تمرین شبیه سازی سری پنجم
علی قبله ۹۹۱۰۹۹۷۱

۱. ولگشت ۲ بعدی

در این برنامه می‌خواهیم یک ولگشت ۲ بعدی را شبیه سازی کنیم. ابتدا احتمال های حرکت را که ثابت هستند و حداقل و حداکثر تعداد قدم هارا مشخص می‌کنیم. حرکت ولگرد با یک آرایه رندوم، بسیار ساده است. با مشخص کردن ۴ بازه و انطباق هر بازه به یک نوع حرکت می‌توان این ولگشت را ساخت.

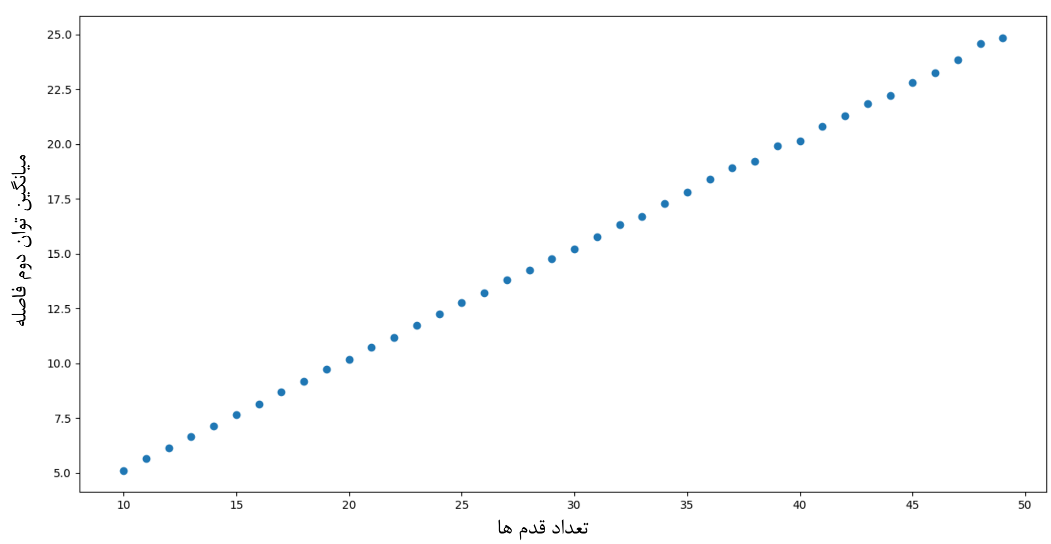
بخش مهم این سوال به دست آوردن یک نمودار برای رابطه زیر است:

$$< r > = ۲dDt$$

حال با تعریف دو تابع کار را ادامه می‌دهیم. تابع اول میانگین واریانس x را پیدا می‌کند و تابع دوم همینکار را برای y انجام می‌دهد. در کنار این دو تابع کاربردی یک تابع دیگر به نام بهترین شیب کافیست تعریف کنیم تا شیب نمودار را به ما اعلام کند تا بتوانیم درستی عبارت را اثبات کنیم. این برنامه برای چندین بار اجرا می‌شود (sample number) برای آنکه مقادیر واریانس را بتوانیم به صورت یک پلات حاضر کنیم و با شیب آن عبارت را ثابت کنیم.

در این برنامه یک متغیر به نام grid به صورت global تعریف شده است که یک آرایه با ۳ بعد است و به این دلیل اینگونه تعریف شده است که در بدنه اصلی برنامه پس از مقدار دهی بتوان در بخش های پیدا کردن واریانس از آن استفاده نمود. این عبارت به نوعی حافظه برای ذخیره مسیر هاست که بعد در توابع واریانس از آن استفاده شود.

