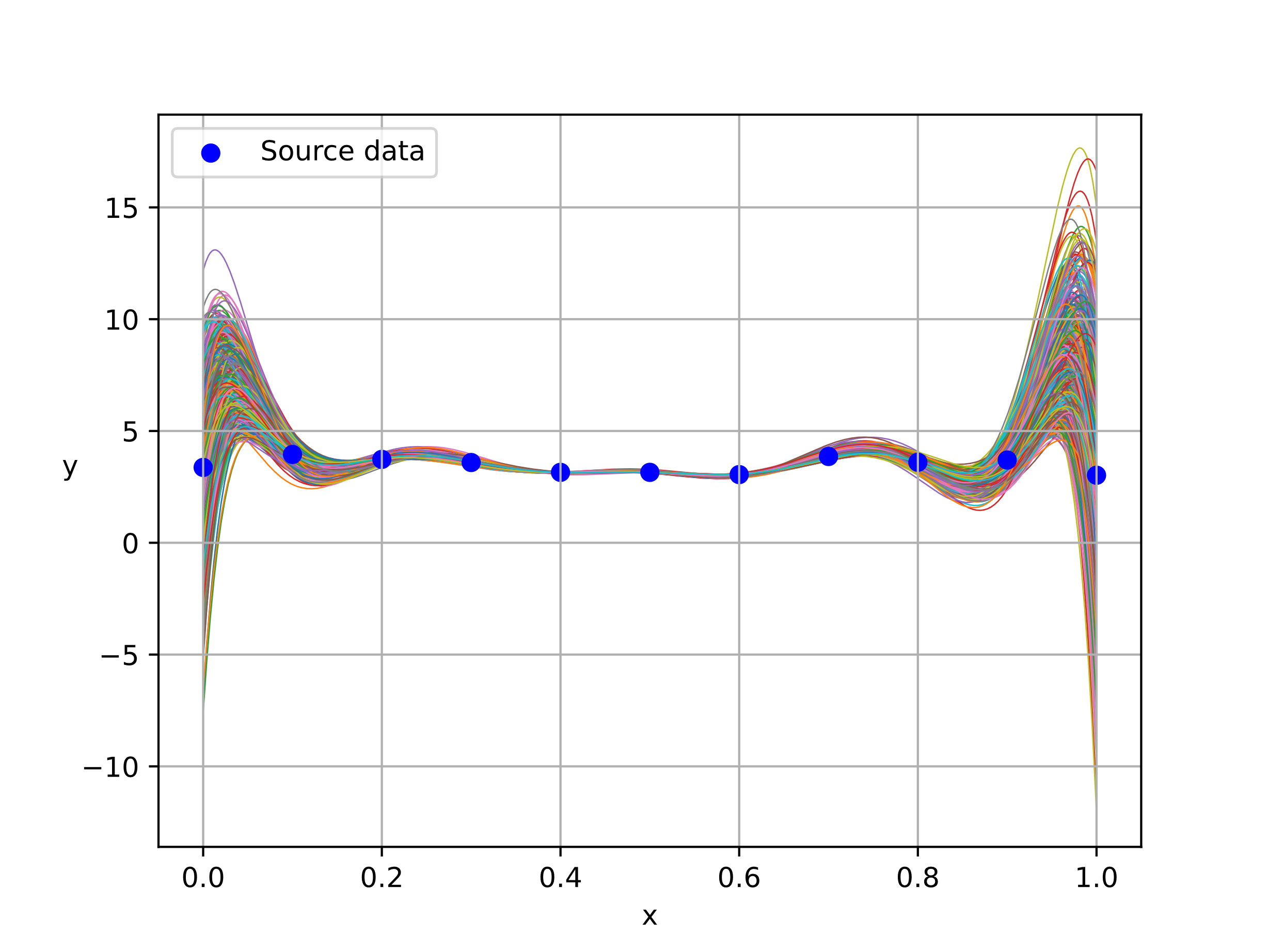


Рис. 5.1 – 1000 полиномов Лагранжа, построенные с погрешностью входных данных из таблицы [1](#Таблица_1) с ошибкой по оси абсцисс

