

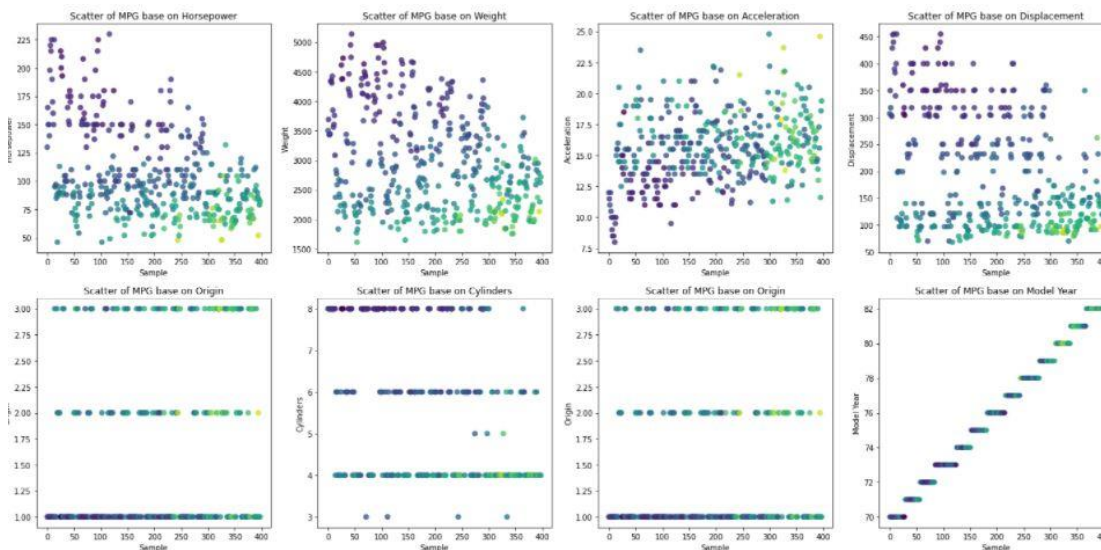
تمرین سوم درس شبکه های عصبی

آماده سازی داده ها

در این تمرین به دنبال تخمین مصرف بنزین اتومبیل های مختلف بر مبنای ویژگی های آن ها هستیم. پس از بارگذاری مجموعه داده های Auto MPG های نامعلوم را مشخص می کنیم. ۶ مقدار نامعلوم در ستون Horse Power می باشد که آن ها را با مقدار میانگین این ستون جایگزین می کنیم.

با توجه به اینکه هدف ما regression می باشد مناسب است داده ها را نرمالیزه کنیم. زمانی که داده ها دارای ویژگی ها با مقیاس های متفاوت باشند عملکرد شبکه کاهش می یابد. این موضوع خصوصا درباره الگوریتم های مبتنی بر فاصله همچون KNN در این تمرین نیز استفاده خواهد شد اثر منفی بیشتری دارد. برای نرمالیزه کردن از تابع MinMaxScaler استفاده می کنیم که هر یک از ستون های مجموعه داده را به فاصله بین صفر و یک انتقال می دهد و به این ترتیب مقیاس ویژگی ها را یکسان می کند. سپس داده ها را به سه مجموعه آموزشی، تست و اعتبار سنجی به صورت تصادفی تقسیم می کنیم.

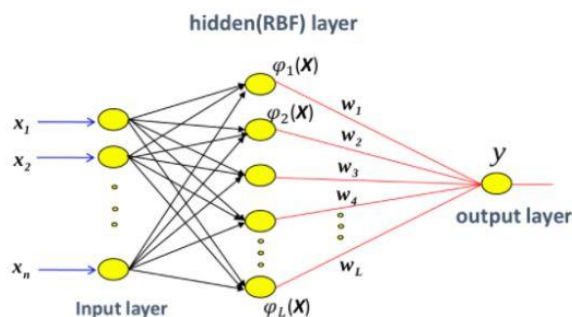
همچنین در شکل زیر نمودار scatter برای میزان مصرف سوخت بر اساس هر یک از ویژگی ها به نمایش گذاشته شده است.



