**БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ**

**Кафедра теории вероятностей и математической статистики**

**ОТЧЕТ**

по лабораторной работе № 4

«Исследование статистических свойств оценок семивариограммы»

учебной дисциплины

«Математические методы анализа данных»

Вариант № 1

**Выполнили:**

Крученков Евгений Андреевич,

Кендысь Алексей Максимович,

3 курс, 7а группа, специальность «прикладная математика»

**Преподаватель:**

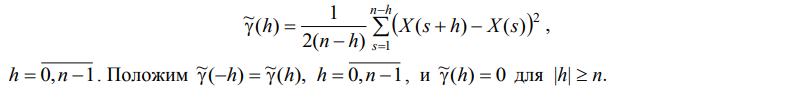
Цеховая Татьяна Вячеславовна,

кандидат физико-математических наук, доцент

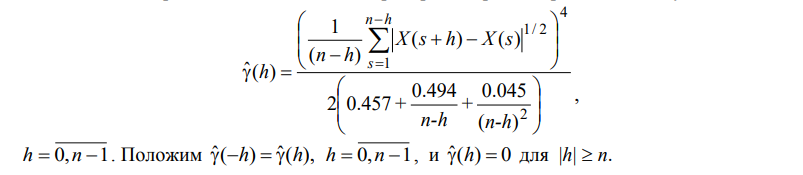
Минск, 2023

**Постановка задачи**

Исследовать статистические свойства классической и робастной оценок семивариограммы гауссовского случайного процесса. Пусть последовательных, полученных через равные промежутки времени наблюдений за случайным процессом. В качестве классической оценки семивариограммы рассмотрим статистику вида:



В качестве робастной оценки семивариограммы рассмотрим статистику:



1) Вычислить оценки семивариограммы и представить их графически.

**Необходимо:**

• Смоделировать , наблюдений за гауссовским стационарным случайным процессом с непрерывным временем и известной ковариационной функцией (можно воспользоваться результатами лабораторной работы 1).

• Найти вид семивариограммы процесса. Построить графики наблюдений, ковариационной функции и семивариограммы.

• Вычислить классическую и робастную оценки семивариограммы. На одном графике представить классическую, робастную оценки и истинную семивариограмму.

• Вычислить отклонения и для . Сделать вывод. Отобразить графически.

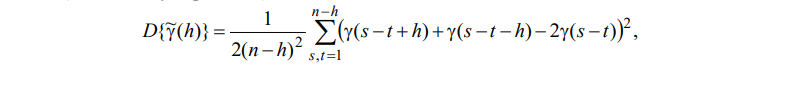
2) Исследовать оценки семивариограммы на состоятельность в среднеквадратическом смысле.

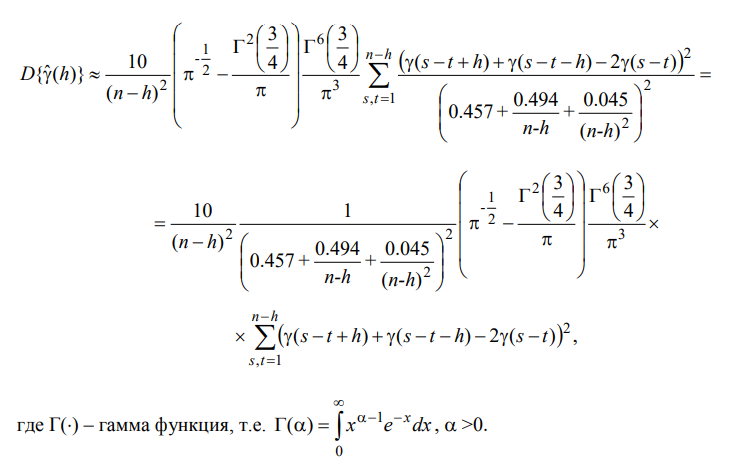
**Необходимо:**

• Смоделировать 10 наборов наблюдений за гауссовским стационарным случайным процессом с непрерывным временем и известной ковариационной функцией. Число наблюдений n, соответствующих каждому набору, выбрать из таблицы 5 согласно варианту.



• Вычислить значения дисперсий классической и робастной оценок для требуемых n и h (см. таблицу вариантов).





• Для каждого фиксированного h из таблицы вариантов построить графики зависимостей дисперсий и от количества наблюдений n, представить таблицы соответствующих зависимостей. Сделать сравнительный анализ и вывод о состоятельности оценок.