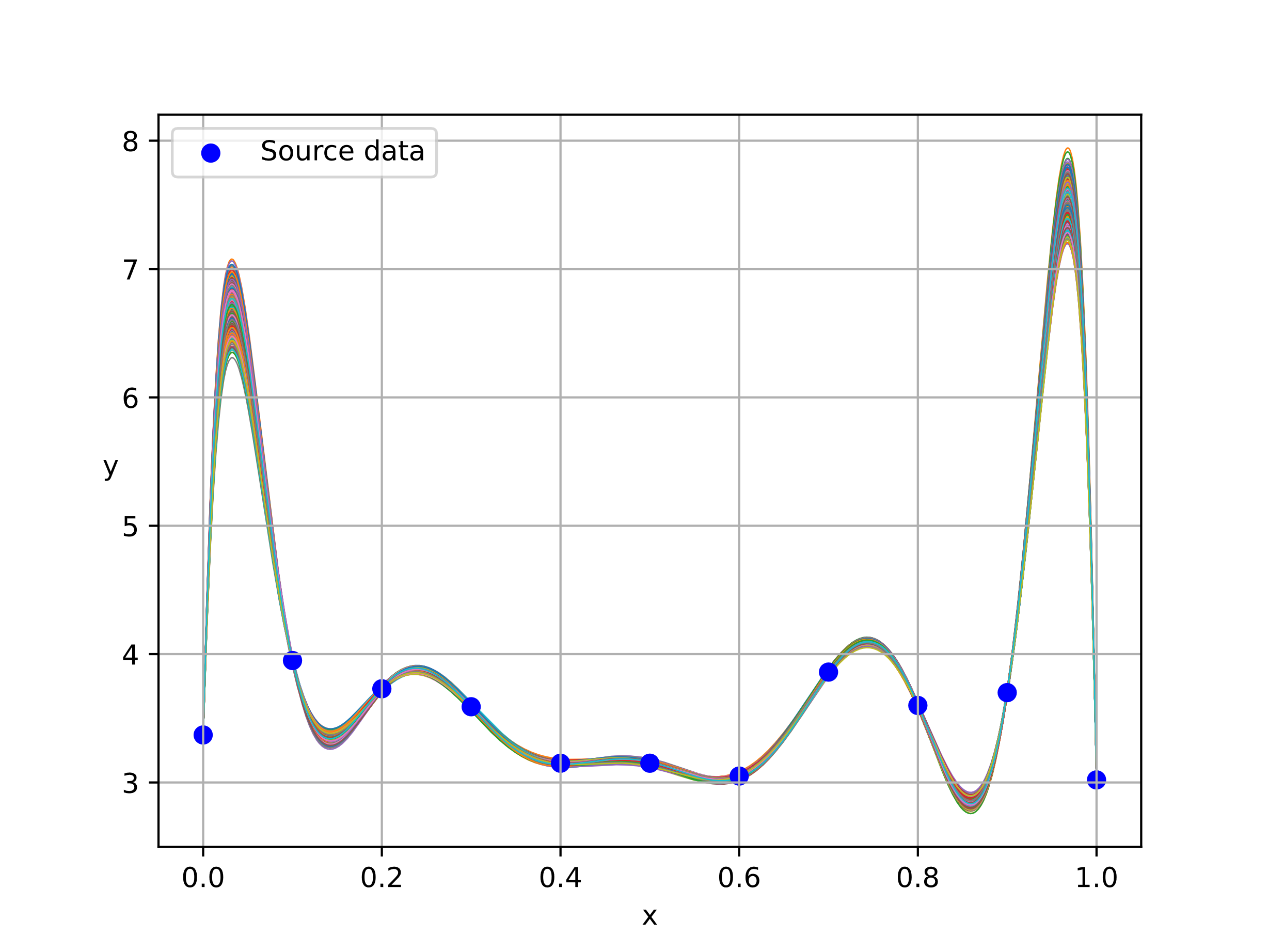


Рис. 5.2 – 1000 полиномов Лагранжа, построенные с погрешностью входных данных из таблицы [1](#Таблица_1) с ошибкой по оси ординат

Далее я провел анализ данных графиков, предполагая, что каждый интерполянт является равновероятным событием, построю функции , где . – нижняя кривая доверительный полосы.– верхняя кривая доверительной полосы. – усредненный интерполянт.

Пусть у нас всего сплайнов и вероятность попадания в интервал равна , тогда вне этого интервала находятся всех сплайнов, а выше и ниже находятся всех сплайнов.

Для нахождения данных кривых была создана функция *DeterminationOfConfidenceCurves* (листинг [5.1](#Листинг_5_1)), на вход ей подаются интерполянты и вероятность попадания в доверительную полосу, как результат получая 3 массива значений кривых .

