

هدف از این برنامه پیدا کردن بیت هایی از سیگنال های ECG می باشد که به درستی برچسب زده شده اند تا محققان با استفاده از داده های صحیح به تحقیق در مورد سیگنال های ECG بپردازند.

ورودی در این برنامه سیگنال های مربوط به 100.dat در پایگاه داده MIT-BIH arrhythmia می باشد.

الگوریتم به کار رفته : NSGA2

کدینگ کروموزوم:

کروموزوم ما شامل مقادیر ۰ و ۱ به تعداد بیت های برچسب دار (cardiologist) ها این برچسب ها را ایجاد کرده اند.

ما برای ورودی از ۱۰۰۰۰ بیت اول سیگنال های 100.dat استفاده کردیم که از این تعداد تنها ۳۵ بیت دارای برچسب می باشند. پس اندازه کروموزوم ما به اندازه ۳۵ می باشد.

در صورتی که کروموزوم مقدار ۰ داشته باشد یعنی آن بیت دارای برچسب معتبر و در غیر این صورت دارای برچسب نامعتبر هست و باید حذف شود.

تابع برازندگی از نوع دوهدفه می باشد.

هدف اول: تعداد بیت های با برچسب غلط در هر کروموزوم را به ما می دهد.

هدف دوم: براساس کلاستر بندی KNN به دست می آید. پس از کلاستر بندی بیت های دارای برچسب به ۶ کلاس L, RB, V, A, N و LB (برای سادگی محاسبات این کلاسها را به ترتیب از ۱ تا ۶ شماره گذاری کردم). در صورتی که این کلاس بندی درست انجام شده باشد مقدار ۱ و در غیر اینصورت مقدار صفر میگیرد. و سپس میانگین این اعداد به عنوان مقدار تابع هدف دوم قرار می گیرد.

برای استفاده از تابع `KNNClassifier` با استفاده از اسکریپت `ExtractFeature` به ازای تمام بیت های موجود (۱۰۰۰۰) ۵ ویژگی استخراج کردم و در ماتریس `Feature` ذخیره کردم. که این ماتریس در تابع `KNNClassifier` جهت کلاس بندی بیت ها فراخوانی می شود.

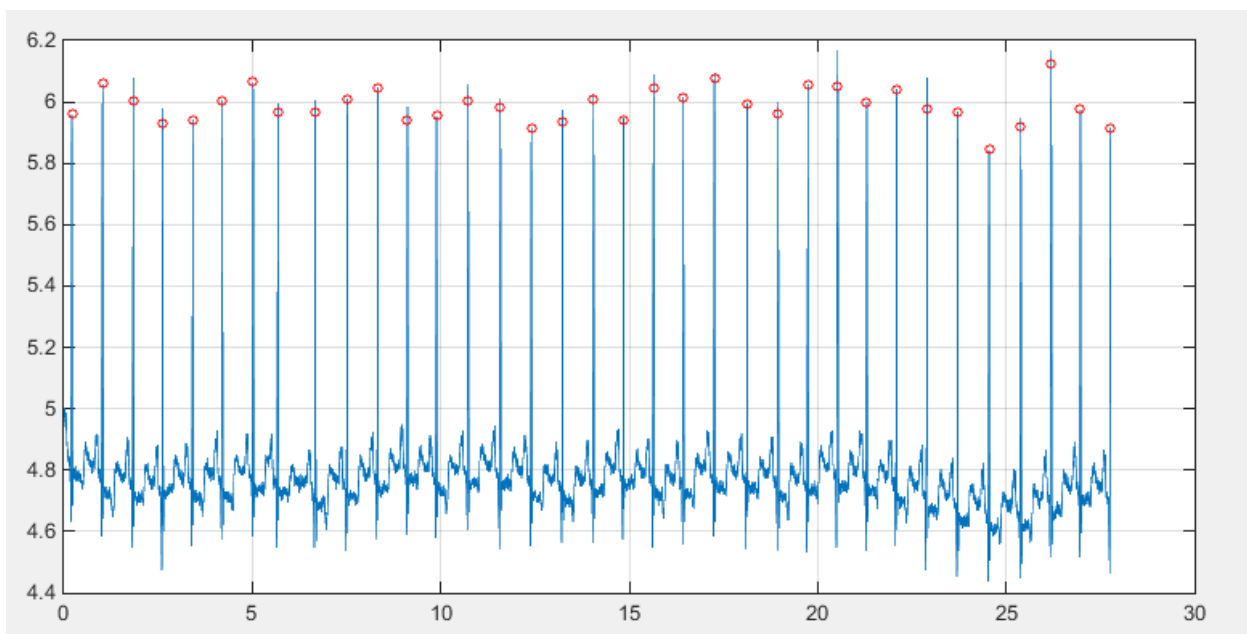
اسکریپت `initData` به ایجاد مقادیر برجسب ها، مقادیر سیگنال، تعداد برجسب ها می پردازد. این اسکریپت فقط یکبار نیاز به اجرا دارد. و نتایج حاصل در ماتریس `data` ذخیره شده است. که من اینکارو کردم. اما شما هم میتونید اجرا کنید.

