**فاز3 : تشخیص و شناسایی پیک طیف گاما**

این اسکریپت Detect\_Separate\_Peak.ipynb برای تحلیل و پردازش یک مجموعه داده‌های سری زمانی (مقادیر counts در برابر energy) طراحی شده است تا پیک‌های موجود را شناسایی و تحلیل کرده، تکنیک‌های دکانولوشن (deconvolution) را اعمال کند و نتایج را بصورت تصویری نمایش دهد. در اینجا توضیح اجزای اصلی کد آورده شده است:

### ۱. **بارگذاری داده‌ها:**

* اسکریپت داده‌ها را از یک فایل CSV به نام 15.5.csv با استفاده از کتابخانه Pandas بارگذاری می‌کند. آرایه‌های energy و counts به ترتیب داده‌های ستون اول و دوم را ذخیره می‌کنند.

### ۲. **شناسایی و پردازش پیک‌ها:**

* detect\_peaks**:** از تابع scipy.signal.find\_peaks برای شناسایی پیک‌ها در سیگنال counts بر اساس پارامترهایی مانند ارتفاع، فاصله و برجستگی (prominence) استفاده می‌کند.
* filter\_end\_peaks**:** پیک‌های نزدیک به انتهای سیگنال را فیلتر می‌کند تا از اثرات مرزی جلوگیری شود.
* calculate\_peak\_properties**:** ارتفاع پیک‌ها را برای تحلیل بیشتر استخراج می‌کند.

### ۳. **تکنیک‌های دکانولوشن:**

* چندین روش دکانولوشن (مانند **Tikhonov-Miller**، **Van Cittert**، **Gold**، **Muller**، **Richardson-Lucy**) برای حذف تاری یا نویز از سیگنال پیاده‌سازی شده‌اند.
* هر روش سیگنال زیربنایی (background\_) را با استفاده از پیچش‌های تکراری با یک تابع پخش نقطه‌ای (PSF) که به عنوان یک فیلتر گاوسی مدل شده است، تخمین می‌زند.
* boosted\_deconvolution**:** از روش Richardson-Lucy به عنوان روش پایه استفاده می‌کند.
* nonlinear\_fitting**:** با استفاده از روش کمترین مربعات غیرخطی یک تابع گاوسی را به داده‌ها برازش می‌دهد.
* perform\_ica**:** تحلیل مؤلفه‌های مستقل (ICA) را برای استخراج مؤلفه‌های مستقل از سیگنال اعمال می‌کند.
* moving\_average\_filter**:** یک فیلتر میانگین متحرک ساده را پیاده‌سازی می‌کند.
* snip\_clipping**:** یک تکنیک پس‌زمینه‌زدایی را با استفاده از میانگین محلی پیاده‌سازی می‌کند.