Листинг 5.1 – функция определения кривых

1. def DeterminationOfConfidenceCurves(inputSplines, percent):
2. lenInput = len(inputSplines)
3. h\_lPosition = int((lenInput - (lenInput \* percent)) / 2) - 1
4. h\_uPosition = lenInput - h\_lPosition - 2
5. inputSplines = np.sort(inputSplines, axis=0)
6. h\_u = inputSplines[h\_uPosition]
7. h\_l = inputSplines[h\_lPosition]
8. h\_m = np.mean(inputSplines, axis=0)
9. return np.array((h\_u, h\_m, h\_l))

Результаты данных вычислений представлены на рисунке [5.3](#Рисунок_5_3) для погрешности по оси абсцисс и рисунке [5.](#Рисунок_5_4)4 для погрешности по оси ординат.

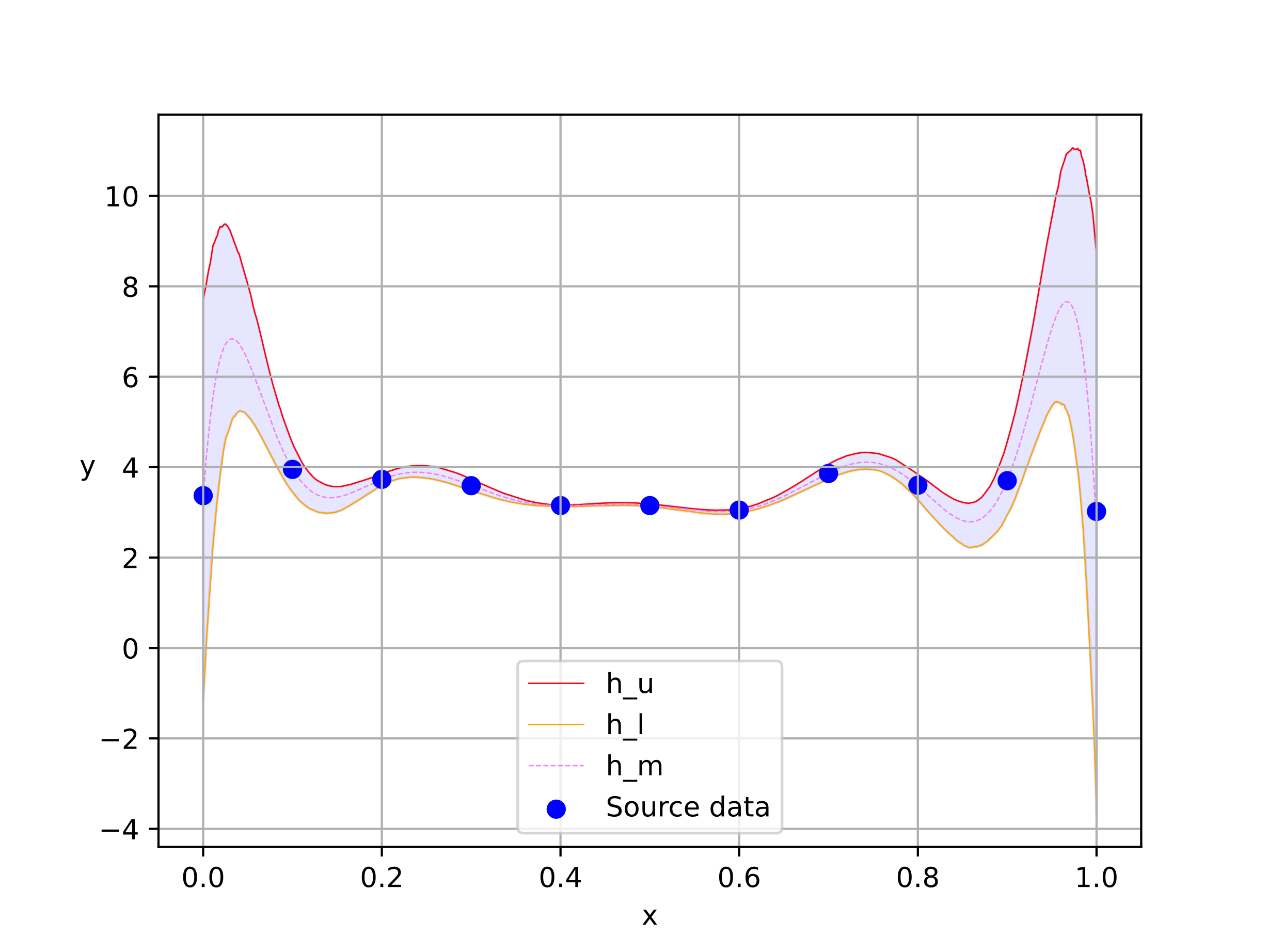


Рис. 5.3 – Доверительная полоса и усредненный интерполянт Лагранжа при погрешности по оси абсцисс

