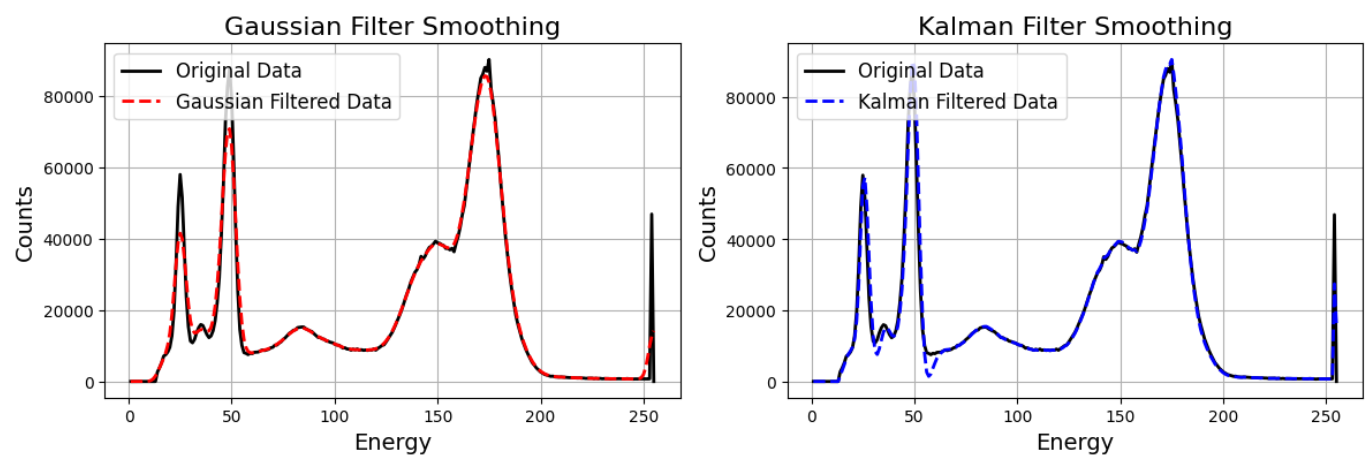
**گزارش کار فاز اول پروژه**

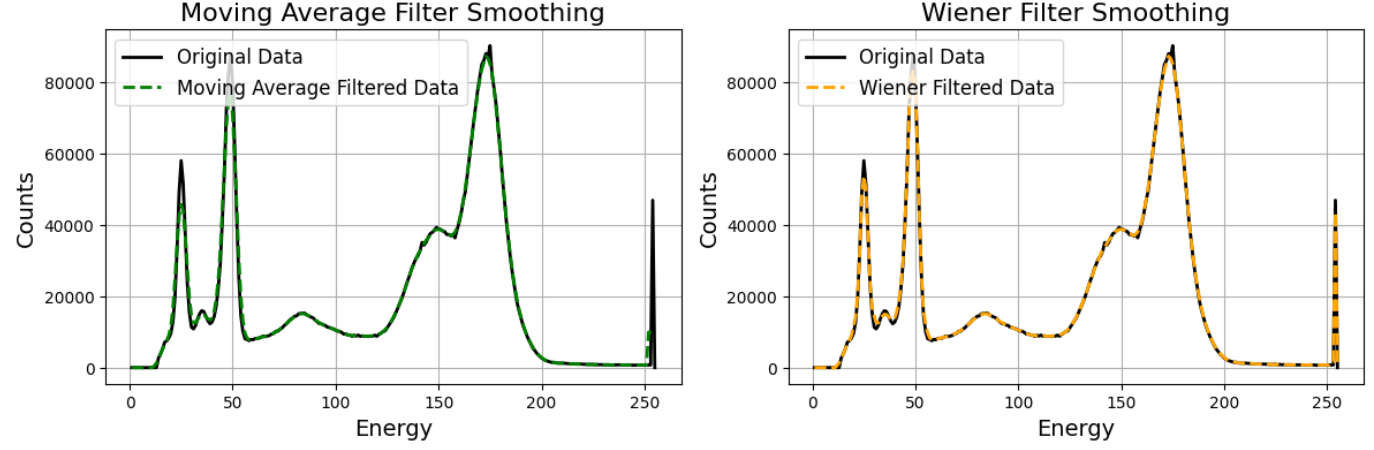
**فاز اول : انجام پروژه هموارسازی طیف گاما**