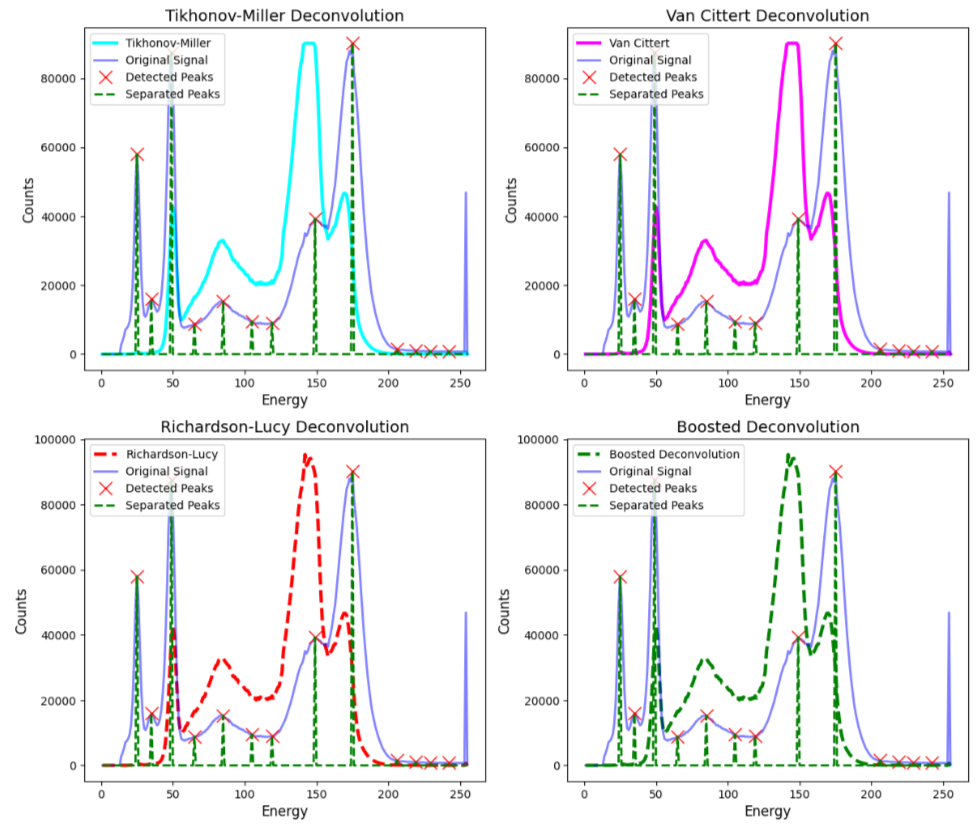
### ۴. **بصری‌سازی:**

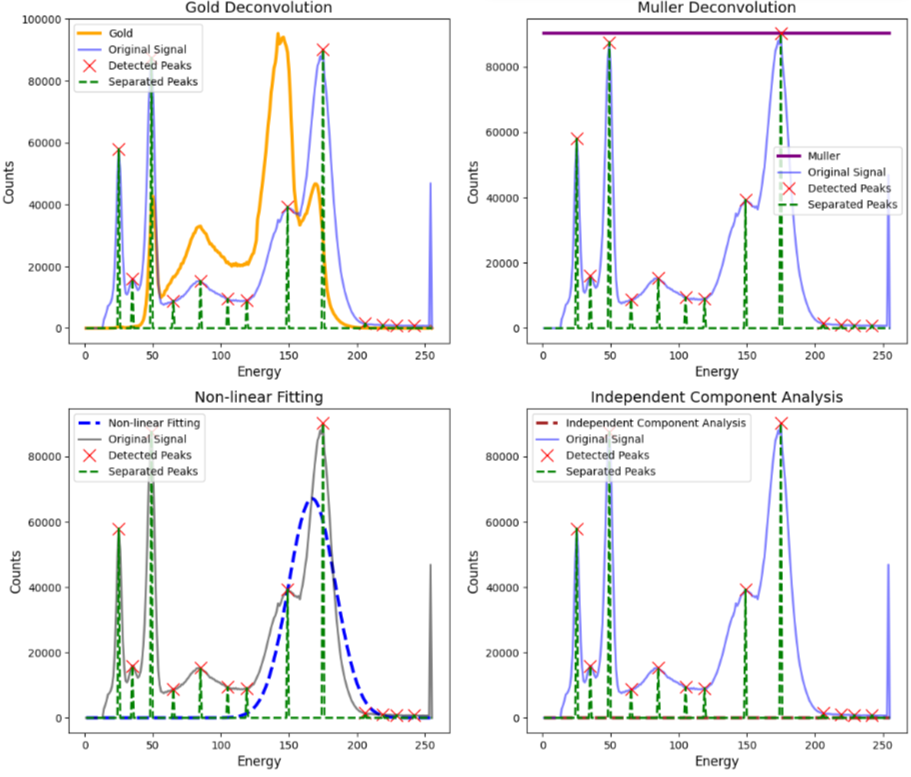
* اسکریپت نتایج هر روش دکانولوشن، سیگنال اصلی، پیک‌های شناسایی‌شده و پیک‌های جداشده را نمایش می‌دهد.
* از کتابخانه matplotlib برای ایجاد نمودارهای زیر (subplot) برای مقایسه بصری سیگنال‌های اصلی و پردازش‌شده استفاده می‌کند.