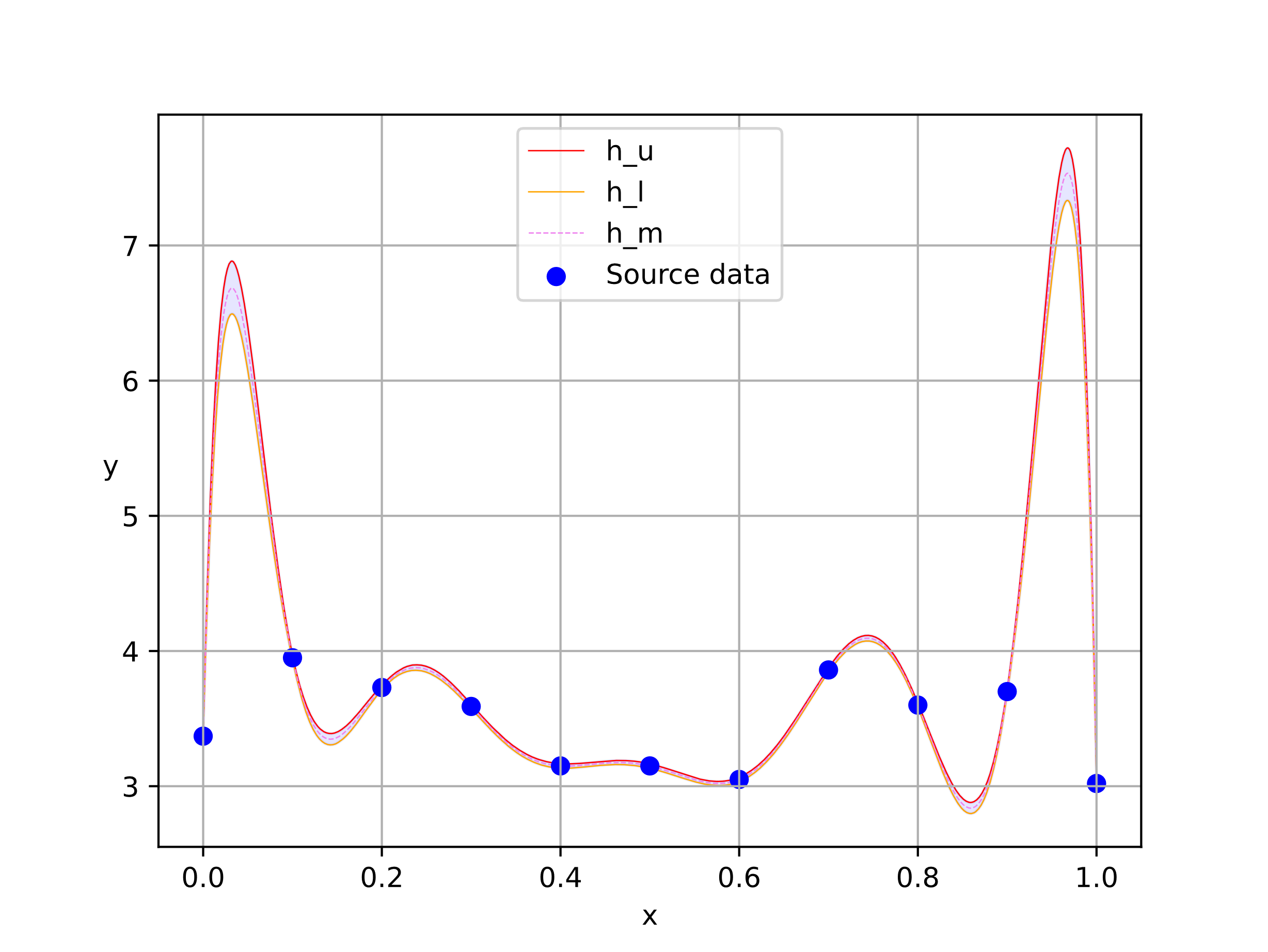


Рис. 5.4 – Доверительная полоса и усредненный интерполянт Лагранжа при погрешности по оси ординат

Проведя анализ полученных графиков, можно сделать вывод, что наибольшее влияние имеет во время погрешности по оси абсцисс. Также можно заметить, что т.к. мы можем наблюдать «паразитные» осцилляции на краях интерполируемого отрезка (их можно уменьшить использую узлы Чебышева), как раз на этих осцилляциях и видно наиболее сильное влияние отклонения от усредненного интерполянта.

