Лабораторная работа № 4

Изучение описательной статистики

Цель работы: Изучение описательной статистики средствами языка Python 3.

Содержание работы:

1. Описательная статистика

Описательная статистика представляет собой из разделов статистической науки, в рамках которого изучаются методы описания и представления основных свойств данных. Позволяет обобщать первичные результаты, полученные при наблюдении или в эксперименте. Применение описательной статистики включает следующие этапы:

1) сбор данных;
2) категоризация данных;
3) обобщение данных;
4) представление данных;

Пусть $X_1, X_2 \dots X_n$ – выборка независимых случайных величин. Упорядочим эти величины по возрастанию, иными словами, построим вариационный ряд:

$$X(1) < X(2) < \dots < X(n), \quad (1)$$

где $X(1) = \min(X_1, X_2 \dots X_n)$, а $X(n) = \max(X_1, X_2 \dots X_n)$.

Элементы вариационного ряда (1) называются *порядковыми статистиками*.

Величины $d(i) = X(i+1) - X(i)$ называются *спейсингами* или *расстояниями между порядковыми статистиками*.

Размахом выборки называется величина

$$R = X(n) - X(1).$$

Иными словами, размах – это расстояние между максимальным и минимальным членом вариационного ряда.

Выборочное среднее равно: $\bar{X} = (X_1 + X_2 + \dots + X_n)/n$

Среднее арифметическое

Вероятно, большинство из вас использовало такую важную описательную статистику, как среднее арифметическое.

Среднее арифметическое – очень информативная мера «центрального положения» наблюдаемой переменной, особенно если сообщается ее доверительный интервал. Исследователю нужны такие статистики, которые

позволяют сделать вывод относительно популяции в целом. Одной из таких статистик является среднее арифметическое.

Доверительный интервал для среднего представляет интервал значений вокруг оценки, где с данным уровнем доверия, находится «истинное» (неизвестное) среднее популяции.

Например, если среднее выборки равно 23, а нижняя и верхняя границы доверительного интервала с уровнем $p = 0{,}95$ равны 19 и 27 соответственно, то можно заключить, что с вероятностью 95% интервал с границами 19 и 27 накрывает среднее популяции.

Если вы установите больший уровень доверия, то интервал станет шире, поэтому возрастает вероятность, с которой он «накрывает» неизвестное среднее популяции, и наоборот.

Хорошо известно, например, что чем «неопределенней» прогноз погоды (т.е. шире доверительный интервал), тем вероятнее он будет верным. Заметим, что ширина доверительного интервала зависит от объема или размера выборки, а также от разброса (изменчивости) данных. Увеличение размера выборки делает оценку среднего более надежной. Увеличение разброса наблюдаемых значений уменьшает надежность оценки.

Вычисление доверительных интервалов основывается на предположении нормальности наблюдаемых величин. Если это предположение не выполнено, то оценка может оказаться плохой, особенно для малых выборок.

При увеличении объема выборки, скажем, до 100 или более, качество оценки улучшается и без предположения нормальности выборки.

Довольно трудно «ощутить» числовые измерения, пока данные не будут содержательно обобщены. Диаграмма часто полезна в качестве отправной точки. Мы можем также сжать информацию, используя важные характеристики данных. В частности, если бы мы знали, из чего состоит представленная величина, или если бы мы знали, насколько широко рассеяны наблюдения, то мы бы смогли сформировать образ этих данных.

Среднее арифметическое, которое очень часто называют просто «среднее», получают путем сложения всех значений и деления этой суммы на число значений в наборе.

Это можно показать с помощью алгебраической формулы. Набор n наблюдений переменной X можно изобразить как $X_1, X_2, X_3, \ldots, X_n$. Например, за X можно обозначить рост индивидуума (см), X_1 обозначить рост 1-го индивидуума, а X_i – рост i-го индивидуума. Формула для определения среднего арифметического наблюдений (произносится «икс с чертой»):

$$\bar{X} = (X_1 + X_2 + \ldots + X_n) / n$$

Можно сократить это выражение:

$$\bar{X} = \frac{\sum_{i=1}^{n} X_i}{n}$$

где (греческая буква «сигма») означает «суммирование», а индексы внизу и вверху этой буквы означают, что суммирование производится от $i = 1$ до $i = n$.