برای ۴۰۰۰۰ داده و حدفاصل قدم های ۱۰ تا ۵۰، نمودار به شکل زیر خواهد بود:

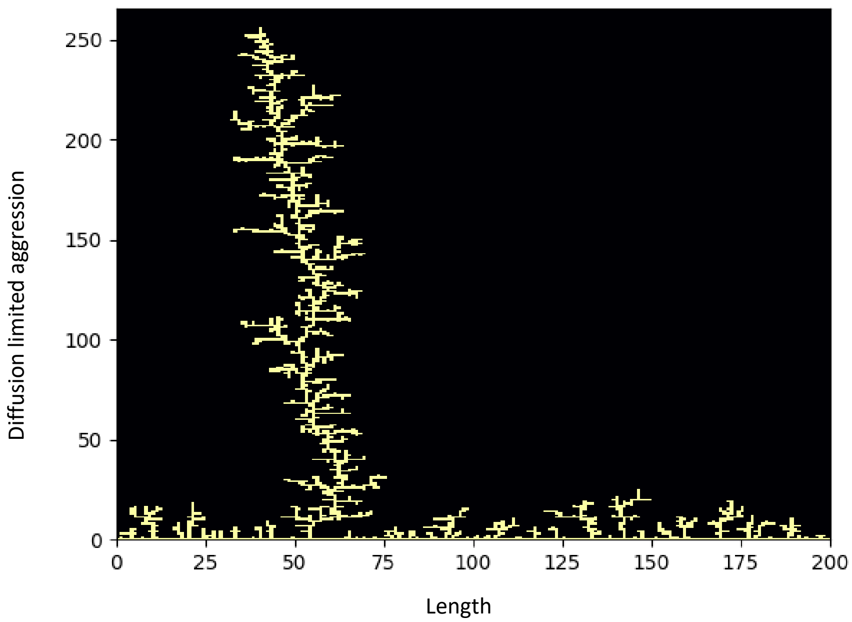


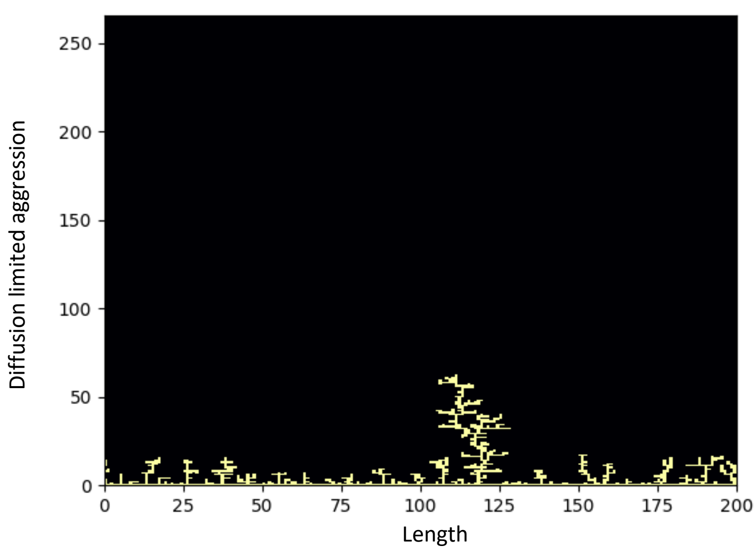
شیب را از برنامه داریم: ۰.۵۰۸۷۶۰۸۵۳۰۹۰۵۶۹۵که این عبارت بسیار نزدیک به ۰.۵ در تئوری است. لازم به ذکر است که میانگین توان دوم فاصله روی ۴۰۰۰۰ میانگین گرفته شده است. (Run time = ۵۰ min)

۲. خوشه های تجمع پخش محدود

مراحل پیدا کردن خوشه‌ها را در سری های پیشین داشتیم. در تابع اول با استفاده از یک رندوم حرکت در دو بعد را با توجه به مرزها مشخص می‌کنیم. این تابع دو حلقه تو در تو دارد یکی پایان مشخص ندارد و تا زمانی ذره به یک خوشه متصل نباشد ادامه خواهد داشت و حلقه بیرونی به تعداد ذره‌ها می‌باشد. لازم به ذکر است که در این کد می‌توان ۲ رویکرد داشت. یکی احترام به مرز مشخص و دیگری پتانسیل جابه‌جایی مرز برای آنکه خوشه ها به آن نرسند. تفاوت این مسئله شکل نمودار ها و زمان صرف شده است که با احترام به مرزها نصف زمان لحاظ نکردن مرز ها خواهد بود. در هردو شکل یکی خواهد بود اما طول یک خوشه تغییر خواهد کرد.