شکل ۱: میزان مصرف سوخت بر اساس هر یک از ویژگی ها

KNN RBF

شبکه RBF در شکل ۱ نمایش داده شده است. این شبکه عمل رگرسیون را به کمک جمع وزن دار توابع گوسی انجام می‌دهد.



شکل ۱: نمای کلی شبکه RBF

در شکل بالا φ_k به صورت زیر تعریف می‌شود:

$$\varphi_k(x) = \exp\left(-\frac{1}{2\sigma_j^2}\|x - \mu_j\|\right) \quad j = 1, 2, \dots, k$$

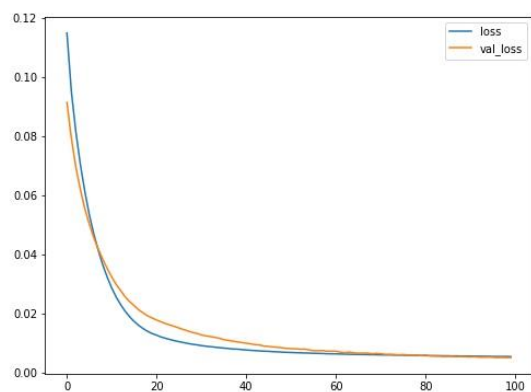
در واقع الگوریتم عمل تخمین را به واسطه مجموعه‌ای از توابع گوسی انجام می‌دهد. برای محاسبه میانگین این توابع گوسی، الگوریتم با اعمال KNN داده‌ها را خوشه بندی کرده و مرکز هر یک از مرکز خوشه‌ها را به عنوان میانگین هر یک از توابع گوسی در نظر می‌گیرد. برای پیاده سازی الگوریتم فوق از کتابخانه rbf_keras استفاده می‌کنیم. پیاده سازی این کتابخانه همانند الگوریتم فوق است.

الگوریتم را با تعداد نورون‌های مختلف پیاده می‌کنیم. در جدول زیر مقادیر ارزیابی برای داده های تست را مشاهده می‌کنید.

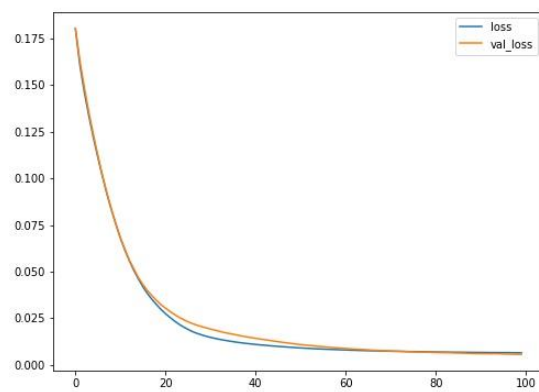
جدول ۱: مقدار ارزیابی MSE برای تعداد نورون‌های مختلف لایه پنهان شبکه KNN-RBF

| #Neuron | MSE |
|---------|---------|
| 5 | 0.00629 |
| 10 | 0.00514 |
| 50 | 0.00531 |
| 100 | 0.00539 |
| 150 | 0.00495 |

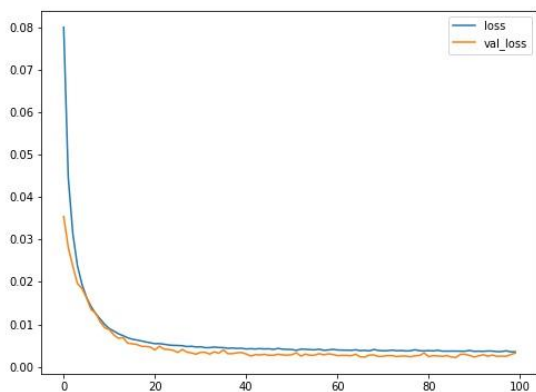
در شکل‌های زیر نمودار خطای mse برای مجموعه داده‌های آموزش و ارزیابی در هر تکرار از الگوریتم نمایش داده شده است.