**Исходные данные (алгоритм выполнения)**

В качестве ковариационной функции берётся ковариационная функция из 1-ой лабораторной работы:

Моделирование случайного процесса с заданной ковариационной функцией:

=1

Вычислим семивариограмму:

В лабораторной представлены результаты для случая .

**Результат**

**Задание 1. Вычисление оценок семивариограммы.**

Задание 1 было выполнено в Excel. В качестве параметров и выбираются В таблице 1 приведены: сгенерированный вектор из нормального распределения, полученные наблюдения за случайным процессом, значения ковариационной функции и семивариограммы.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Количество наблюдений, n** | **Нормальное стандартное распределение, xn** | **Случайный процесс** | **Значения ковариационной функции** | **Значения семивариограммы** | **t** |
| 1 | -0.50762992 | -0.857074 | 3 | 0 | 0 |
| 2 | 2.65223207 | 4.28674577 | 0.66939048 | 2.33060952 | 1 |
| 3 | -0.258311275 | 0.52037378 | 0.149361205 | 2.850638795 | 2 |
| 4 | 0.168283805 | 0.40023868 | 0.03332699 | 2.96667301 | 3 |
| 5 | -1.5703381 | -2.5620276 | 0.007436257 | 2.992563743 | 4 |
| 6 | 0.594097855 | 0.43139938 | 0.001659253 | 2.998340747 | 5 |
| 7 | -0.225148824 | -0.2838793 | 0.000370229 | 2.999629771 | 6 |
| 8 | -0.028576324 | -0.1115898 | 8.26093E-05 | 2.999917391 | 7 |
| 9 | 0.724392066 | 1.19815254 | 1.84326E-05 | 2.999981567 | 8 |
| 10 | -1.160021839 | -1.6912177 | 4.11288E-06 | 2.999995887 | 9 |
| 11 | 0.956868007 | 1.23819845 | 9.17707E-07 | 2.999999082 | 10 |
| 12 | 0.962324975 | 1.90105302 | 2.04768E-07 | 2.999999795 | 11 |
| 13 | 0.15626938 | 0.6880249 | 4.56899E-08 | 2.999999954 | 12 |
| 14 | -0.388986336 | -0.503239 | 1.01948E-08 | 2.99999999 | 13 |
| 15 | -2.028773451 | -3.5376354 | 2.27477E-09 | 2.999999998 | 14 |
| 16 | 0.048710262 | -0.7071115 | 5.07569E-10 | 2.999999999 | 15 |
| 17 | -1.08752829 | -1.9939427 | 1.13254E-10 | 3 | 16 |
| 18 | -0.110496785 | -0.6314697 | 2.52704E-11 | 3 | 17 |
| 19 | 1.506887202 | 2.40330353 | 5.63859E-12 | 3 | 18 |
| 20 | 1.009589141 | 2.24082313 | 1.25814E-12 | 3 | 19 |
| 21 | 0.453954954 | 1.26644527 | 2.80729E-13 | 3 | 20 |
| 22 | -0.750496838 | -0.9845443 | 6.2639E-14 | 3 | 21 |