برای طول ۲۰۰ و تعداد ۲۰۰۰ ذره خواهیم داشت:





نمودار اول با جابه‌جایی مرز هاست و دومی بودن آن است. پر واضح است که یک خوشه تنها رشد می‌کند و خوشه های دیگر به آن متصل نخواهند شد. (Run time: ۲۰ seconds)

۳. ولگشت خودپرهیز

اول لازم به ذکر است که برای این سوال درکنار کد خودم (که ناقص بود و در ادامه توضیح خواهم داد)، کد آقای سینا معمر و کوروش علامه نیز وجود دارد (برای ایده گرفتن از روی ایده این دو نفر)(در داخل یک فایل به نام Extra). برای این سوال کافی بود همان مسیر ولگشت دو بعدی را ادامه دهیم و تنها کافیست که هربار که یک مسیری را طی می‌کند در حافظه ذخیره کند و در ادامه این مسیر را تکرار نکند. در این کد هربار که ولگرد به نقطه قبلی که از آن عبور کرده است می‌رسد به نقطه یکی قبل باز می‌گردد و دوباره امتحان می‌کند.

مشکل من در این سوال در ابتدا یک تابع برای ذخیره مسیر کلی و شمارش تعداد مسیر های متفاوت بود. طبعا مسیر های خودپرهیز با بزرگ شدن طول، زیاد می‌شود.

با توجه به ذخیره مسیر های محتمل می‌توان تعداد را به راحتی به دست آورد که متاسفانه من نتوانستم آن را بدست آورم.

در کد آقای معمر دو تابع مهم وجود دارد یکی تابع شمارش مسیر های بسته و دیگری تابع مسیر های خودپرهیز. در تابع اول در صورت وجود حرکتی برای ولگرد، این متحرک حرکت می‌کند و در غیر این صورت از حرکت باز می‌ایستد. در تابع دوم تمامی مسیر های ممکن را از مسیر های بسته کم کرده و تعداد تابع های خودپرهیز را از این طریق به دست می‌آورد.

در کد آقای علامه از همان ایده اولیه تولید یک حافظه استفاده شده است که به برنامه ما کاملا نزدیک است.(اصل برنامه در یک Jupyter NoteBook بوده است.

با اینکه کد برنامه به صورت کامل محاسبه نشده است اما می توان با استفاده از دانش ریاضی به صورت ساده تا طول ۶ را به سادگی به دست آورد.

N = ۱ : ۴ possible ways to move   N=۲ : ۴*۳ possible ways to move

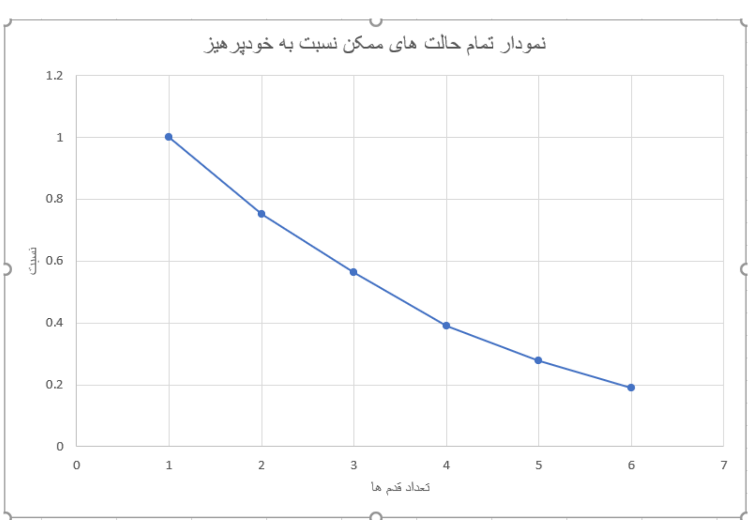
N = ۳ : ۴*$۳^{۲}$ possible ways to move N=۴ : ۴*$۳^{۳}$-۸ possible ways (۸ of them are closed paths)

N = ٥ : ۴*$۳^{۴}$-۴*۸-۸ possible ways (٤*۸ of them are closed path for N=٥ and ۸ are for N=٤)

N = ٦ : ۴*$۳^{۵}$-۴*(٤*۸-۸) – ٤*۸ possible ways (like above)