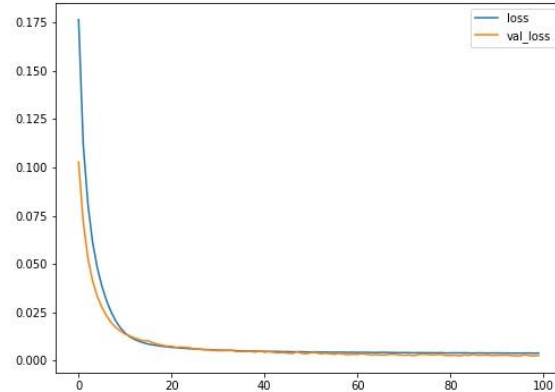
شکل ۳: نمودار خطای KNN-RBF با ۱۰ نورون



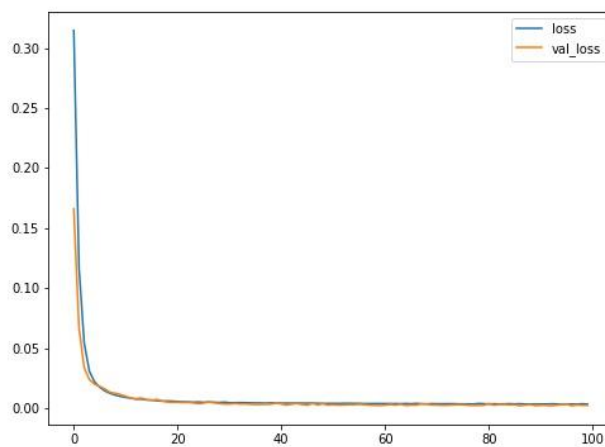
شکل ۲: نمودار خطای KNN-RBF با ۵ نورون



شکل ۵: نمودار خطای KNN-RBF با ۱۰۰ نورون



شکل ۴: نمودار خطای KNN-RBF با ۵۰ نورون

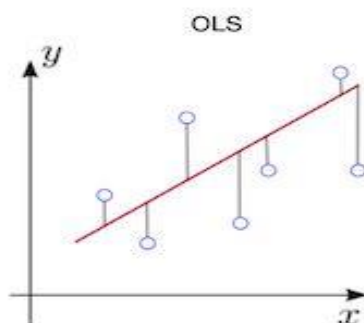


شکل ۶: نمودار خطای KNN-RBF با ۱۵۰ نورون

از جدول فوق می‌توانیم نتیجه بگیریم با افزایش تعداد نورون‌ها میزان mse برای داده‌های تست کاهش پیدا کرده است. از جهت دیگر از شکل‌های بالا به خوبی قابل مشاهده است که با افزایش تعداد نورون‌ها سرعت همگرایی شبکه بیشتر شده است. از جهت دیگر فاصل دو نمودار برای $loss$ و $validation loss$ کاهش پیدا کرده است که به معنی افزایش $generalization$ مسئله می‌باشد. بنابراین با افزایش تعداد نورون‌های لایه مخفی در مدل KNN-RBF ما سه مزیت شامل کاهش خطای تست، افزایش سرعت همگرایی و افزایش $generalization$ به دست آوردیم. بهترین مدل نیز، مدل با ۱۵۰ نورون در لایه مخفی KNN-RBF می‌باشد. در واقع با افزایش نورون‌های مخفی تعداد رگرورها افزایش می‌یابد که بهبود نتایج اشاره شده است

OLS RBF

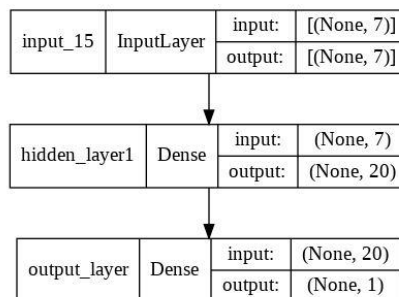
شبکه RBF با کمک OLS به این صورت می‌باشد که به ابتدا به هریک از داده‌ها را مقدار میانگین معادله خط تابع گوسی RBF در نظر می‌گیرد. حال برای تمام نقاط فاصله داده‌های آموزش را تا معادله خط همان طور که در شکل زیر نشان داده شده است محاسبه می‌کند. حال به تعداد نورون‌های لایه مخفی، از بین رگرورهای موجود با کمترین میزان خطا انتخاب می‌کند و مدل RBF را با رگرورهای به دست آمده با کمک $back propagation$ آموزش می‌دهد. این روش به مراتب نسبت به KNN-RBF حجم محاسباتی بیشتری دارد.