Медиана

Если упорядочить данные по величине, начиная с самой маленькой величины и заканчивая самой большой, то медиана также будет характеристикой усреднения в упорядоченном наборе данных.

Медиана делит ряд упорядоченных значений пополам с равным числом этих значений как выше, так и ниже ее (левее и правее медианы на числовой оси).

Вычислить медиану легко, если число наблюдений n нечетное. Это будет наблюдение номер $(n + 1)/2$ в нашем упорядоченном наборе данных.

Например, если $n = 11$, то медиана – это $(11 + 1)/2$, т. е. 6-е наблюдение в упорядоченном наборе данных.

Если n четное, то, строго говоря, медианы нет. Однако обычно мы вычисляем ее как среднее арифметическое двух соседних средних наблюдений в упорядоченном наборе данных (т. е. наблюдений номер $(n/2)$ и $(n/2 + 1)$).

Так, например, если $n = 20$, то медиана – это среднее арифметическое наблюдений номер $20/2 = 10$ и $(20/2 + 1) = 11$ в упорядоченном наборе данных.

Мода

Мода M – это значение, которое встречается наиболее часто в наборе данных; если данные непрерывные, то мы обычно группируем их и вычисляем модальную группу.

Некоторые наборы данных не имеют моды, потому что каждое значение встречается только 1 раз. Иногда бывает более одной моды; это происходит тогда, когда 2 значения или больше встречаются одинаковое число раз и встречаемость каждого из этих значений больше, чем любого другого значения.

Как обобщающую характеристику моду используют редко.

Среднее геометрическое

При несимметричном распределении данных среднее арифметическое не будет обобщающим показателем распределения.

Если данные скошены вправо, то можно создать более симметричное распределение, если взять логарифм (по основанию 10 или по основанию e) каждого значения переменной в наборе данных. Среднее арифметическое значений этих логарифмов – характеристика распределения для преобразованных данных.

Чтобы получить меру с теми же единицами измерения, что и первоначальные наблюдения, нужно осуществить обратное преобразование – потенцирование (т. е. взять антилогарифм) средней логарифмированных данных; мы называем такую величину *среднее геометрическое*.

Если распределение данных логарифма приблизительно симметричное, то среднее геометрическое подобно медиане и меньше, чем среднее необработанных данных.

Взвешенное среднее

Взвешенное среднее используют тогда, когда не-которые значения интересующей нас переменной x более важны, чем другие. Мы присоединяем вес w_i к каждому из значений X_i в нашей выборке для того, чтобы учесть эту важность.

Если значения $X_1, X_2, \ldots, X_n$ имеют соответствующий вес $w_1, w_2, \ldots, w_n$, то взвешенное арифметическое среднее выглядит следующим образом:

$$\frac{w_1X_1 + w_2X_2 + \cdots + w_nX_n}{w_1 + w_2 + \cdots + w_n} = \frac{\sum w_iX_i}{\sum w_i}$$

Например, предположим, что мы заинтересованы в определении средней продолжительности госпитализации в каком-либо районе и знаем средний реабилитационный период больных в каждой больнице. Учитываем количество информации, в первом приближении принимая за вес каждого наблюдения число больных в больнице.

Взвешенное среднее и среднее арифметическое идентичны, если каждый вес равен единице.

Размах (интервал изменения)

Размах – это разность между максимальным и минимальным значениями переменной в наборе данных; этими двумя величинами обозначают их разность. Обратите внимание, что размах вводит в заблуждение, если одно из значений есть выброс.

Размах, полученный из квантилей

Что такое квантили?

Предположим, что мы расположим наши данные упорядоченно от самой маленькой величины переменной X и до самой большой величины. Величина $X_{0.01}$, до которой расположен 1% наблюдений (и выше которой расположены 99% наблюдений), называется *первым квантилем*.