پس خواهیم داشت:

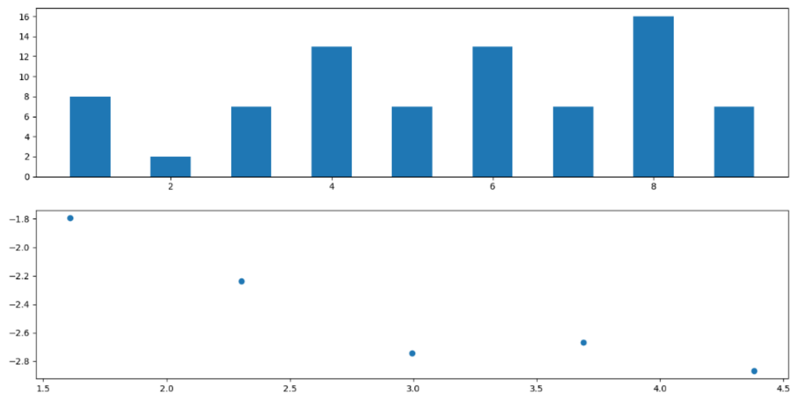
N=۱	۴
N=۲	۱۲
N=۳	۳۶
N=٤	۱۰۰
N=٥	۲۸۴
N=٦	۷۸۰