برای اجرای برنامه شما کافیسیت اسکریپت `NSGA` را اجرا کنید. بقیه توابع از قبل اجرا شده اند.

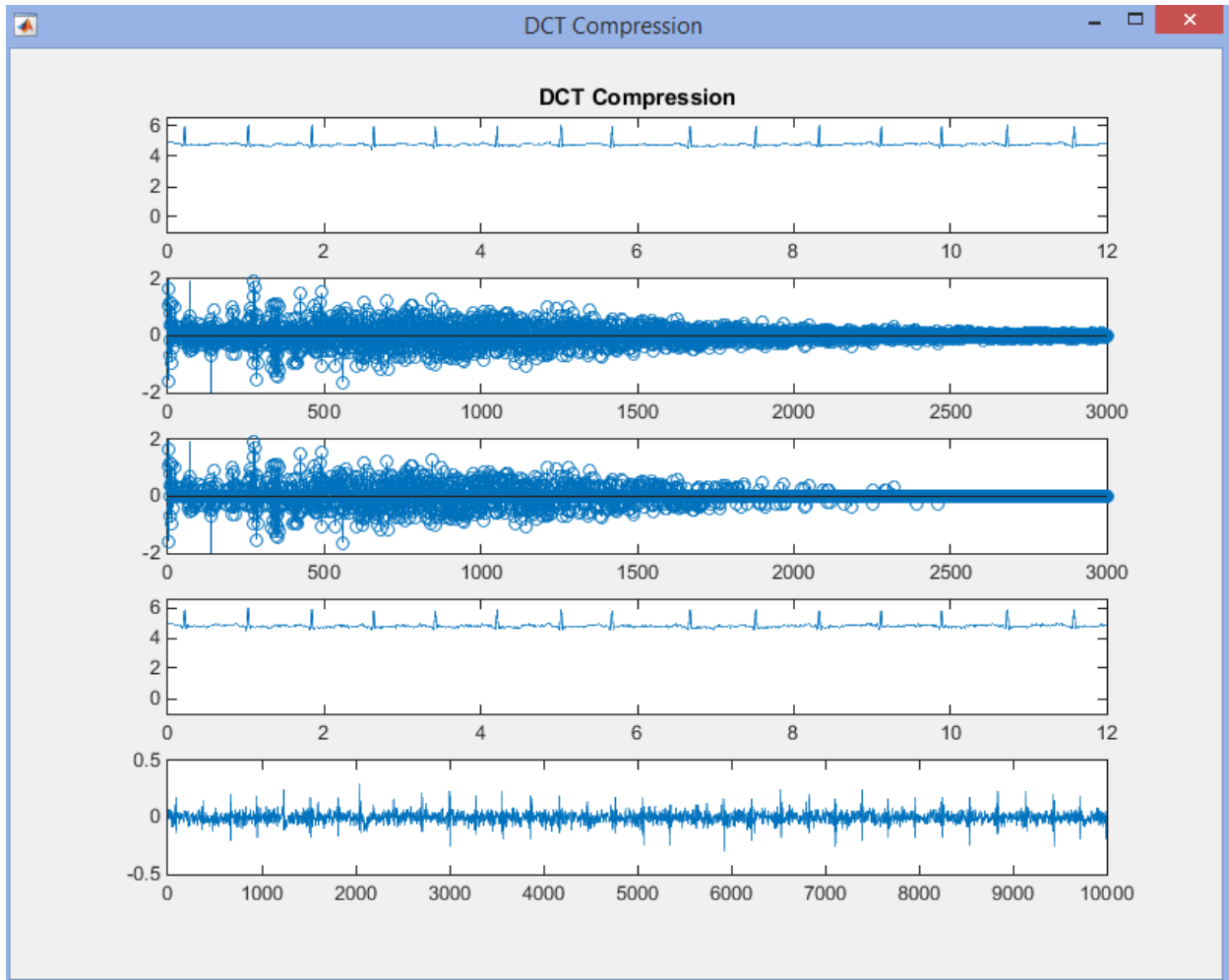
نتایج برنامه در `pareto.pos` قرار می گیرد. و در انتهای برنامه نمایش داده می شود. ضمنا پس از آن `Annotation` های معتبر هم نشان داده می شود.