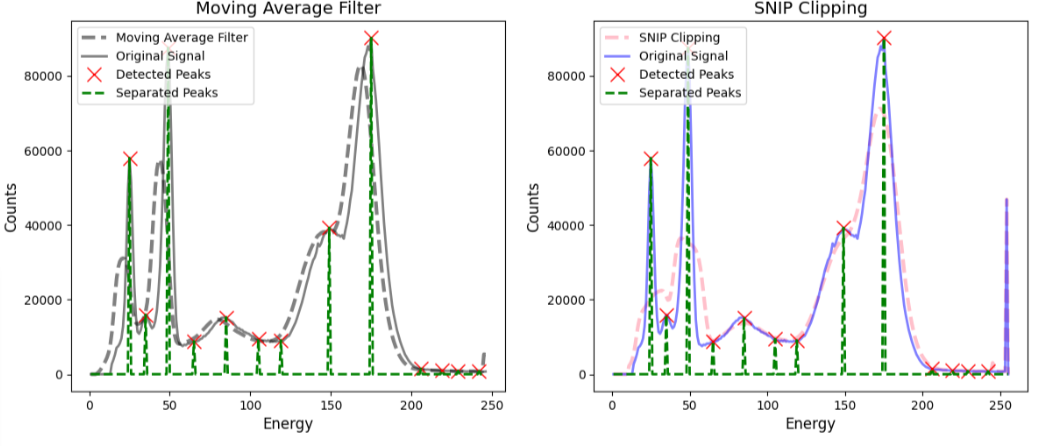
### ۵. **خروجی:**

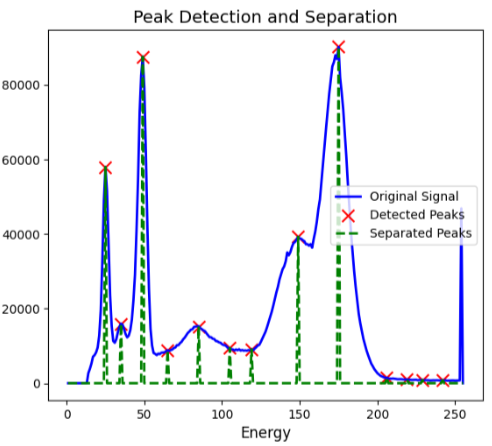
* ارتفاع پیک‌ها و ویژگی‌های آن‌ها برای تحلیل بیشتر چاپ می‌شوند.

این اسکریپت برای تحلیل داده‌های طیف‌سنجی مانند شناسایی و جداسازی پیک‌های هم‌پوشان در اندازه‌گیری‌های تجربی مفید است.