| 23 | 0.696527422 | 0.95632386 | 1.39767E-14 | 3 | 22 |
| 24 | -1.053874712 | -1.56596 | 3.11861E-15 | 3 | 23 |
| 25 | 1.504035936 | 2.18997654 | 6.95857E-16 | 3 | 24 |
| 26 | 0.044958597 | 0.56455717 | 1.55267E-16 | 3 | 25 |
| 27 | 2.227097866 | 3.88616499 | 3.46447E-17 | 3 | 26 |
| 28 | -0.763536718 | -0.4220222 | 7.73027E-18 | 3 | 27 |
| 29 | -0.98121518 | -1.7508334 | 1.72486E-18 | 3 | 28 |
| 30 | -0.945683496 | -1.9873401 | 3.84868E-19 | 3 | 29 |
| 31 | 0.969153007 | 1.19286641 | 8.58756E-20 | 3 | 30 |
| 32 | 0.644796501 | 1.35482824 | 1.91614E-20 | 3 | 31 |
| 33 | 1.981934474 | 3.64856848 | 4.27549E-21 | 3 | 32 |
| 34 | -2.32255843 | -3.1072636 | 9.53991E-22 | 3 | 33 |
| 35 | -0.363368144 | -1.306829 | 2.12864E-22 | 3 | 34 |
| 36 | -0.138714995 | -0.5257971 | 4.74964E-23 | 3 | 35 |
| 37 | 1.047365004 | 1.65103259 | 1.05979E-23 | 3 | 36 |
| 38 | 1.610715117 | 3.08789997 | 2.36471E-24 | 3 | 37 |
| 39 | -0.659748594 | -0.424905 | 5.27638E-25 | 3 | 38 |
| 40 | -0.155108637 | -0.356692 | 1.17732E-25 | 3 | 39 |
| 41 | -0.186780653 | -0.3949461 | 2.62695E-26 | 3 | 40 |
| 42 | 0.686035264 | 1.07016622 | 5.86152E-27 | 3 | 41 |
| 43 | 0.473205546 | 1.03773879 | 1.30788E-27 | 3 | 42 |
| 44 | -0.484013754 | -0.58565 | 2.91828E-28 | 3 | 43 |
| 45 | -0.804946012 | -1.4897337 | 6.51157E-29 | 3 | 44 |
| 46 | 0.024060682 | -0.2917809 | 1.45293E-29 | 3 | 45 |
| 47 | -1.661155693 | -2.869773 | 3.24192E-30 | 3 | 46 |
| 48 | -0.192544576 | -0.965422 | 7.2337E-31 | 3 | 47 |
| 49 | 1.420294211 | 2.18258653 | 1.61406E-31 | 3 | 48 |
| 50 | -0.656423254 | -0.6212933 | 3.60145E-32 | 3 | 49 |
| 51 | -0.997185907 | -1.8222615 | 8.03591E-33 | 3 | 50 |
| 52 | -1.55606358 | -3.0338335 | 1.79305E-33 | 3 | 51 |
| 53 | -1.613238965 | -3.4007058 | 4.00084E-34 | 3 | 52 |
| 54 | 1.288485691 | 1.41665794 | 8.92709E-35 | 3 | 53 |
| 55 | 0.337175834 | 0.88538122 | 1.9919E-35 | 3 | 54 |
| 56 | 0.989916771 | 1.86891438 | 4.44454E-36 | 3 | 55 |
| 57 | 1.053342658 | 2.19545751 | 9.9171E-37 | 3 | 56 |
| 58 | 1.288835847 | 2.66592195 | 2.2128E-37 | 3 | 57 |
| 59 | 2.133310772 | 4.19669427 | 4.93743E-38 | 3 | 58 |
| 60 | -1.562511898 | -1.7017102 | 1.10169E-38 | 3 | 59 |
| 61 | -0.30968863 | -0.902576 | 2.4582E-39 | 3 | 60 |