# Анализ влияния погрешности входных данных на кубический сплайн

Следующим задание было провести аналогичный анализ для 1000 кубических сплайнов, используя вектора значений из пункта [5](#_Анализ_влияния_погрешности). Графики представлены на рисунке [6.1](#Рисунок_6_1) с ошибкой по оси абсцисс и [6.2](#Рисунок_6_2) с ошибкой по оси ординат.

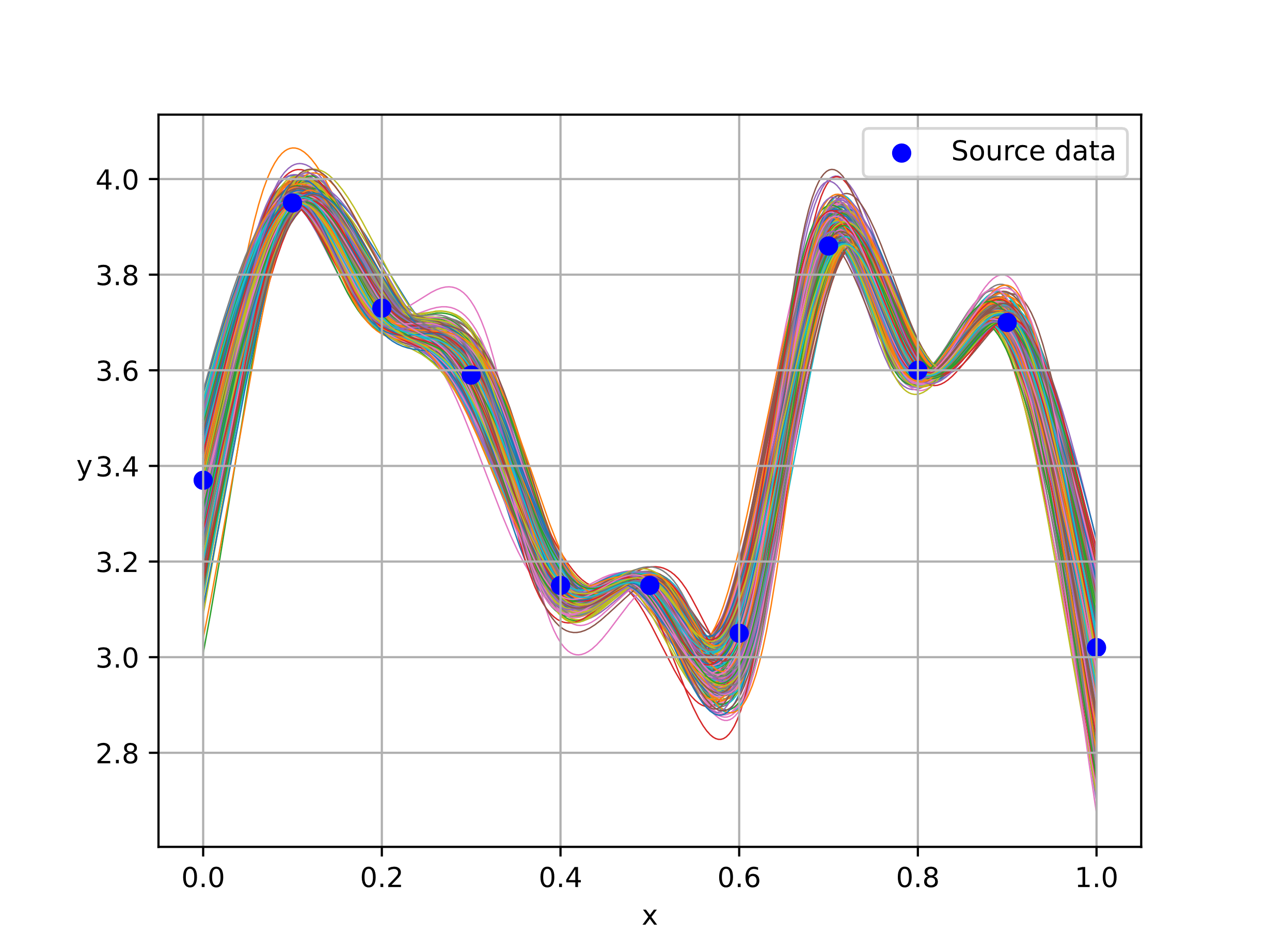


Рис. 6.1 – 1000 интерполяций кубическим сплайном, построенные с погрешностью входных данных из таблицы [1](#Таблица_1) с ошибкой по оси абсцисс