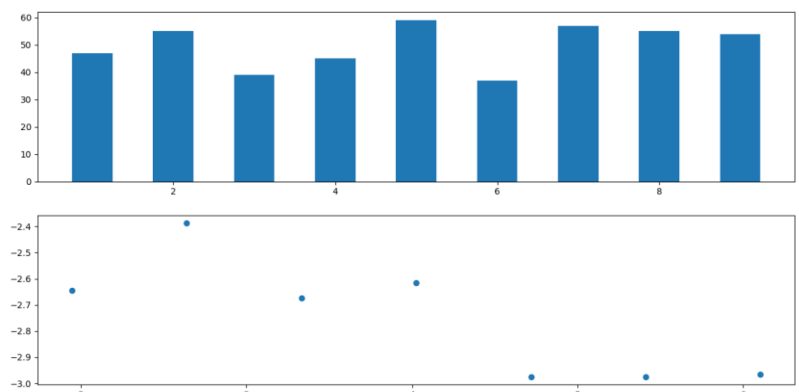


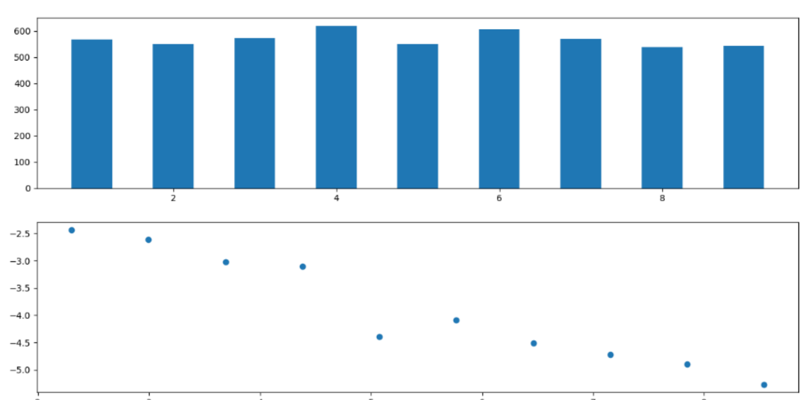
٤. تابع توزیع احتمال

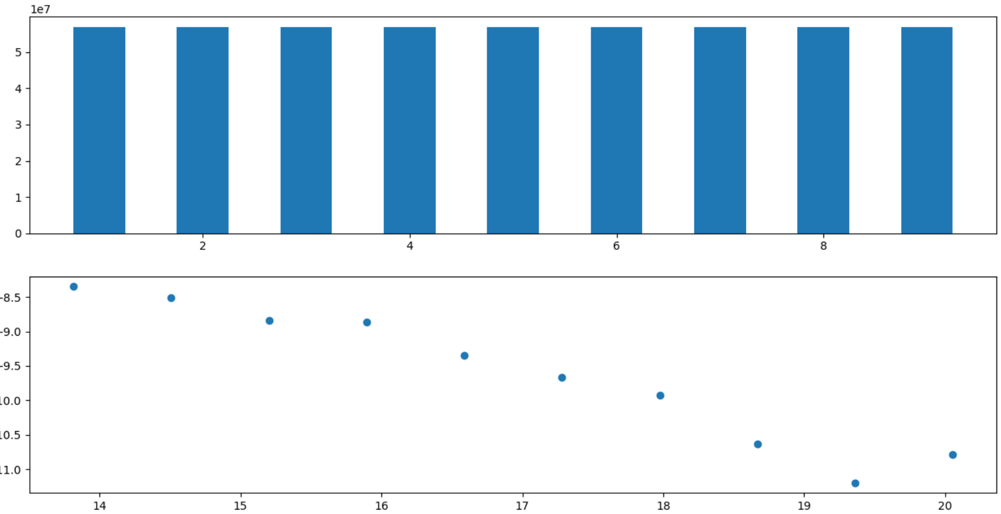
در این سوال هدف مشخص کردن میزان رندوم بودن، رندوم کامپیوتر است. برای این سوال چندین بار اعداد ۰ تا ۹ را تولید می‌کنیم. در کنار اینکار تابع بهترین شیب را برای نمودار واریانس میسازیم و با استفاده از آن شیب را به دست می‌آوریم.

به ترتیب برای *increament = 5, 7, 10, 10000* داریم (۲۰ عدد و ۶۰ عدد و ۶۰۰ عدد و ۶ میلیون عدد)









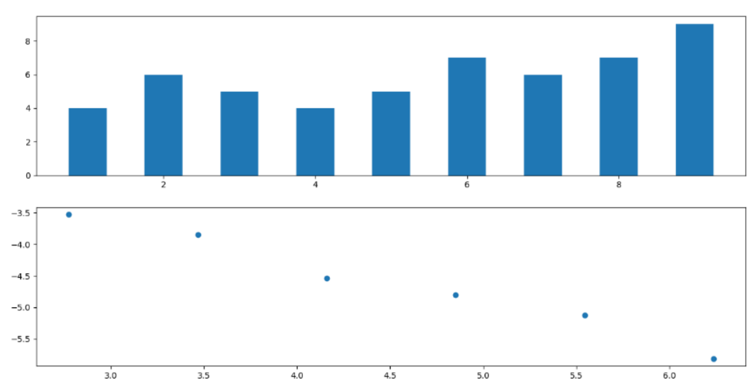
نمودار اول تعداد هر عدد به دست آمده در رندوم هاست. نمودار دوم لوگاریتم واریانس *(sigma/N)* بر حسب *Log N* می باشد.(شکل ها به صورت فشرده و در هم بود و لیبل را زیر نمودار به همین صورت نوشتم)

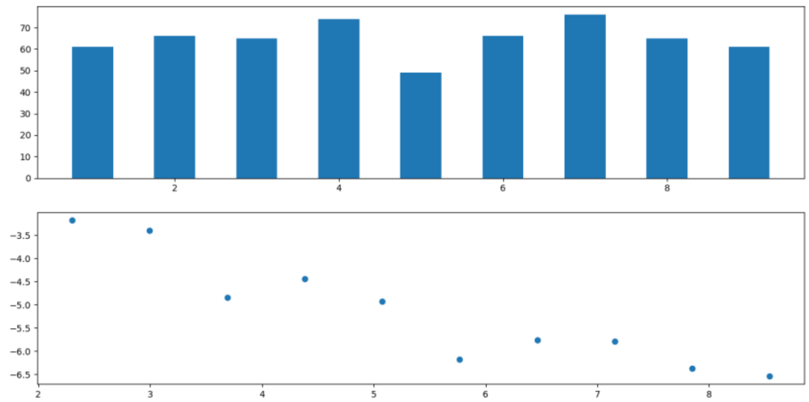
برای تعداد بالا شیب نمودار ۰.۴۵۹۹۳۶۱۲۵۸۹۵۰۶۹۸- خواهد بود. طبق همین می‌توان فرمول داده شده را اثبات کرد. و مشاهده کرد که انحراف با جذر عکس *N* به سمت صفر می‌رود.