| 62 | 0.652917151 | 0.9009826 | 5.48499E-40 | 3 | 61 |
| 63 | -0.524408961 | -0.684367 | 1.22387E-40 | 3 | 62 |
| 64 | -0.466370693 | -0.9401155 | 2.73082E-41 | 3 | 63 |
| 65 | 0.684583483 | 0.94607132 | 6.09328E-42 | 3 | 64 |
| 66 | 0.73527417 | 1.45252181 | 1.35959E-42 | 3 | 65 |
| 67 | 1.231635451 | 2.40357438 | 3.03366E-43 | 3 | 66 |
| 68 | -0.896918664 | -0.9780327 | 6.76902E-44 | 3 | 67 |
| 69 | -0.466370693 | -1.0056412 | 1.51037E-44 | 3 | 68 |
| 70 | -0.335394361 | -0.7906632 | 3.3701E-45 | 3 | 69 |
| 71 | 0.98269993 | 1.48275353 | 7.5197E-46 | 3 | 70 |
| 72 | -1.000082648 | -1.357676 | 1.67787E-46 | 3 | 71 |
| 73 | 0.899208317 | 1.21527001 | 3.74384E-47 | 3 | 72 |
| 74 | -1.455127858 | -2.1856505 | 8.35363E-48 | 3 | 73 |
| 75 | 0.981090125 | 1.16877183 | 1.86395E-48 | 3 | 74 |
| 76 | 0.767433903 | 1.55651097 | 4.15903E-49 | 3 | 75 |
| 77 | -0.724392066 | -0.8757471 | 9.28005E-50 | 3 | 76 |
| 78 | 0.194103222 | 0.13231509 | 2.07066E-50 | 3 | 77 |
| 79 | -0.653390089 | -1.0736495 | 4.62026E-51 | 3 | 78 |
| 80 | -1.386501935 | -2.5805106 | 1.03092E-51 | 3 | 79 |
| 81 | 0.112344196 | -0.3861097 | 2.30029E-52 | 3 | 80 |
| 82 | -0.742107886 | -1.3391154 | 5.13265E-53 | 3 | 81 |
| 83 | 0.826714768 | 1.09701453 | 1.14525E-53 | 3 | 82 |
| 84 | -0.681975507 | -0.9066592 | 2.5554E-54 | 3 | 83 |
| 85 | 1.29217824 | 1.9793894 | 5.70186E-55 | 3 | 84 |
| 86 | -2.285578375 | -3.4172713 | 1.27226E-55 | 3 | 85 |
| 87 | 0.424049631 | -0.0465379 | 2.83879E-56 | 3 | 86 |
| 88 | 0.444308625 | 0.73977934 | 6.33419E-57 | 3 | 87 |
| 89 | 0.05200377 | 0.25286939 | 1.41335E-57 | 3 | 88 |
| 90 | 0.116888259 | 0.25377499 | 3.15361E-58 | 3 | 89 |
| 91 | 0.491422725 | 0.88633487 | 7.03665E-59 | 3 | 90 |
| 92 | -0.558869715 | -0.7458184 | 1.57009E-59 | 3 | 91 |
| 93 | 1.095440894 | 1.68310974 | 3.50334E-60 | 3 | 92 |
| 94 | -0.982824986 | -1.2838329 | 7.81702E-61 | 3 | 93 |
| 95 | -0.690299657 | -1.4519524 | 1.74421E-61 | 3 | 94 |
| 96 | -0.284583166 | -0.8044599 | 3.89186E-62 | 3 | 95 |
| 97 | -0.53541271 | -1.0834812 | 8.68392E-63 | 3 | 96 |
| 98 | 0.110651399 | -0.0549353 | 1.93764E-63 | 3 | 97 |
| 99 | -0.173330363 | -0.3049059 | 4.32347E-64 | 3 | 98 |
| 100 | 0.620500487 | 0.97960908 | 9.64696E-65 | 3 | 99 |