شکل ۷: نحوه محاسبه خطای رگرور بر مبنای ols

MLP

در این بخش به دنبال یافتن بهترین معماری MLP برای حل این مسئله هستیم ابتدا از بررسی تعداد لایه‌ها شروع می‌کنیم. در زیر نمونه ایی از معماری MLP با یک لایه مخفی نمایش داده شده است.



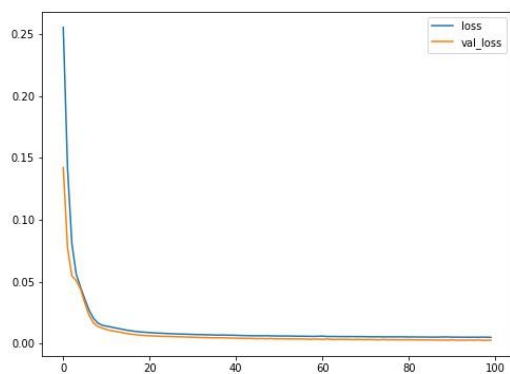
شکل ۷: معماری شبکه MLP با یک لایه مخفی

برای اینکه بتوانیم قضاوت صحیحی داشته باشیم در تمام حالت‌ها تعداد نوروں‌های لایه(ها)ی مخفی را ۲۰ نوروں در نظر می‌گیریم. در جدول زیر مقادیر ارزیابی برای داده‌های تست را به ازای تعداد لایه‌های مختلف مشاهده می‌کنید.

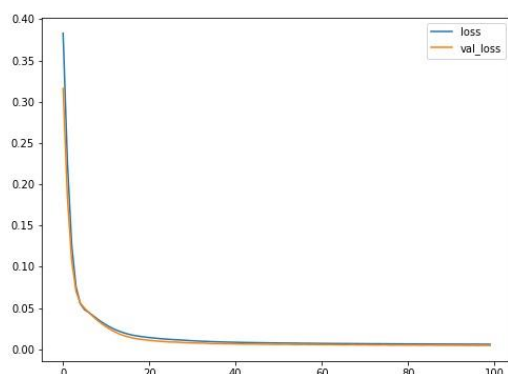
جدول ۲: مقدار ارزیابی MSE برای تعداد لایه‌های مختلف شبکه MLP

| #Hidden Layer | MSE |
|---------------|--------|
| 1 | 0.0051 |
| 2 | 0.0051 |
| 3 | 0.0049 |
| 5 | 0.0052 |
| 10 | 0.0049 |
| 50 | 0.0372 |
| 100 | 0.0368 |

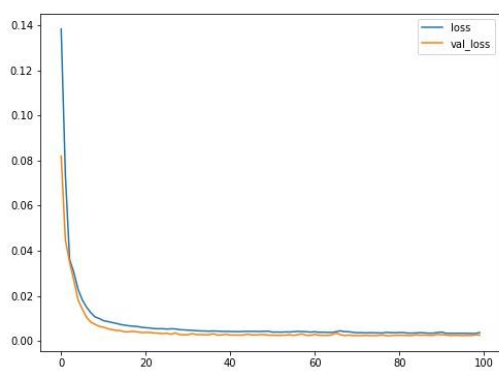
در شکل‌های زیر نمودار خطای mse برای مجموعه داده‌های آموزش و ارزیابی در هر تکرار از الگوریتم نمایش داده شده است.