Величина $X_{0.02}$, до которой находится 2% наблюдений, называется *вторым квантилем*, и т. д.

Смысл квантили состоит в том, что левее точки x_p лежит приблизительно $100p\%$ наблюдений.

Величины $X_{0.1}, X_{0.2}, \dots, X_{0.9}$, которые делят упорядоченный набор значений на 10 равных групп, т. е. 10-й, 20-й, 30-й,..., 90 и квантили, называются *децилями*. Величины $X_{0.25}, X_{0.5}, X_{0.75}$, которые делят упорядоченный набор значений на 4 равные группы, т.е. 25-й, 50-й и 75-й квантили, называются *квартилями* Q. 50-й квантиль – это *медиана*.

Применение квантилей

Мы можем добиться такой формы описания рассеяния, на которую не повлияет выброс (аномальное значение), исключая экстремальные величины и определяя размах остающихся наблюдений.

Межквартильный размах d (иногда используется обозначение «интерквартильный размах» *IR*) – это разница между 1-м и 3-м квартилями, т.е. между 25-м и 75-м квантилями:

$$d = Q_{0.75} - Q_{0.25}.$$

В него входят центральные 50% наблюдений в упорядоченном наборе, где 25% наблюдений находятся ниже центральной точки и 25% – выше.

Интердецильный размах содержит в себе центральные 80% наблюдений, т. е. те наблюдения, которые располагаются между 10-м и 90-м квантилями.

Мы часто используем размах, который содержит 95% наблюдений, т.е. он исключает 2,5% наблюдений снизу и 2,5% сверху. Указание такого интервала актуально, например, для осуществления диагностики болезни. Такой интервал называется *референтный интервал*, *референтный размах* или *нормальный размах*.

Дисперсия

Один из способов измерения рассеяния данных заключается в том, чтобы определить степень отклонения каждого наблюдения от средней арифметической.

Очевидно, что чем больше отклонение, тем больше изменчивость, вариабельность наблюдений.

Однако мы не можем использовать среднее этих отклонений как меру рассеяния, потому что положительные отклонения компенсируют отрицательные отклонения (их сумма равна нулю). Чтобы решить эту проблему, мы возводим в квадрат каждое отклонение и находим среднее возведенных в квадрат отклонений; эта величина называется вариацией, или дисперсией.

Возьмем n наблюдений $X_1, X_2, X_3, \ldots, X_n$, среднее которых равняется $\bar{X} = \frac{\sum_{i=1}^{n} X_i}{n}$.

Вычисляем *дисперсию*:

$$D = \sigma^2 = \frac{\sum_{i=1}^{n} (X_i - \bar{X})^2}{n}.$$

В случае, если мы имеем дело не с генеральной совокупностью, а с выборкой, то вычисляется *выборочная дисперсия*:

$$s^2 = \frac{\sum_{i=1}^{n} (X_i - \bar{X})^2}{n-1}.$$

Теоретически можно показать, что получится более точная дисперсия по выборке, если разделить не на n, а на $(n-1)$. Единицы измерения (размерность) вариации – это квадрат единиц измерения первоначальных на-блюдений.

Например, если измерения производятся в килограммах, то единица измерения вариации будет килограмм в квадрате.

Среднеквадратическое отклонение, стандартное отклонение выборки

Среднеквадратическое отклонение – это положительный квадратный корень из дисперсии.

$$s = \sqrt{D}$$

Стандартное отклонение выборки – корень из выборочной дисперсии:

$$s = \sqrt{\frac{\sum_{i=1}^{n} (X_i - \bar{X})^2}{n-1}}$$

Мы можем представить себе стандартное отклонение как своего рода среднее отклонение наблюдений от среднего. Оно вычисляется в тех же единицах (размерностях), что и исходные данные.

Если разделить стандартное отклонение на среднее арифметическое и выразить результат в процентах, получится коэффициент вариации.

Он является мерой рассеяния, не зависит от единиц измерения (безразмерный), но имеет некоторые теоретические неудобства и поэтому не очень одобряется статистиками.

