

ID ссылки: 349351436

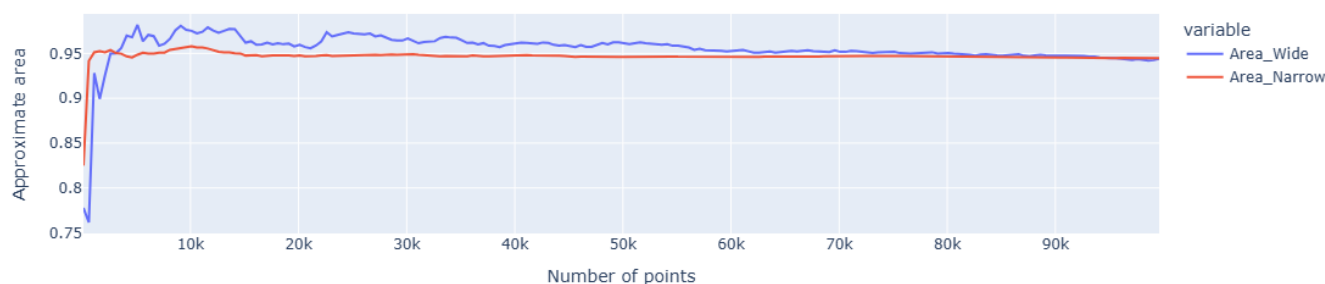
```
import pandas as pd
import plotly.express as px

df = pd.read_csv("results.csv")

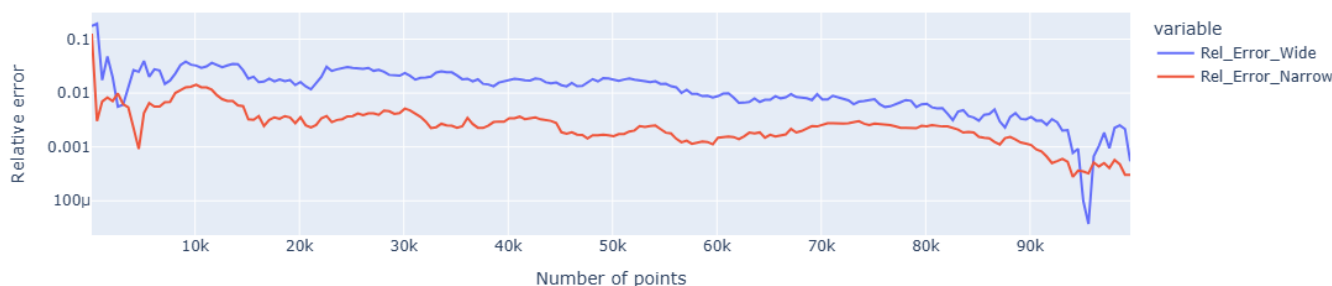
fig_area = px.line(df, x='N', y=['Area_Wide', 'Area_Narrow'],
                   labels={'value': 'Approximate area', 'N': 'Number of points'},
                   title='Monte Carlo: Approximate Area vs N')
fig_area.show()

fig_error = px.line(df, x='N', y=['Rel_Error_Wide', 'Rel_Error_Narrow'],
                   labels={'value': 'Relative error', 'N': 'Number of points'},
                   title='Monte Carlo: Relative Error vs N')
fig_error.update_yaxes(type='log')
fig_error.show()
```

Monte Carlo: Approximate Area vs N



Monte Carlo: Relative Error vs N



Видим, что получаемая площадь при генерации в широкой области нестабильна и сильно меняется от запуска к запуску, хотя в абсолютных значениях скачки уменьшаются при увеличении  $N$ . В отличие от этого при генерации в узкой области имеем стабильный результат (т.е. второй подход более предсказуем, но куда важнее отклонение от ответа). Рассматривая относительное отклонение, видим, что при втором подходе (узкая область) результат практически всегда ближе к идеальному, чем при первом подходе, кроме пары выбросов. Итого можем заключить, что второй подход

предпочтительней, ведь он дает более предсказуемые и более близкие к ответу результаты, кроме редких случаев выбросов. Это происходит тк узкая область уменьшает долю случайных точек, которые не входят в пересечение, что снижает дисперсию оценки.