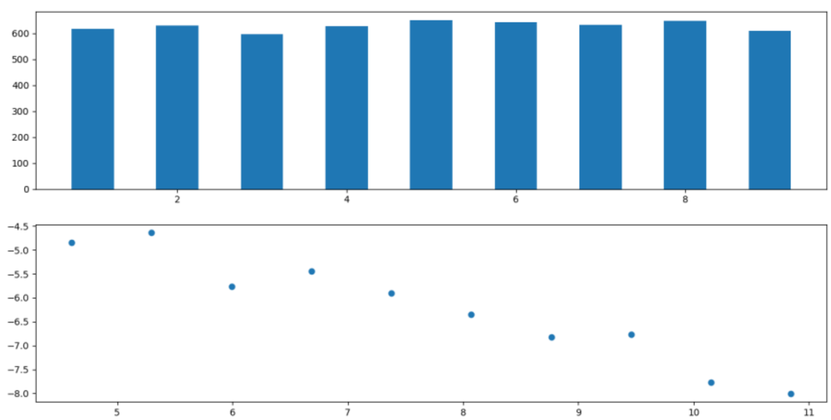
شباهت این سوال با ولنشست دقیقا آن است که همانند ولنشست داریم اعداد رندوم را در بازه ۱ تا ۹ تولید می‌کنیم.

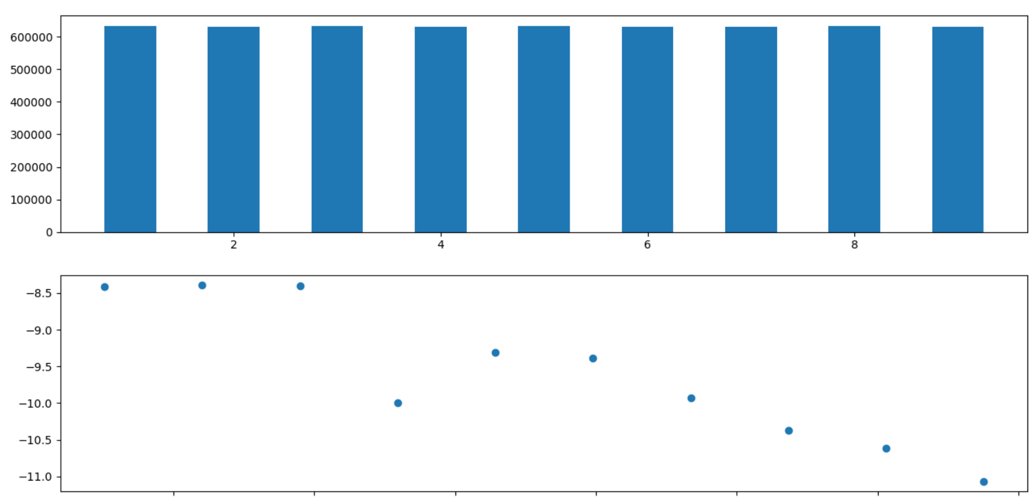
۵. همبستگی

این سوال دقیقا تکرار سوال پیشین است فقط با این تفاوت که یک شرط برای تشخیص عدد پیش از ۴ نیاز داریم. همانند بالا نمودار برای ۱۰ و ۱۰۰ و ۱۰۰۰ و ششصد هزار بار خواهیم داشت:




0 10 20 30 40 50 60 70
2 4 6 8
−3.5 −4.0 −4.5 −5.0 −5.5 −6.0 −6.5
2 3 4 5 6 7 8


0 100 200 300 400 500 600
2 4 6 8
−4.5 −5.0 −5.5 −6.0 −6.5 −7.0 −7.5 −8.0
5 6 7 8 9 10 11


0 100000 200000 300000 400000 500000 600000
2 4 6 8
−8.5 −9.0 −9.5 −10.0 −10.5 −11.0
12 13 14 15 16 17 18

همانند سوال قبل برای شیب خواهیم داشت: ۰.۴۸۵۰۹۴۳۴۸۲۵۷۰۳۱۴۶-

واضح است که در تعداد بالا باز هم تابع توزیع یک نواخت است و شیب با تئوری سازگار می‌باشد.