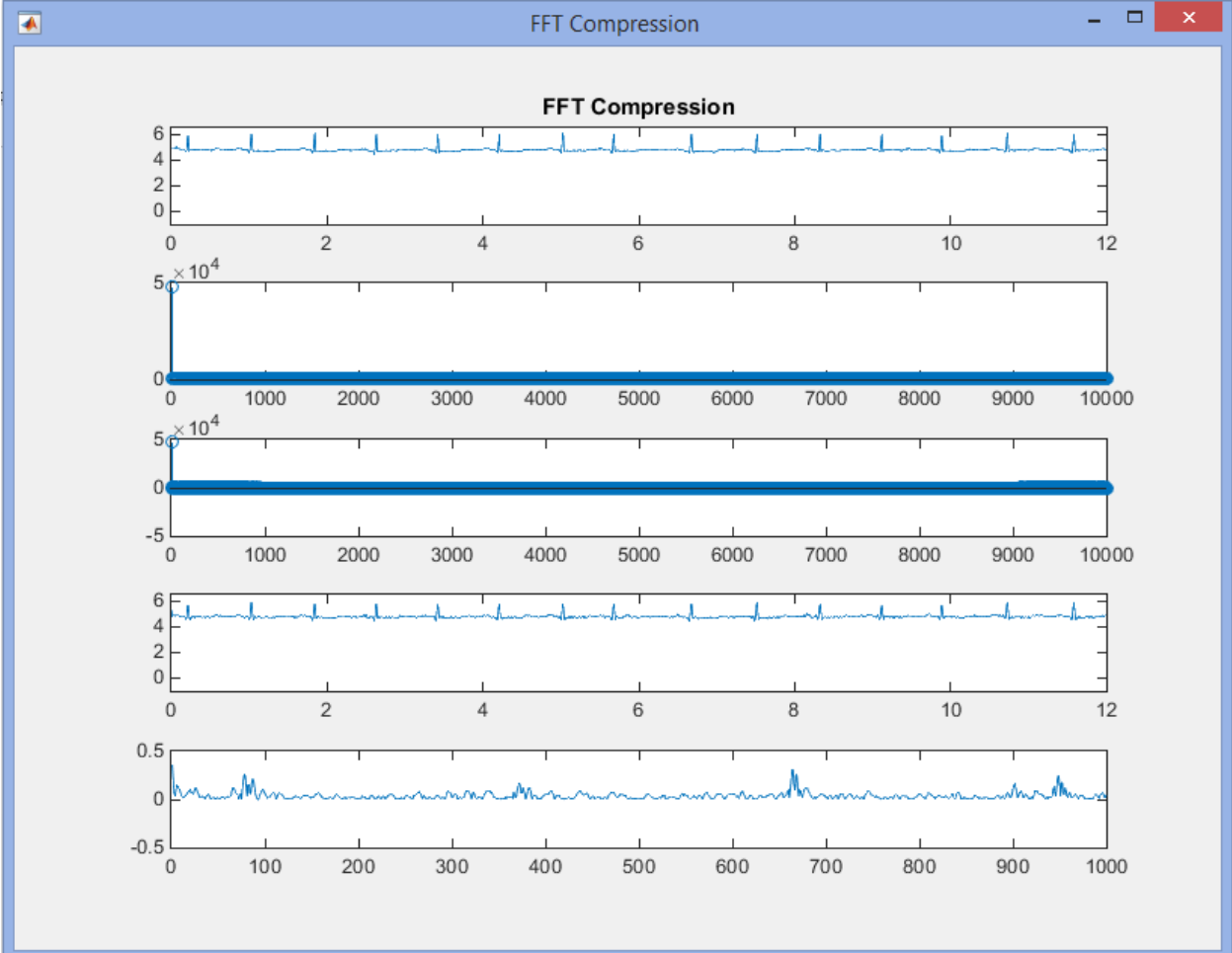
اما برای تست روند برنامه می توانید اسکریپت های `ExtractFeature` و `initData` و `test_with_2d` را هم اجرا کنید.