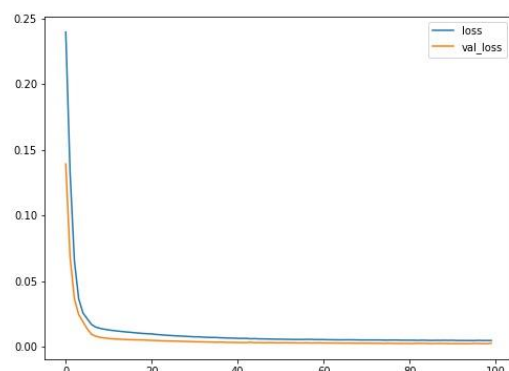
شکل ۹: نمودار خطای MLP با ۲ لایه



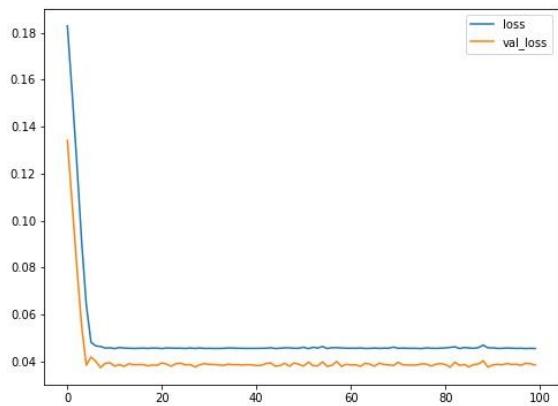
شکل ۸: نمودار خطای MLP با ۱ لایه



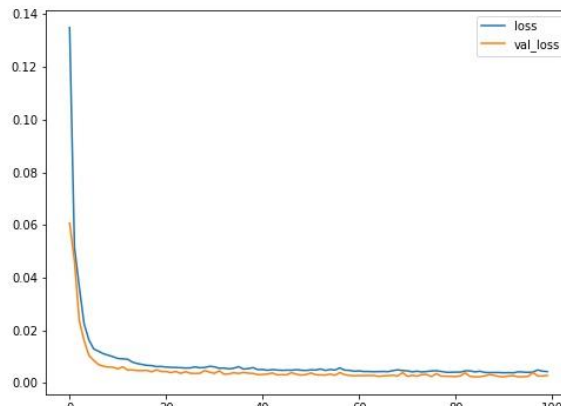
شکل ۱۰: نمودار خطای MLP با ۵ لایه



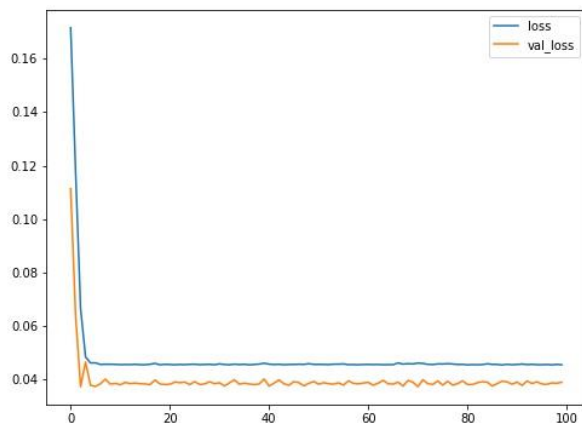
شکل ۹: نمودار خطای MLP با ۳ لایه



شکل ۱۲: نمودار خطای MLP با ۵۰ لایه



شکل ۱۱: نمودار خطای MLP با ۱۰ لایه



شکل ۱۳: نمودار خطای MLP با ۱۰۰ لایه

از جدول فوق می‌توانیم نتیجه بگیریم با افزایش تعداد لایه‌ها لزوماً باعث بهبود نمی‌شود. همان‌طور که مشاهده می‌کنید، تا ۳ لایه میزان mse برای داده‌های تست کاهش یافته اما با ۵۰ و ۱۰۰ لایه میزان خطای mse داده‌های تست افزایش می‌یابد. این یافته برخلاف عملکرد شبکه KNN-RBF است که با افزایش تعداد نورون‌ها عملکرد شبکه بهبود یافت. به‌طور کلی می‌توان گفت افزایش تعداد لایه‌ها و عمیق کردن شبکه تا ۳ لایه کارکرد داشته و عمق دادن بیش از این حد سبب بهبود نشده است.

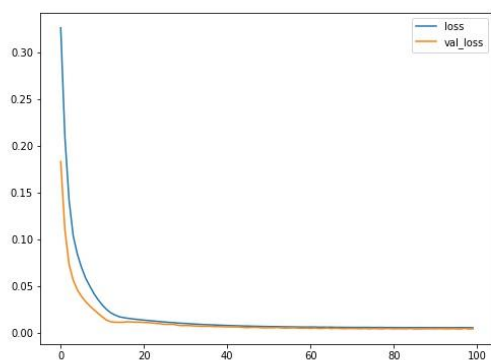
MLP Neurons

با توجه به بخش قبل مدل ۳ لایه گزینه مناسبی می‌باشد زیرا هم از نظر میزان خطای داده‌های تست و سرعت همگرایی و نمودار خطا عملکرد بهتری دارد. در این بخش می‌خواهیم برای این مدل منتخب تعداد نورون‌های مناسب در لایه مخفی را بیابیم. مدل را با تعداد نورون‌های مختلف پیاده می‌کنیم. در جدول زیر مقادیر ارزیابی برای داده‌های تست را مشاهده می‌کنید.