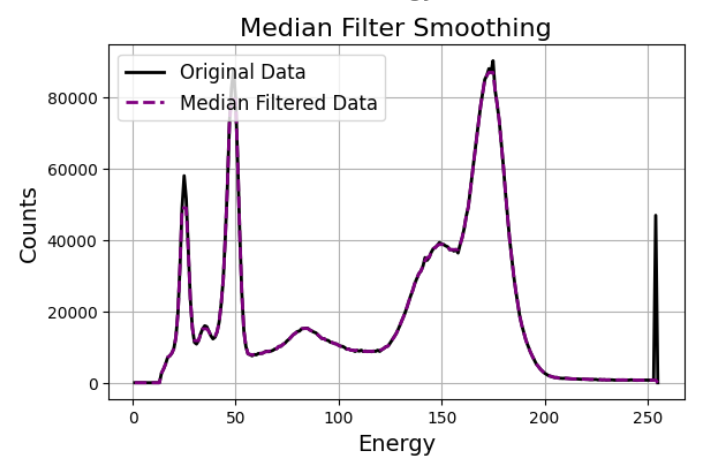
کد Smoothing.ipynb هموارسازی داده یک داده‌ی سری زمانی (counts) را از یک فایل CSV می‌خواند و سپس نویز تصادفی به آن اضافه می‌کند. سپس این داده‌ی نویزی را با استفاده از چندین روش فیلتر کردن مختلف، هموار (smooth) می‌کند و نتایج را نمایش می‌دهد. جزئیات مراحل این کد به شرح زیر است:

1. **خواندن داده‌ها:**
   * داده‌ها از فایلی به نام '15.5.csv' خوانده می‌شوند. ستون اول به‌عنوان انرژی (energy) و ستون دوم به‌عنوان تعداد (counts) استخراج می‌شوند.
2. **افزودن نویز:**
   * نویز گوسی با میانگین ۰ و انحراف معیار ۱۰ به داده‌های counts اضافه می‌شود تا داده‌های نویزی (noisy\_counts) تولید شود.
3. **فیلتر کردن داده‌ها:**
   * **فیلتر گوسی (Gaussian Filter):** از تابع gaussian\_filter1d با سیگما (sigma) برابر با ۲ استفاده می‌شود تا نویز کاهش یابد.
   * **فیلتر کالمن (Kalman Filter):** یک فیلتر کالمن دو‌بعدی تعریف می‌شود که با استفاده از مشاهدات، نویز را تخمین زده و داده‌ها را هموار می‌کند.
   * **فیلتر میانگین متحرک (Moving Average Filter):** داده‌ها با استفاده از میانگین متحرک هموار می‌شوند، جایی که اندازه پنجره برابر ۵ انتخاب شده است.
   * **فیلتر وینر (Wiener Filter):** از تابع wiener برای کاهش نویز استفاده می‌شود.
   * **فیلتر میانه (Median Filter):** از تابع medfilt با اندازه کرنل ۵ برای هموارسازی داده‌ها استفاده می‌شود.
4. **نمایش نتایج:**
   * نتایج هر فیلتر به‌صورت گرافیکی در ۵ نمودار مجزا در کنار داده‌های اصلی نمایش داده می‌شوند.

این کد به شما امکان می‌دهد تا تأثیر فیلترهای مختلف بر داده‌های نویزی را بررسی کنید و نتایج را به‌صورت بصری مقایسه کنید.







## تحلیل مقایسه‌ای نمودارهای هموار‌سازی با فیلترهای مختلف

**مقدمه**

در این نمودارها، نتایج اعمال پنج فیلتر مختلف (گاوسی، کالمن، میانگین متحرک، وینر و میانه) بر روی یک مجموعه داده اولیه (Original Data) نمایش داده شده است. هدف اصلی از اعمال این فیلترها، کاهش نویز و بهبود سیگنال اصلی است.

**مقایسه فیلترها**

1. **فیلتر گاوسی (Gaussian Filter):**
   * **ویژگی‌ها:** به دلیل استفاده از تابع گاوسی، وزن‌دهی به داده‌های اطراف هر نقطه به صورت نرم و صاف انجام می‌شود.
   * **نتایج:** این فیلتر به خوبی نویز را کاهش داده و شکل کلی سیگنال را حفظ کرده است. اما ممکن است برخی از جزئیات ریز سیگنال را نیز از بین ببرد.
2. **فیلتر کالمن (Kalman Filter):**
   * **ویژگی‌ها:** این فیلتر به صورت بازگشتی عمل می‌کند و از اطلاعات قبلی برای پیش‌بینی مقدار فعلی استفاده می‌کند.
   * **نتایج:** فیلتر کالمن به خوبی نویز را کاهش داده و سیگنال را صاف کرده است. همچنین، نسبت به نویزهای ناگهانی مقاوم‌تر است.
3. **فیلتر میانگین متحرک (Moving Average Filter):**
   * **ویژگی‌ها:** این فیلتر با گرفتن میانگین از تعدادی داده متوالی، سیگنال را صاف می‌کند.
   * **نتایج:** این فیلتر به خوبی نویز را کاهش داده است، اما ممکن است لبه‌های سیگنال را کند کند و برخی از جزئیات را از بین ببرد.
4. **فیلتر وینر (Wiener Filter):**
   * **ویژگی‌ها:** این فیلتر با استفاده از یک مدل آماری از سیگنال و نویز، بهینه‌ترین تخمین از سیگنال اصلی را ارائه می‌دهد.
   * **نتایج:** فیلتر وینر به خوبی نویز را کاهش داده و سیگنال را صاف کرده است. همچنین، نسبت به نویزهای رنگی (correlated noise) مقاوم‌تر است.
5. **فیلتر میانه (Median Filter):**
   * **ویژگی‌ها:** این فیلتر به جای گرفتن میانگین، میانه داده‌های متوالی را جایگزین می‌کند.
   * **نتایج:** این فیلتر بسیار موثر در حذف نویزهای پالسی (impulse noise) است و لبه‌های سیگنال را به خوبی حفظ می‌کند. اما ممکن است برخی از جزئیات ریز سیگنال را از بین ببرد.