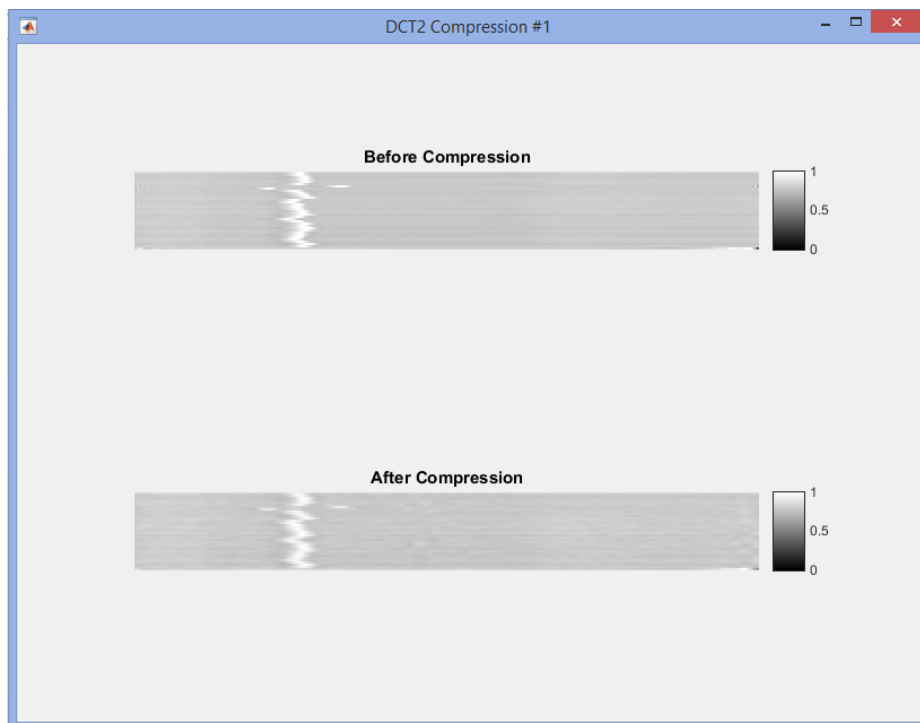
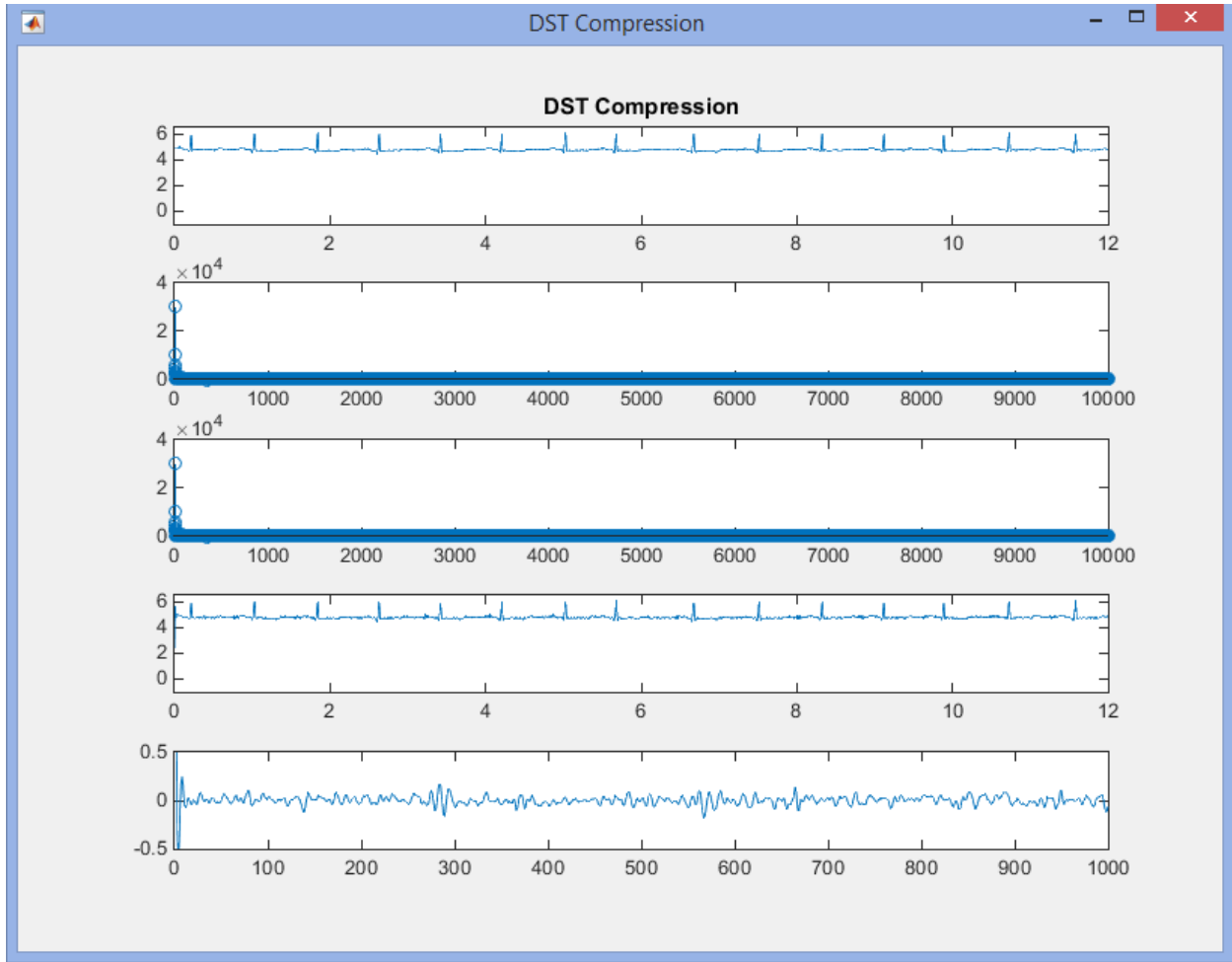
نتایج اجرای اسکریپت های `initData`: نقاط قرمز بیت های برجسب دار