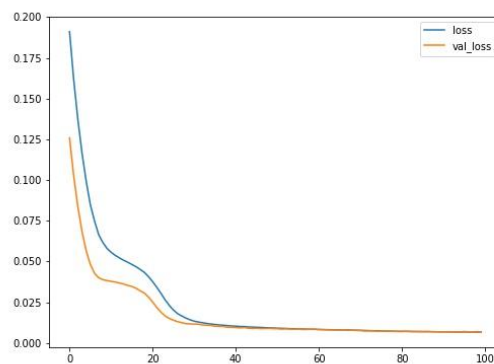
جدول ۳: مقدار ارزیابی MSE برای تعداد نورون‌های مختلف لایه پنهان
شبکه MLP با ۳ لایه

| #Neuron | MSE |
|---------|--------|
| 5 | 0.0051 |
| 10 | 0.0055 |
| 50 | 0.0041 |
| 100 | 0.0048 |
| 150 | 0.0058 |

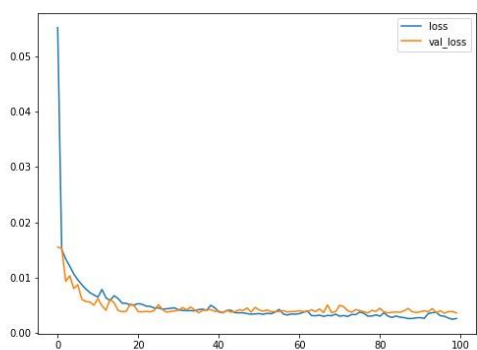
در شکل‌های زیر نمودار خطای mse برای مجموعه داده‌های آموزش و ارزیابی در هر تکرار از الگوریتم نمایش داده شده است.



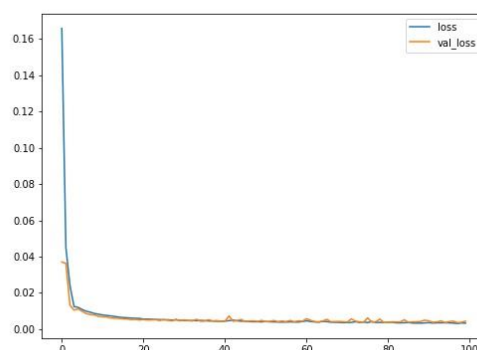
شکل ۱۵: نمودار خطای MLP ۳ لایه با ۱۰ نورون



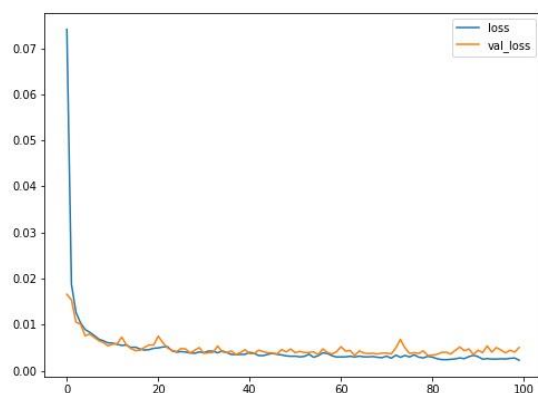
شکل ۱۴: نمودار خطای MLP ۳ لایه با ۵ نورون



شکل ۱۷: نمودار خطای MLP ۳ لایه با ۱۰۰ نورون



شکل ۱۶: نمودار خطای MLP ۳ لایه با ۵۰ نورون



شکل ۱۸: نمودار خطای MLP ۳ لایه با ۱۵۰ نورون

با افزایش تعداد نورون‌های MLP با ۳ لایه مشاهده می‌کنیم مقدار mse داده‌های تست تا ۵۰ نورون کاهش یافته است. همچنین سرعت همگرایی بهبود یافته است. از جهت دیگر فاصله بین نمودارهای val_loss و loss نیز تا ۵۰ نورون کاهش یافته که به معنای بهبود generalization مدل می‌باشد. افزایش بیش از این مقدار بهبود خاصی در عملکرد مدل ندارد. بنابراین مدل منتخب دارای ۳ لایه هر یک با ۵۰ نورون می‌باشد. حال به سراغ بررسی سرعت یادگیری می‌رویم.