Табл. 1 – результаты моделирования

На рисунке 1 представлен график наблюдений, а на рисунке 2 – график ковариационной функции и семивариограммы.

Рис. 1 – график наблюдений

Рис. 2 – график ковариационной функции и семивариограммы

Далее были вычислены классическая и робастная оценки семивариограммы, а также их отклонения от истинных значений. Результаты приведены в таблице 2.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **h** | **Классическая оценка семивариограммы** | **Робастная оценка семивариограммы** | **Отклонения для классической оценки** | **Отклонения для робастной оценки** |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 2.635527361 | 2.447345874 | -0.30491784 | -0.116736355 |
| 2 | 2.813179098 | 2.699396008 | 0.037459696 | 0.151242787 |
| 3 | 3.727741376 | 4.063390328 | -0.76106837 | -1.096717317 |
| 4 | 3.27159136 | 3.803798564 | -0.27902762 | -0.811234821 |
| 5 | 2.764323311 | 2.606078991 | 0.234017436 | 0.392261756 |
| 6 | 2.710027782 | 2.920752832 | 0.289601988 | 0.078876938 |
| 7 | 3.018723483 | 3.166818522 | -0.01880609 | -0.166901132 |
| 8 | 2.836493209 | 2.909420162 | 0.163488359 | 0.090561405 |
| 9 | 3.257526273 | 3.387430921 | -0.25753039 | -0.387435034 |
| 10 | 2.892540577 | 2.562937717 | 0.107458505 | 0.437061365 |
| 11 | 3.122165849 | 3.215941459 | -0.12216605 | -0.215941664 |
| 12 | 2.855658255 | 2.574331112 | 0.144341699 | 0.425668843 |
| 13 | 3.279396682 | 3.036614986 | -0.27939669 | -0.036614996 |
| 14 | 3.480294602 | 3.766558803 | -0.4802946 | -0.766558805 |
| 15 | 3.66592987 | 3.6532159 | -0.66592987 | -0.6532159 |
| 16 | 2.767170421 | 3.371695612 | 0.232829578 | -0.371695612 |
| 17 | 2.247708882 | 2.476578114 | 0.752291118 | 0.523421886 |
| 18 | 2.679785181 | 2.432486026 | 0.320214819 | 0.567513974 |
| 19 | 2.872403444 | 2.896400124 | 0.127596556 | 0.103599876 |
| 20 | 3.634388918 | 3.029449079 | -0.63438892 | -0.029449079 |
| 21 | 3.082846135 | 3.362662424 | -0.08284613 | -0.362662424 |
| 22 | 3.248691713 | 2.675494339 | -0.24869171 | 0.324505661 |
| 23 | 3.242185663 | 3.092662483 | -0.24218566 | -0.092662483 |
| 24 | 2.943486999 | 3.181357591 | 0.056513001 | -0.181357591 |
| 25 | 3.143197602 | 3.316504531 | -0.1431976 | -0.316504531 |
| 26 | 3.078442336 | 3.257590377 | -0.07844234 | -0.257590377 |
| 27 | 3.211416388 | 3.280997838 | -0.21141639 | -0.280997838 |
| 28 | 3.611787589 | 3.246725243 | -0.61178759 | -0.246725243 |
| 29 | 2.359751772 | 2.184546232 | 0.640248228 | 0.815453768 |
| 30 | 2.781068786 | 3.405965394 | 0.218931214 | -0.405965394 |
| 31 | 2.836310011 | 3.035318056 | 0.163689989 | -0.035318056 |
| 32 | 3.462676308 | 3.334376488 | -0.46267631 | -0.334376488 |
| 33 | 3.078180506 | 2.458430633 | -0.07818051 | 0.541569367 |
| 34 | 2.338050724 | 1.881169634 | 0.661949276 | 1.118830366 |
| 35 | 2.481414689 | 2.606128379 | 0.518585311 | 0.393871621 |
| 36 | 2.724744834 | 2.701472297 | 0.275255166 | 0.298527703 |
| 37 | 3.306659151 | 3.303424814 | -0.30665915 | -0.303424814 |
| 38 | 2.811457072 | 3.255438951 | 0.188542928 | -0.255438951 |
| 39 | 3.017571346 | 3.185149899 | -0.01757135 | -0.185149899 |
| 40 | 3.00712704 | 3.742694302 | -0.00712704 | -0.742694302 |
| 41 | 3.833307486 | 4.289027325 | -0.83330749 | -1.289027325 |
| 42 | 3.264957897 | 2.993480154 | -0.2649579 | 0.006519846 |
| 43 | 3.184889857 | 3.387302822 | -0.18488986 | -0.387302822 |
| 44 | 3.17769877 | 2.898333585 | -0.17769877 | 0.101666415 |