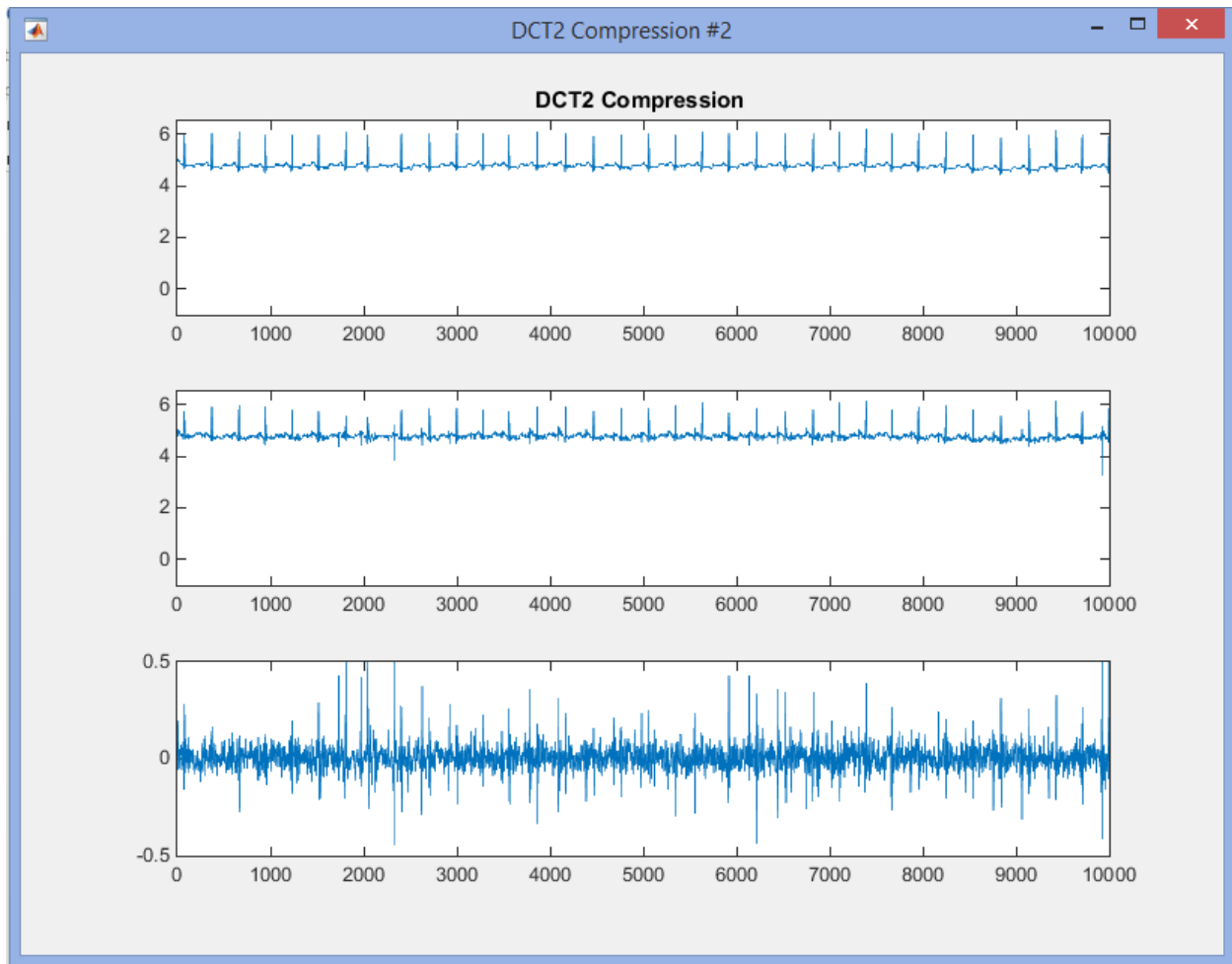


نتایج اجرای ExtractFeature (تعداد زیادی ویژگی استخراج کردم اما تنها ۵ مورد را جهت استفاده در تابع KNNClassifier به کار بردم.