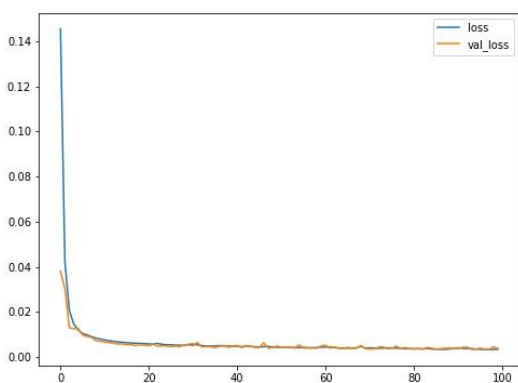
نرخ یادگیری

در این بخش می‌خواهیم برای مدل منتخب به دست آمده در قسمت قبل که یک MLP با ۳ لایه با ۵۰ نورون می‌باشد. بهترین نرخ یادگیری را بیابیم. در جدول و شکل‌های زیر روندی مشابه بخش‌های قبل برای یافتن بهترین گزینه را پیموده‌ایم.

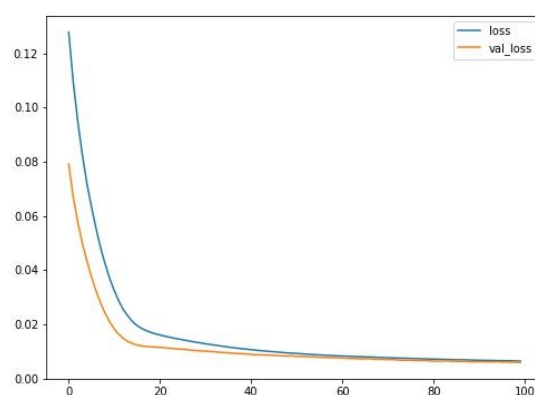
جدول ۴: مقدار ارزیابی MSE برای نرخ‌های یادگیری مختلف

مدل ۳ لایه با ۵۰ نورون

| Learning Rate | MSE |
|---------------|--------|
| 1e-4 | 0.0067 |
| 1e-3 | 0.0040 |
| 1e-2 | 0.0052 |
| 1e-1 | 0.0048 |
| 0.9 | 0.1454 |
| 1 | 0.536 |



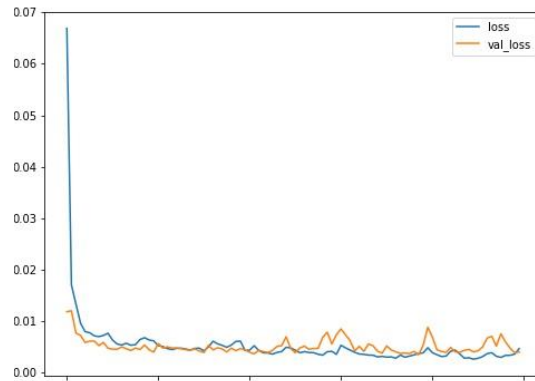
شکل ۲۰: نمودار خطای MLP ۳ لایه با ۵۰ نورون
با نرخ یادگیری $1e-3$



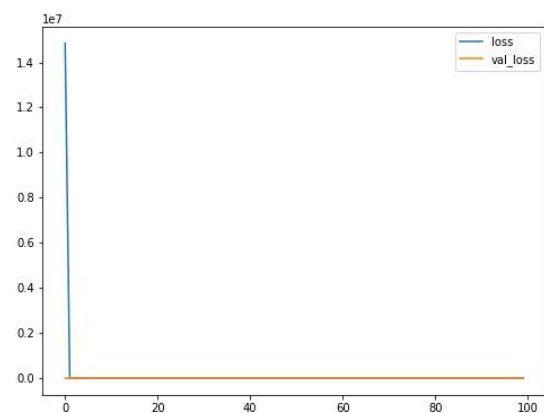
شکل ۱۹: نمودار خطای MLP ۳ لایه با ۵۰ نورون
با نرخ یادگیری $1e-4$