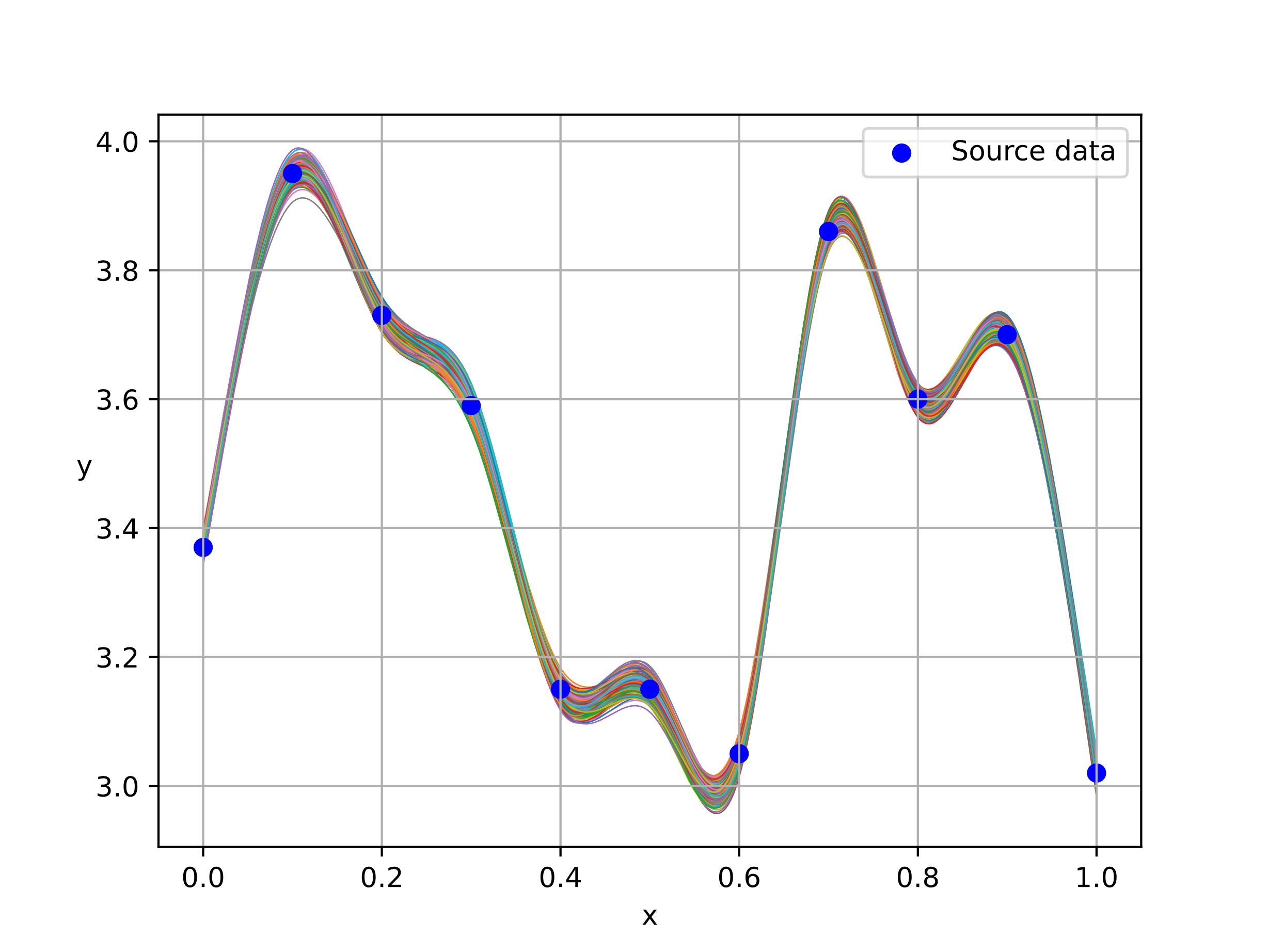


Рис. 6.2 – 1000 интерполяций кубическим сплайном, построенные с погрешностью входных данных из таблицы [1](#Таблица_1) с ошибкой по оси ординат

Результаты анализа представлены на рисунках [6.](#Рисунок_6_3)3 и [6.](#Рисунок_6_4)4, с погрешностью по оси абсцисс и оси ординат соответственно.