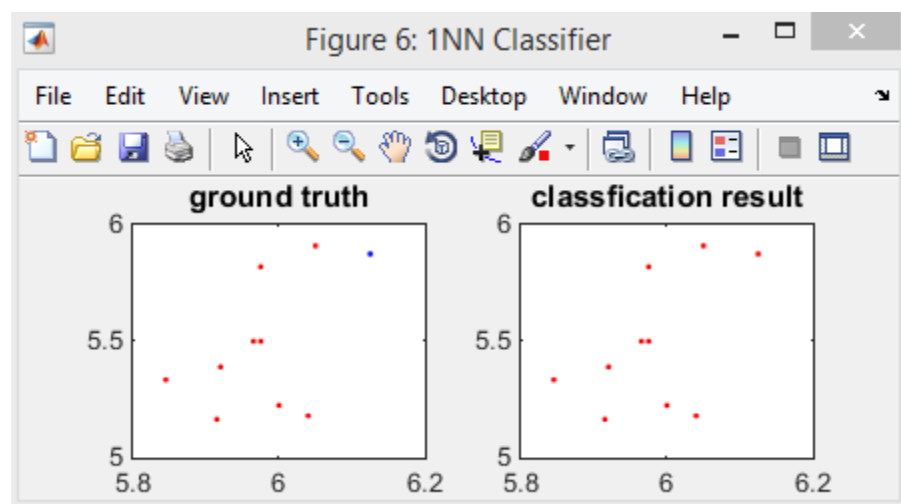




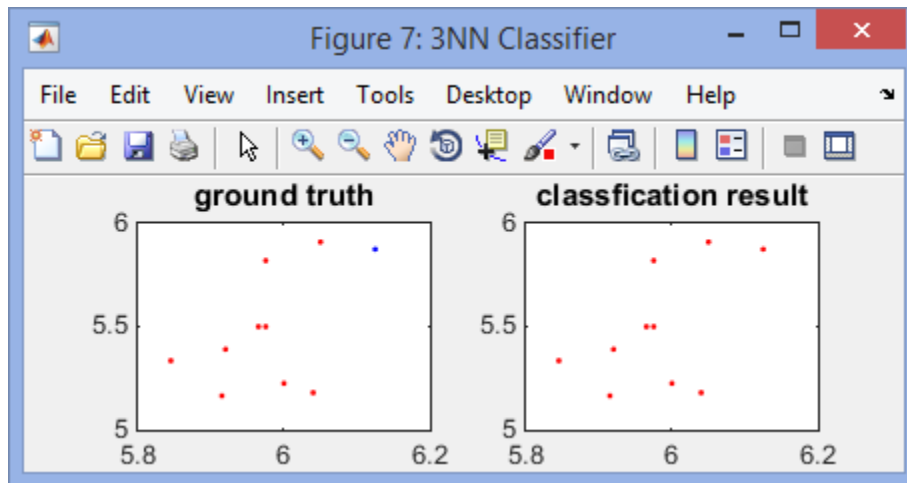


نتایج اجرای `test_with_2d`

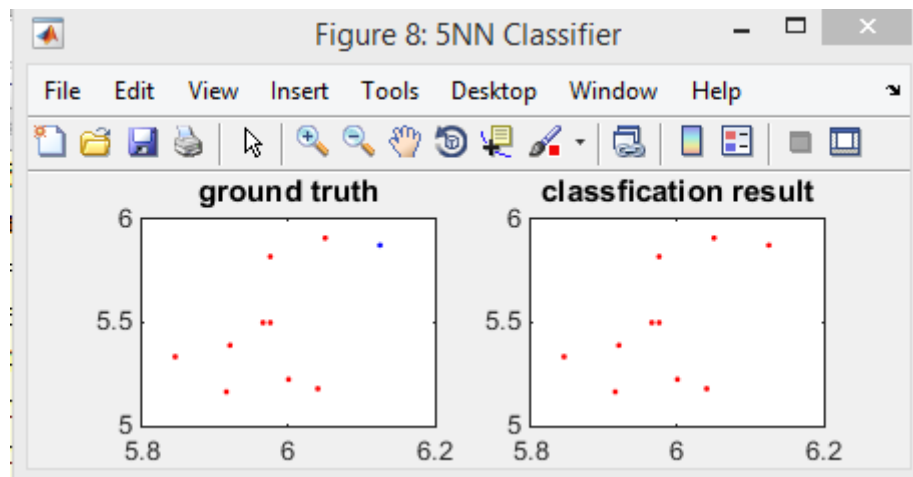
$K=1$ همسایه نزدیک



K=3 همسایه نزدیک



K=5 همسایگی



لازم به ذکر است به دلیل اینکه ما فقط ۳۵ بیت برچسب دار در ۱۰۰۰۰ سیگنال اول داشتیم. این ۳۵ بیت فقط دارای برچسب های N و A بودند. به همین دلیل عملا ما کلاس های دیگر را نداریم. به همین دلیل در شکل های 6 و 7 و 8 تنها دو رنگ ابی و قرمز مشاهده می کنید. موفق باشید.