| 45 | 2.802488315 | 2.777615831 | 0.197511685 | 0.222384169 |
| 46 | 2.457049198 | 2.461562351 | 0.542950802 | 0.538437649 |
| 47 | 2.71629952 | 2.544735066 | 0.28370048 | 0.455264934 |
| 48 | 2.877542714 | 2.7303794 | 0.122457286 | 0.2696206 |
| 49 | 3.727798327 | 3.990543349 | -0.72779833 | -0.990543349 |
| 50 | 2.752550479 | 2.524542225 | 0.247449521 | 0.475457775 |
| 51 | 3.239891124 | 3.202555266 | -0.23989112 | -0.202555266 |
| 52 | 2.565635451 | 2.396098845 | 0.434364549 | 0.603901155 |
| 53 | 3.11599028 | 2.966397026 | -0.11599028 | 0.033602974 |
| 54 | 2.886805121 | 3.051647085 | 0.113194879 | -0.051647085 |
| 55 | 2.536207651 | 2.65520801 | 0.463792349 | 0.34479199 |
| 56 | 2.247277761 | 2.426706787 | 0.752722239 | 0.573293213 |
| 57 | 2.575906234 | 2.597657391 | 0.424093766 | 0.402342609 |
| 58 | 2.734914939 | 3.041478789 | 0.265085061 | -0.041478789 |
| 59 | 3.237550003 | 2.975387879 | -0.23755 | 0.024612121 |
| 60 | 2.449324396 | 2.25964543 | 0.550675604 | 0.74035457 |
| 61 | 3.535232832 | 4.02254508 | -0.53523283 | -1.02254508 |
| 62 | 2.492406798 | 1.894857012 | 0.507593202 | 1.105142988 |
| 63 | 2.532595666 | 2.699326055 | 0.467404334 | 0.300673945 |
| 64 | 1.908917994 | 2.198366641 | 1.091082006 | 0.801633359 |
| 65 | 2.242031922 | 2.444078236 | 0.757968078 | 0.555921764 |
| 66 | 3.00636391 | 3.411051517 | -0.00636391 | -0.411051517 |
| 67 | 2.638172594 | 2.631419385 | 0.361827406 | 0.368580615 |
| 68 | 2.909000055 | 2.787832576 | 0.090999945 | 0.212167424 |
| 69 | 2.100757514 | 2.542703585 | 0.899242486 | 0.457296415 |
| 70 | 3.053638046 | 2.766323231 | -0.05363805 | 0.233676769 |
| 71 | 2.334254279 | 2.638202025 | 0.665745721 | 0.361797975 |
| 72 | 2.623188548 | 2.522498271 | 0.376811452 | 0.477501729 |
| 73 | 2.334161016 | 2.778362581 | 0.665838984 | 0.221637419 |
| 74 | 2.206397934 | 1.897552498 | 0.793602066 | 1.102447502 |
| 75 | 2.828477984 | 2.649175994 | 0.171522016 | 0.350824006 |
| 76 | 2.469956056 | 2.529920929 | 0.530043944 | 0.470079071 |
| 77 | 2.477625703 | 2.320876342 | 0.522374297 | 0.679123658 |
| 78 | 3.299803186 | 2.685983634 | -0.29980319 | 0.314016366 |
| 79 | 1.818852385 | 2.024096105 | 1.181147615 | 0.975903895 |
| 80 | 2.449556693 | 1.869247971 | 0.550443307 | 1.130752029 |
| 81 | 1.233957725 | 1.380181767 | 1.766042275 | 1.619818233 |
| 82 | 2.284325595 | 2.005774095 | 0.715674405 | 0.994225905 |
| 83 | 2.638685465 | 2.498776194 | 0.361314535 | 0.501223806 |
| 84 | 3.411048668 | 2.255522744 | -0.41104867 | 0.744477256 |
| 85 | 2.575536009 | 1.915932786 | 0.424463991 | 1.084067214 |
| 86 | 1.859336704 | 2.113528491 | 1.140663296 | 0.886471509 |
| 87 | 1.453638788 | 1.548335382 | 1.546361212 | 1.451664618 |
| 88 | 2.202724205 | 2.456879211 | 0.797275795 | 0.543120789 |
| 89 | 1.181868697 | 1.49221215 | 1.818131303 | 1.50778785 |
| 90 | 2.267445837 | 2.112884777 | 0.732554163 | 0.887115223 |
| 91 | 1.055473734 | 0.811012572 | 1.944526266 | 2.188987428 |
| 92 | 2.902439221 | 1.962006129 | 0.097560779 | 1.037993871 |
| 93 | 3.249674913 | 2.566318415 | -0.24967491 | 0.433681585 |
| 94 | 2.870743512 | 1.93025098 | 0.129256488 | 1.06974902 |
| 95 | 4.221350617 | 1.892963128 | -1.22135062 | 1.107036872 |
| 96 | 2.489776468 | 1.071082487 | 0.510223532 | 1.928917513 |
| 97 | 3.656264687 | 1.87845263 | -0.65626469 | 1.12154737 |
| 98 | 2.810510682 | 1.881309339 | 0.189489318 | 1.118690661 |
| 99 | 1.686702324 | 1.693476229 | 1.313297676 | 1.306523771 |