Вариация в пределах субъектов и между субъектами

Если провести повторные измерения непрерывной переменной у исследуемого объекта, то можно увидеть ее изменения (внутрисубъектные изменения). Это можно объяснить тем, что объект не всегда может дать точные и те же самые ответы, и/или ошибкой, погрешностью измерения. Однако при измерениях у одного объекта вариация обычно меньше, чем вариация единичного измерения в группе (межсубъектные изменения).

Например, вместимость легкого 17-летнего мальчика составляет от 3,60 до 3,87 л, когда измерения повторяются не менее 10 раз; если провести однократное измерение у 10 мальчиков того же возраста, то объем будет между 2,98 и 4,33 л. Эти концепции важны в плане исследования.

Выброс

Резко отклоняющееся значение наблюдаемой величины.

Выбросом считается наблюдение, которое лежит аномально далеко от остальных из серии параллельных наблюдений. Выбросы – это значения количественного признака, располагающиеся на краях интервала допустимых значений.

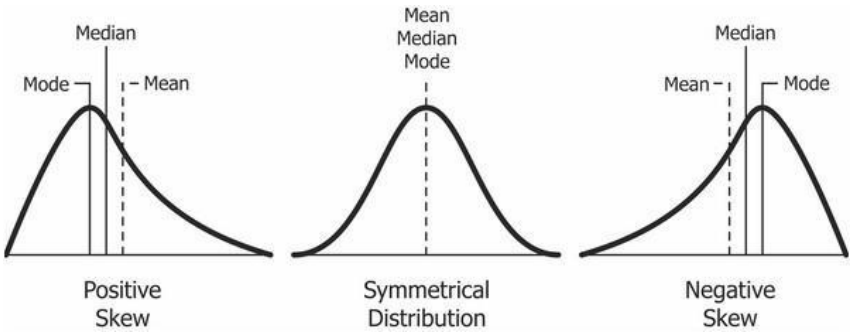
Асимметрия

Асимметрия – это мера асимметрии распределения вероятностей вещественной случайной величины относительно ее среднего значения. Значение асимметрии может быть положительным, отрицательным или неопределенным.

В идеальном нормальном распределении хвосты с обеих сторон кривой являются точными зеркальными отражениями друг друга.

Когда распределение перекошено влево, хвост на левой стороне кривой длиннее, чем хвост на правой стороне, а среднее значение меньше моды. Эта ситуация также называется отрицательной асимметрией.

Когда распределение перекошено вправо, хвост с правой стороны кривой длиннее, чем хвост с левой стороны, а среднее значение больше моды. Эта ситуация также называется положительной асимметрии.



Коэффициента асимметрии характеризует симметричность в распределении наблюдений:

$$As = \frac{\frac{1}{n}\sum_{i=1}^{n}(X_i - \bar{X})^3}{D^3}.$$

Наличие симметрии характеризуется близостью коэффициента асимметрии к нулю.

Коэффициент эксцесса характеризует вероятность больших (по модулю) значений и равен

$$Kurt = \frac{\frac{1}{n}\sum_{i=1}^{n}(X_i - \bar{X})^4}{D^4}.$$

эксцесс характеризует островершинность распределения, а также частоту появления значений, которые удалены от среднего, т.е. насколько много наблюдений находится в «хвостах» распределения.

Таким образом, асимметрия и эксцесс – это оценки для третьего и четвертого центральных нормированных моментов. Существуют и другие формулы для их оценивания. Следует отметить, что для нормального распределения коэффициент асимметрии равен 0, а эксцесс равен 3. Если эксцесс сильно отличается от трёх, то говорят о наличии «тяжёлых хвостов».

2. Файлы. Работа с файлами.

Прежде, чем работать с файлом, его надо открыть. С этим замечательно справится встроенная функция open:

```
f = open('text.txt', 'r')
```

У функции open много параметров, нам пока важны 3 аргумента. Первый – это имя файла (путь). Путь к файлу может быть относительным или абсолютным. Второй аргумент, это режим, в котором мы будем открывать файл.