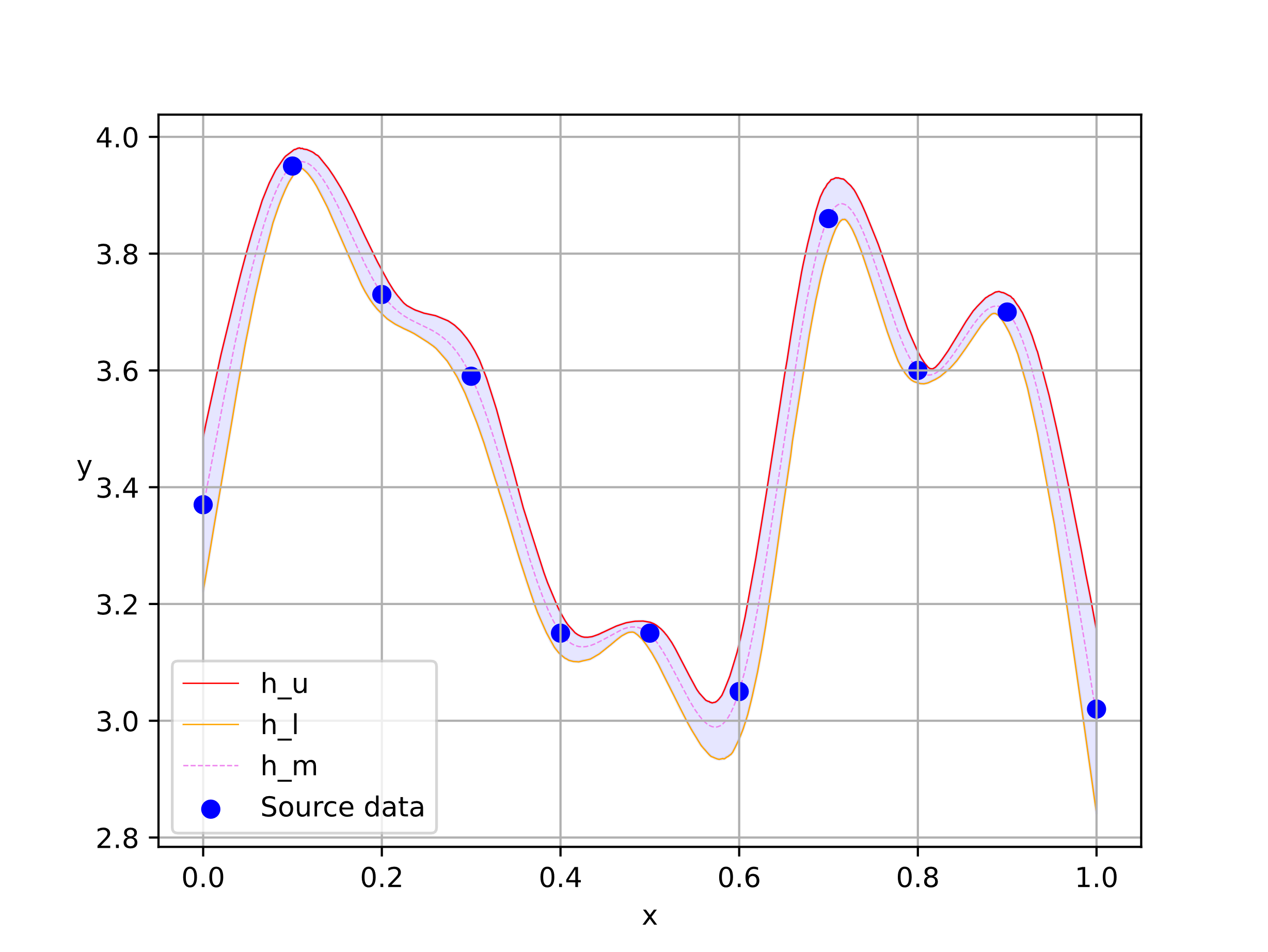


Рис. 6.3 – Доверительная полоса и усредненный интерполянт кубического при погрешности по оси абсцисс