Табл. 2 – оценки семивариограммы

На рисунке 3 изображены графики двух оценок семивариограммы и самой семивариограммы, а на рисунке 4 – отклонения оценок от истинных значений.

Рис. 3 – оценки семивариограммы

Рис. 4 – отклонения оценок от истинных значений на каждом лаге

**Вывод:**

Был рассмотрен случайный процесс с известным видом ковариационной функции. Был найден вид семивариограммы из вида ковариационной функции: . Были построены классическая и робастная оценки семивариограммы. Из графического представления оценок сложно сделать какой-либо вывод о том, какая из оценок лучше описывает поведение семивариограммы, они довольно похожи. Робастные оценки устойчивы к отклонениям от однородности в выборке (в нашем случае отклонениям от нормального распределения). Тем не менее, для наших наблюдений заметной разницы между классической и робастной оценкой не наблюдается.

Также, как видно из графика оценок и из графика отклонений, начиная примерно с 80-го лага обе оценки имеют довольно большое отклонение и плохо отражают истинное поведение семивариограммы. Обычно рекомендуется считать лаги до из-за того, что для бо́льших лагов оценка строится по малому количеству наблюдений.

До 80-го лага оценки вполне адекватно описывают поведение семивариограммы, тем не менее отклонения присутствует (достигают примерно единицы).

**Задание 2. Исследование оценок на состоятельность в среднем квадратическом смысле.**

Задание 2 было выполнено в Python. В качестве параметров и выбираются (те же, что ранее). В таблице 3 приведены полученные значения для дисперсии классической оценки, а в таблице 4 – для робастной оценки.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Дисперсия классической оценки** | **n=10** | **n=25** | **n=50** | **n=75** | **n=100** | **n=150** | **n=200** | **n=250** | **n=300** | **n=350** |
| **h=2** | 2.793 | 1.029 | 0.501 | 0.331 | 0.247 | 0.164 | 0.123 | 0.098 | 0.082 | 0.07 |
| **h=5** | 3.906 | 1.357 | 0.635 | 0.414 | 0.307 | 0.202 | 0.151 | 0.12 | 0.1 | 0.086 |
| **h=7** | 6.408 | 1.436 | 0.655 | 0.423 | 0.312 | 0.205 | 0.153 | 0.122 | 0.101 | 0.086 |

Табл. 3 – дисперсия классической оценки

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Дисперсия робастной оценки** | **n=10** | **n=25** | **n=50** | **n=75** | **n=100** | **n=150** | **n=200** | **n=250** | **n=300** | **n=350** |
| **h=2** | 1.949 | 0.846 | 0.432 | 0.29 | 0.218 | 0.146 | 0.109 | 0.088 | 0.073 | 0.063 |
| **h=5** | 2.365 | 1.1 | 0.546 | 0.362 | 0.271 | 0.18 | 0.135 | 0.108 | 0.09 | 0.077 |
| **h=7** | 3.072 | 1.151 | 0.562 | 0.37 | 0.275 | 0.182 | 0.136 | 0.109 | 0.09 | 0.077 |

Табл. 4 – дисперсия робастной оценки

Графики зависимости дисперсий от количества наблюдений представлены на рисунке 5 и рисунке 6 для классической и робастной оценки соответственно.

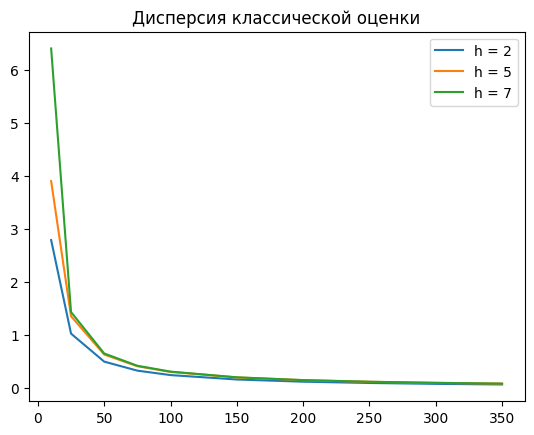


Рис. 5 – зависимость дисперсии классической оценки от количества наблюдений

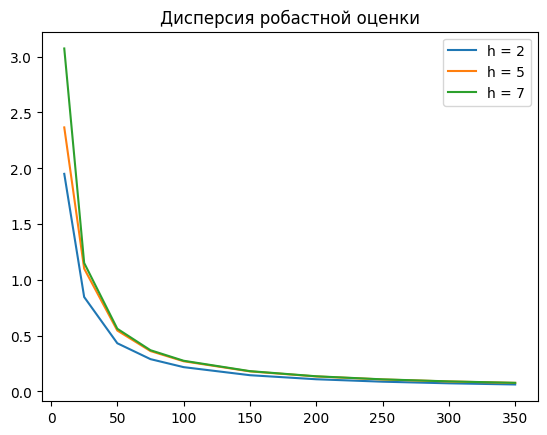


Рис. 6 – зависимость дисперсии робастной оценки от количества наблюдений

**Вывод:**

Из графиков на рис. 5 и 6 видно, что дисперсия оценок на любом из рассмотренных лагов стремится к нулю при больших n. Соответственно, можем сделать вывод, что в нашем случае оценки являются состоятельными в среднем квадратическом, что согласуется с теорией.