Режим	Обозначение
'r'	Открытие на чтение (значение по умолчанию)
'w'	Открытие на запись, содержимое файла удаляется, если файла не существует, создается новый.
'x'	Открытие на запись, если файла не существует, иначе исключение.
'a'	Открытие на запись. Если файла не существует, создается новый. Если файл существует, информация добавляется в конец файла.
'b'	Открытие в бинарном режиме.
't'	Открытие в текстовом режиме (значение по умолчанию).
'+'	Открытие на чтение и запись

Режимы могут быть объединены, то есть, к примеру, 'rb' - чтение в двоичном режиме. По умолчанию режим равен 'rt'.

И последний аргумент, encoding, нужен только в текстовом режиме чтения файла. Этот аргумент задает кодировку.

После окончания работы с файлом его обязательно нужно закрыть с помощью метода close:

```
f.close()
```

В более сложных случаях (словарях, вложенных кортежей и т. д.) алгоритм записи придумать сложнее. Но этого и не требуется. В python уже давно придумали средства, такие как pickle или json, позволяющие сохранять в файле сложные структуры.

3. Использование библиотеки numpy при работе с файлами.

Библиотека Numpy предоставляет широкий набор модулей и функций для обработки числовых данных, в том числе и для чтения массивов из файлов.

Одна из реализаций возможна с помощью функции loadtxt, результат работы которой будет записан в numpy.array.

```
import numpy as np

data = np.loadtxt("data.txt", delimiter='\t', dtype=np.float)
```

Прототип функции loadtxt выглядит следующим образом:

```
numpy.loadtxt(fname, dtype=<class 'float'>, comments='#', delimiter=None, converters=None, skiprows=0, usecols=None, unpack=False, ndmin=0, encoding='bytes', max_rows=None)¶
```

Параметр	Тип	Назначение
`fname`	str, строковый, file, pathlib.Path	Файл, имя файла или генератор для чтения. Если расширение имени файла .gz или .bz2, файл сначала распаковывается. Обратите внимание, что генераторы должны возвращать байтовые строки для Python 3k.
`dtype`	data-type, необязательный параметр	Тип данных результирующего массива; по умолчанию: float. Если это структурированный тип данных, результирующий массив будет одномерным, и каждая строка будет интерпретироваться как элемент массива. В этом случае количество используемых столбцов должно соответствовать количеству полей в типе данных.

`comments`	str, или последовательность строк, необязательный параметр	Символы или список символов, используемых для обозначения начала комментария. Ни один не подразумевает никаких комментариев. Для обратной совместимости байтовые строки будут декодированы как «latin1». Значением по умолчанию является «#».
`delimiter`	str, необязательный параметр	Строка, используемая для разделения значений. Для обратной совместимости байтовые строки будут декодированы как «latin1». По умолчанию используется пробел.
`converters`	dict, необязательный параметр	Словарь сопоставляет номер столбца с функцией, которая будет анализировать строку столбца в требуемое значение. Например, если столбец 0 является строкой данных: `converters = {0: datestr2num}`. Конвертеры также можно использовать для предоставления значения по умолчанию для пропущенных данных (см. также genfromtxt): `converters = {3: lambda s: float (s.strip () или 0)}`. По умолчанию: `None`.
`skiprows`	int, необязательный параметр	Пропустить первые строки пропуска, включая комментарии; по умолчанию: 0.
`usecols`	int or sequence, необязательный параметр	Какие столбцы читать, с начиная 0 (является первым). Например, `usecols = (1,4,5)` извлечет 2-й, 5-й и 6-й столбцы. Значение по умолчанию `None`, обозначает, что все столбцы читаются. Использование данного параметра изменено в версии 1.11.0. Когда нужно прочитать один столбец, можно использовать целое число вместо кортежа. Например, `usecols = 3` читает четвертый столбец так же, как и `usecols = (3,)`.