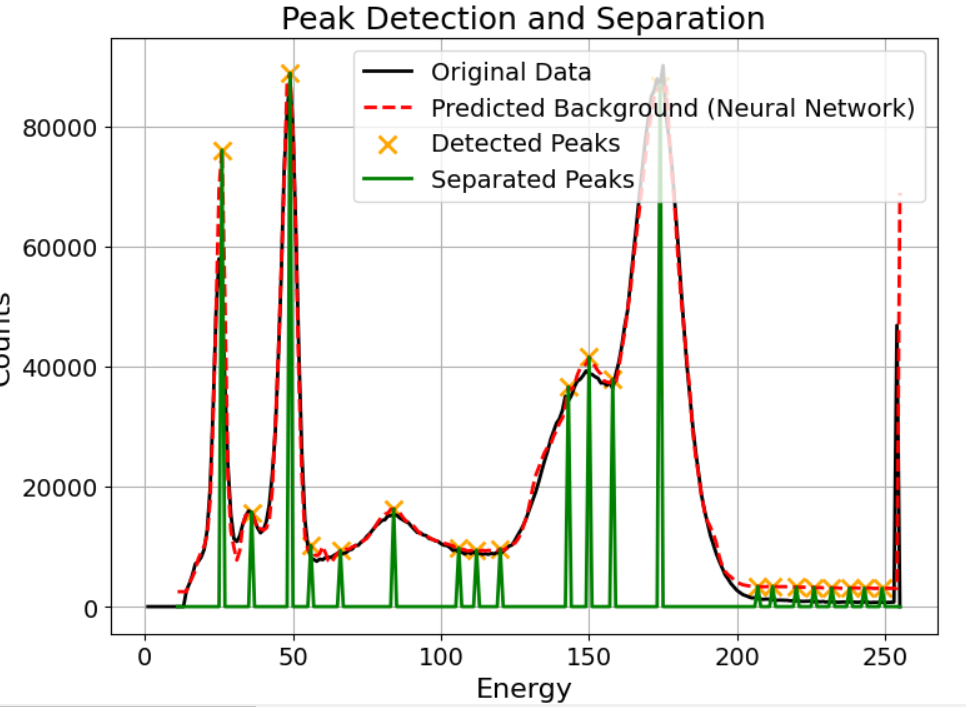


**فاز 3 : تشخیص و شناسایی پیک طیف گاما با استفاده از الگوریتم هوش مصنوعی**

کد  **Detect\_Separate\_Peak\_AI.ipynb**به منظور شناسایی و جداسازی پیک‌ها از داده‌های سیگنال استفاده می‌شود و نتایج را به صورت نمودار نمایش می‌دهد. مراحل اصلی کد به شرح زیر است:

1. **بارگذاری داده‌ها:** داده‌ها از یک فایل CSV بارگذاری می‌شود و مقادیر مربوط به انرژی و تعداد استخراج می‌شود.
2. **شناسایی پیک‌ها:** با استفاده از تابع find\_peaks از کتابخانه scipy.signal، پیک‌ها در داده‌های تعداد شناسایی می‌شوند. این تابع به پارامترهایی مانند ارتفاع، فاصله و بارز بودن پیک‌ها توجه می‌کند.
3. **فیلتر کردن پیک‌های نزدیک به انتهای داده‌ها:** پیک‌هایی که در نزدیکی انتهای داده‌ها قرار دارند، با استفاده از یک فاصله مشخص فیلتر می‌شوند تا از تحلیل نادرست جلوگیری شود.
4. **محاسبه ویژگی‌های پیک‌ها:** ارتفاع پیک‌ها از سیگنال اصلی استخراج شده و به صورت جداگانه ذخیره می‌شود.
5. **ایجاد سیگنال مجزا برای پیک‌ها:** سیگنالی جدید با مقادیر صفر ایجاد می‌شود که تنها مقادیر پیک‌ها در آن قرار داده می‌شود. این سیگنال به نمایش جداگانه پیک‌ها کمک می‌کند.
6. **تنظیمات بصری و نمایش نمودار:** نمودار اصلی داده‌ها، پیک‌های شناسایی شده و پیک‌های جداگانه به صورت بصری نمایش داده می‌شود. این کار به تحلیل داده‌ها و شناسایی ویژگی‌های پیک‌ها کمک می‌کند.
7. **چاپ ویژگی‌های پیک‌ها برای تحلیل:** ارتفاع و ویژگی‌های پیک‌ها چاپ می‌شود تا تحلیل‌های بیشتری بر روی داده‌ها انجام شود.

هدف کلی این کد شناسایی دقیق پیک‌ها و جداسازی آن‌ها از داده‌های سیگنال است. این کار به ویژه برای تحلیل‌های تجربی و علمی که نیاز به شناسایی و بررسی دقیق پیک‌ها دارند، مفید است.



**تحلیل نمودار تشخیص و جداسازی پیک**

**تشریح کلی نمودار:**

این نمودار به صورت گرافیکی فرآیند تشخیص و جداسازی پیک‌ها در یک مجموعه داده را نشان می‌دهد. در این نمودار، محور افقی نشان‌دهنده مقدار انرژی است و محور عمودی تعداد رخدادها (Counts) را نشان می‌دهد.

**اجزای مختلف نمودار:**

* **داده اصلی (Original Data):** خط سیاه ضخیم، داده اصلی و خام را نشان می‌دهد. این داده‌ها معمولاً از یک آزمایش یا شبیه‌سازی به دست می‌آیند.
* **پیک‌های تشخیص داده شده (Detected Peaks):** علامت‌های ضربدر زرد رنگ، موقعیت دقیق پیک‌های شناسایی شده در داده اصلی را نشان می‌دهند. این پیک‌ها معمولاً نقاطی هستند که بیشترین شیب را در داده‌ها دارند.
* **پیک‌های جداسازی شده (Separated Peaks):** خطوط سبز رنگ نقطه‌چین، پیک‌های جداسازی شده را نشان می‌دهند. این خطوط، هر پیک را به صورت جداگانه مدل‌سازی می‌کنند و به ما اجازه می‌دهند تا ویژگی‌های هر پیک را به صورت جداگانه مطالعه کنیم.