Также можно заметить, что дисперсия для бо́льших лагов сходится к нулю немного медленнее. Это можно объяснить тем, что в оценке чем больше лаг, тем меньше слагаемых в сумме.

Кроме того, дисперсия робастной оценки получилась немного меньше классической, особенно для n=10. Т.е. она вышла более устойчивой к малому количеству наблюдений.

**Литература**

* 1. Cressie N. Statistics for Spatial Data / N. Cressie – New York: Wiley Classics Library, 2015. – 928 p.

**Листинг программы**

import math

import numpy as np

import pandas as pd

import scipy as sp

import matplotlib.pyplot as plt

Задание 2.

Задание параметров

D = 3

alpha = 1.5

N = [10, 25, 50, 75, 100, 150, 200, 250, 300, 350]

H = [2, 5, 7]

Заданная ковариационная функция

def R(t):

    return D \* math.exp(- alpha \* abs(t))

def Cum(t):

    return (R(0)-R(t))

Моделирование десяти наборов наблюдений

def gen\_x(n):

    np.random.seed(2023)

    x = np.random.normal(0, 1, n)

    a0 = math.sqrt(D \* (1 - math.exp(- 2 \* alpha)))

    b1 = math.exp(- alpha)

    res = [0.] \* n

    res[0] = a0 \* x[0]

    for j in range(1, n):

        res[j] = a0 \* x[j] + b1 \* res[j - 1]

    return res

def gen\_X(N):

    count = len(N)

    X = [0.]\*count

    for i in range(count):

        X[i] = [.0]\*N[i]

    for i in range(count):

        X[i] = gen\_x(N[i])

    return X

X = gen\_X(N)

Вычисление дисперсий оценок

def gen\_D\_classic(X, H):

    D\_classic = [.0] \* len(H)

    for i in range(len(H)):

        D\_classic[i] = [.0] \* len(X)

    for i in range(len(H)):

        h = H[i]

        for j in range(len(X)):

            D\_classic[i][j] = 0

            for s in range(1, len(X[j]) - h + 1):

                for t in range(1, len(X[j]) - h + 1):

                    D\_classic[i][j] += math.pow(Cum(s - t + h) + Cum(s - t - h) - (2 \* Cum(s - t)), 2)

            D\_classic[i][j] /= 2 \* math.pow(len(X[j]) - h, 2)

    return D\_classic

def gen\_D\_robust(X, H):

    D\_robust = [.0] \* len(H)

    for i in range(len(H)):

        D\_robust[i] = [.0] \* len(X)

    for i in range(len(H)):

        h = H[i]

        for j in range(len(X)):

            D\_robust[i][j] = 0

            for s in range(1, len(X[j]) - h + 1):

                for t in range(1, len(X[j]) - h + 1):

                    D\_robust[i][j] += math.pow(Cum(s - t + h) + Cum(s - t - h) - (2 \* Cum(s - t)), 2)

            D\_robust[i][j] \*= 10 / math.pow(len(X[j]) - h, 2)

            D\_robust[i][j] /= math.pow(0.457 + (0.494 / (len(X[j]) - h)) + (0.045 / math.pow(len(X[j]) - h, 2)), 2)

            D\_robust[i][j] \*= (1 / math.sqrt(math.pi)) - (math.pow(math.gamma(3 / 4), 2) / math.pi)

            D\_robust[i][j] \*= math.pow(math.gamma(3 / 4), 6) / math.pow(math.pi, 3)

    return D\_robust

Вывод значений дисперсии

D\_classic = gen\_D\_classic(X, H)

D\_robust = gen\_D\_robust(X, H)

print("Дисперсия классической оценки:")

for i in range(len(H)):

    for j in range(len(X)):

        print(D\_classic[i][j], end = " ")

    print("")

print("Дисперсия робастной оценки:")

for i in range(len(H)):

    for j in range(len(X)):

        print(D\_robust[i][j], end = " ")

    print("")

Графики зависимости дисперсий от количества наблюдений

h = [.0] \* len(X)

for i in range(len(X)):

    h[i] = len(X[i])

for i in range(len(H)):

    plt.plot(h, D\_classic[i], label = 'h = '+str(H[i]))

    plt.title("Дисперсия классической оценки")

    plt.legend()

plt.show()

for i in range(len(H)):

    plt.plot(h, D\_robust[i], label = 'h = '+str(H[i]))

    plt.title("Дисперсия робастной оценки")

    plt.legend()

plt.show()