`unpack`	bool, необязательный параметр	Если `True`, возвращенный массив транспонируется, так что аргументы могут быть распакованы с использованием `x, y, z = loadtxt(...)`. При использовании со структурированным типом данных массивы возвращаются для каждого поля. По умолчанию установлено значение `False`.
`ndmin`	int, необязательный параметр	Возвращаемый массив будет иметь как минимум размерность ndmin. В противном случае одномерные оси будут сжаты. Допустимые значения: 0 (по умолчанию), 1 или 2. Новшество в версии 1.6.0.
`encoding`	str, необязательный параметр	Кодировка, используемая для декодирования входного файла. Не относится к входным потокам. Специальное значение «bytes» включает обходные пути обратной совместимости, которые гарантируют, что вы получите байтовые массивы в качестве результатов, если это возможно, и передадут закодированные «latin1» строки в преобразователи. Установите это значение, чтобы получать массивы Unicode и передавать строки в качестве входных данных для преобразователей. Если установлено значение `None`, используется системное значение по умолчанию. Значением по умолчанию являются «байты». Новшество в версии 1.14.0.
`max_rows`	int, необязательный параметр	Читает строки содержимого max_rows после строк пропуска. По умолчанию читаются все строки. Новшество в версии 1.16.0.

Возвращаемое функцией значение имеет тип ndarray, и представляет собой информацию из файла.

Пример использования функции с другими параметрами:

```
d = StringIO(u"M 21 72\nF 35 58")
np.loadtxt(d, dtype={'names': ('gender', 'age', 'weight'), 'formats': ('S1', 'i4', 'f4')})
c = StringIO(u"1,0,2\n3,0,4")
x, y = np.loadtxt(c, delimiter=',', usecols=(0, 2), unpack=True)
```

4. Использование регулярных выражений (Regular expression operations).

Данный способ можно назвать стрельбой из пушки по воробьям, однако у него все же есть свои плюсы: если данные в файле расположены хаотично и отсутствует постоянная структура, то функции split невозможно задать конкретный разделитель и для решения задачи можно использовать регулярное выражение, которое найдет в строке все числа, несмотря на их расположение и наличие разделителей.

```
import re

file = open("data.txt")
values = file.read().split("\n")
data = []

for key in values:
    value = re.findall(r"[-+]?\d*\.\d+|\d+", key)

    if value != []:
        data.append(value)
```

Более подробную информацию по регулярным выражениям модно найти по ссылке.

5. Другие способы чтения информации из файлов.

1 Использование CSV Reader. Если данные записаны в виде матрицы с постоянными разделителями, то выполнить их чтение можно при помощи модуля CSV Reader, указав в качестве параметра значение разделителя.

2 Использование Numpy genfromtxt. Данный способ не сильно отличается от loadtxt, за исключением того, что genfromtxt предоставляет более широкий набор входных параметров: указание различных типов данных для каждого из столбцов, передача ключей для создания ассоциативного массива и так далее.

3 Использование Pandas read_csv. Pandas — мощная библиотека для обработки данных на Python. В данном примере рассматривается только чтение данных, но её возможности этим не ограничены. Метод read_csv предоставляет широкий набор входных параметров, а также показывается высокую скорость работы даже при работе с большими объемами данных.

4 Преобразование при помощи функции map.

6. Ход работы

1. Внимательно прочитать текст лабораторной работы и проделать самостоятельно все примеры.
2. В качестве входных значений используйте результаты успеваемости в группах ИСТ, ПРИ, ИВТ (текущего 4, 3, и 2 курса) с официального сайта ВолГУ. Можно использовать прямое копирование данных с сайта в блокнот, в этом случае у вас для обработки данных должна использоваться библиотека re (соответствующие коды должны иметься в отчете). При необходимости можно использовать автоматические средства загрузки (это тоже должно подтверждаться кодом).
3. Выберите не менее 9 дисциплин, которые читаются всему потоку (трем группам на одном курса разом). Каждую дисциплину для данного курса рассматривайте отдельно. Проведите анализ выбранных вами данных с использованием всех средств описательной статистики. Выполните построение соответствующих столбчатых диаграмм, поверх диаграмм должен располагаться график распределения, наиболее близко характеризующий вашу выборку (пример см. на следующей странице).
4. Представить отчет о выполнении работы.

5. Ответить на вопросы преподавателя.