**نتیجه‌گیری کلی**

انتخاب بهترین فیلتر به نوع داده‌ها، میزان نویز و اهمیت جزئیات سیگنال بستگی دارد.

* **اگر حفظ جزئیات سیگنال مهم است:** فیلتر میانه انتخاب مناسبی است.
* **اگر نویزهای ناگهانی وجود دارد:** فیلتر کالمن عملکرد خوبی دارد.
* **اگر نویز رنگی وجود دارد:** فیلتر وینر انتخاب مناسبی است.
* **برای یک صاف‌سازی کلی:** فیلتر گاوسی یا میانگین متحرک قابل استفاده است.

**نکات اضافی**

* **پارامترهای فیلتر:** هر فیلتر دارای پارامترهایی است که بر عملکرد آن تاثیر می‌گذارند (مثلاً اندازه پنجره در فیلتر میانگین متحرک). انتخاب مناسب این پارامترها بسیار مهم است.
* **تغییرات در سیگنال:** اگر سیگنال دارای تغییرات ناگهانی باشد، فیلترهای میانگین متحرک و گاوسی ممکن است این تغییرات را کاهش دهند.
* **ابعاد داده‌ها:** این نمودارها برای داده‌های یک‌بعدی هستند. برای داده‌های چندبعدی، فیلترهای پیچیده‌تری مورد نیاز است.

**فاز اول : انجام پروژه هموارسازی طیف گاما با استفاده از الگوریتم های هوش مصنوعی**

کد Smoothing\_AI.ipynb یک شبکه عصبی کانولوشنی (CNN) را برای پردازش و تحلیل داده‌های سری زمانی پیاده‌سازی می‌کند. در این مورد، هدف از استفاده از شبکه عصبی کانولوشنی کاهش نویز و هموارسازی داده‌های ورودی است. در ادامه، مراحل اصلی کد توضیح داده شده است:

1. **بارگذاری داده‌ها:**
   * داده‌ها از یک فایل CSV با نام 15.5.csv بارگذاری می‌شوند. داده‌ها شامل دو ستون energy و counts هستند.
2. **نرمال‌سازی داده‌ها:**
   * ستون counts که شامل تعداد مشاهدات یا مقدار داده‌هاست، با استفاده از StandardScaler نرمال‌سازی می‌شود. این کار باعث می‌شود که داده‌ها به شکل نرمال با میانگین صفر و انحراف معیار واحد در بیایند.
3. **آماده‌سازی داده‌ها برای شبکه عصبی کانولوشنی:**
   * یک تابع به نام create\_dataset تعریف می‌شود که داده‌ها را به صورت پنجره‌های کوچک‌تر (با اندازه ۱۰) تقسیم می‌کند. این پنجره‌ها به عنوان ورودی به شبکه عصبی کانولوشنی داده می‌شوند.
   * بعد از این، داده‌های ورودی (X) به صورت سه‌بعدی درمی‌آیند تا با ورودی لایه Conv1D سازگار باشند.
4. **تقسیم داده‌ها به مجموعه‌های آموزشی و تست:**
   * داده‌ها به دو مجموعه‌ی آموزشی (X\_train, y\_train) و تست (X\_test, y\_test) تقسیم می‌شوند.
5. **ساخت مدل شبکه عصبی کانولوشنی:**
   * یک مدل دنباله‌ای (Sequential) ساخته می‌شود که شامل یک لایه Conv1D با ۶۴ فیلتر، یک لایه Flatten برای مسطح‌سازی داده‌ها، و یک لایه Dense برای خروجی است.
   * مدل با استفاده از Adam بهینه‌سازی شده و تابع خطای آن mean\_squared\_error انتخاب می‌شود.
6. **آموزش مدل:**
   * مدل با استفاده از داده‌های آموزشی به مدت ۵۰ دوره (epoch) و با اندازه دسته ۳۲ آموزش داده می‌شود.
7. **پیش‌بینی با استفاده از مدل:**
   * مدل برای پیش‌بینی داده‌های جدید استفاده می‌شود و نتایج به مقیاس اصلی بازگردانده می‌شوند.
8. **نمایش نتایج:**
   * داده‌های اصلی و داده‌های هموارسازی شده (که توسط CNN تولید شده‌اند) با استفاده از کتابخانه matplotlib نمایش داده می‌شوند تا تأثیر شبکه عصبی بر کاهش نویز و هموارسازی داده‌ها نشان داده شود.

این کد نشان‌دهنده استفاده از شبکه‌های عصبی کانولوشنی برای پردازش سری‌های زمانی و بهبود کیفیت داده‌هاست.