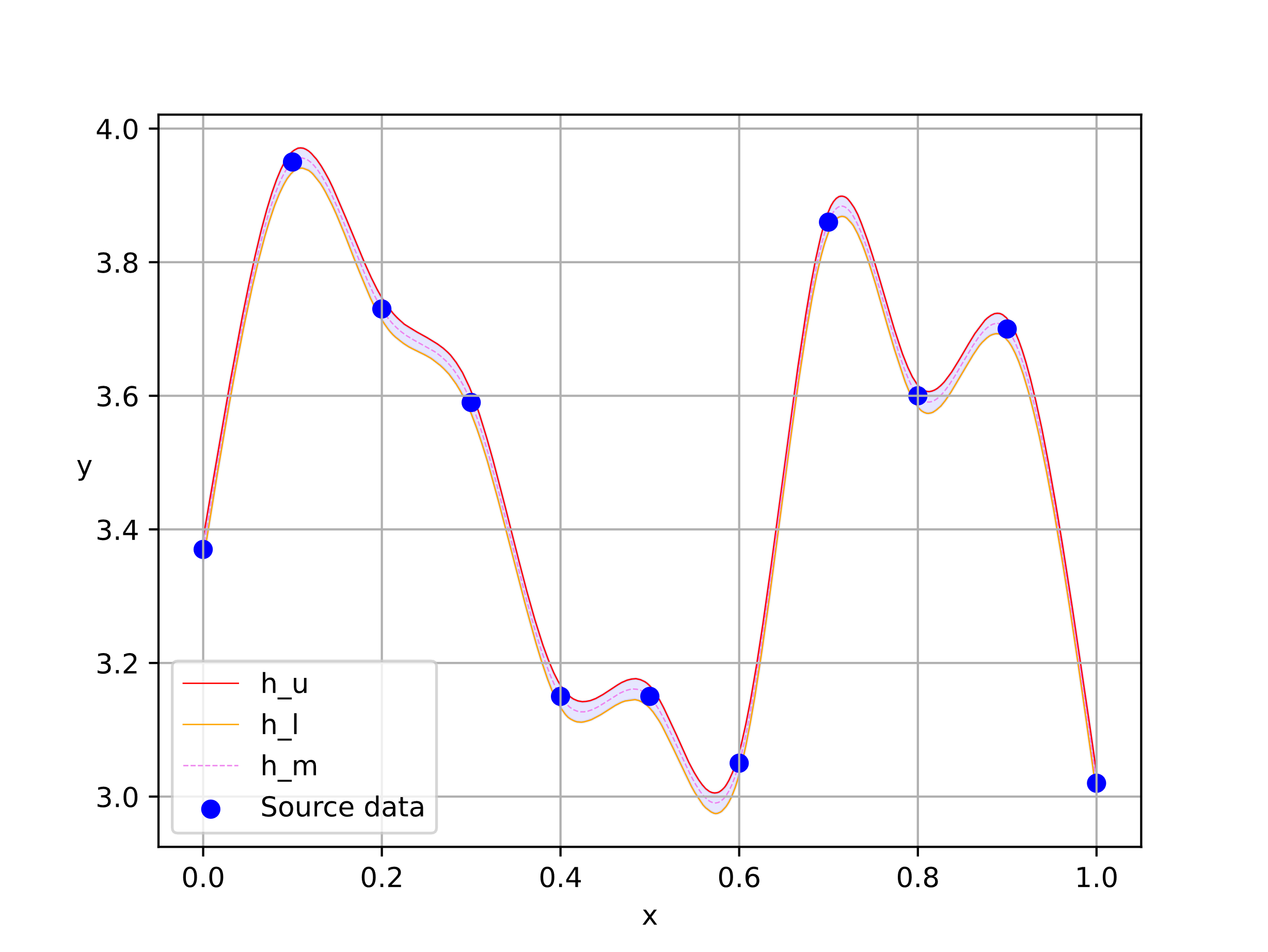


Рис. 6.4 – Доверительная полоса и усредненный интерполянт кубического сплайна при погрешности по оси ординат

Проанализировав эти графики, я заметил, что не видно на всей длине доверительной полосы явных расширений, но сравнивая доверительные полосы, можно заметить, что при погрешности по оси абсцисс доверительная полоса шире, чем при погрешности по оси ординат.

# Заключение

По результатам можно сделать вывод, что при погрешности по оси абсцисс в случае кубического сплайна и полинома Лагранжа, оказывают большее влияние на результат, чем погрешности по оси ординат.

Самыми уязвимыми участками при погрешности при интерполировании полиномом Лагранжа являются края интерполируемого участка, скорее всего по причине «паразитических» осцилляций.

Для кубического сплайна самыми уязвимыми местами являются экстремумы, хотя доверительная полоса намного постоянна, чем доверительная полоса полинома Лагранжа.

# Список использованных источников

1. Першин А.Ю. Лекции по курсу «Вычислительная математика. Москва, 2018-2021, С. 140.