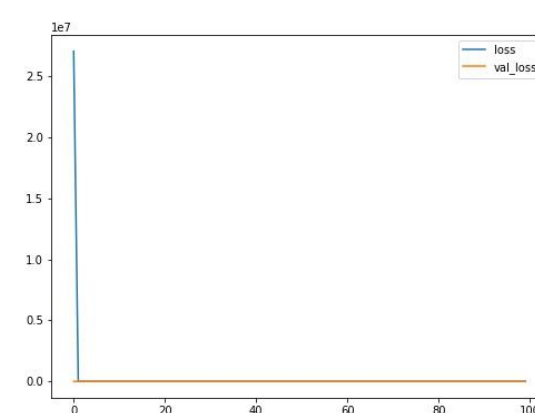
شکل ۲۲: نمودار خطای MLP ۳ لایه با ۵۰ نورون
با نرخ یادگیری $1e-1$



شکل ۲۱: نمودار خطای MLP ۳ لایه با ۵۰ نورون
با نرخ یادگیری $1e-2$



شکل ۲۴: نمودار خطای MLP ۳ لایه با ۵۰ نورون
با نرخ یادگیری 1



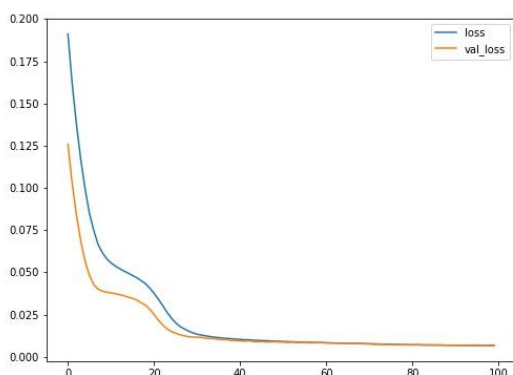
شکل ۲۳: نمودار خطای MLP ۳ لایه با ۵۰ نورون
با نرخ یادگیری 0.9

همان طور که مشاهده می‌کنید عملکرد شبکه با نرخ یادگیری 0.001 هم از نرخ خطای داده‌های تست و هم نمودارهای loss از دو نظر سرعت همگرایی و generalization عملکرد مناسب‌تری دارد و افزایش نرخ یادگیری بیش از این مقدار کارکردی ندارد.

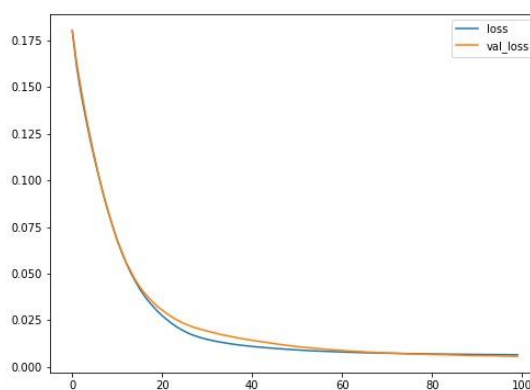
بنابراین مدل منتخب بخش MLP یک شبکه با ۳ لایه هر یک با ۵۰ نورون است که با نرخ یادگیری 0.001 آموزش می‌بیند.

مقایسه

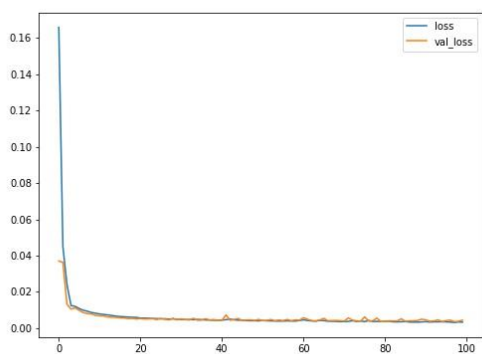
در رابطه با حالت های مختلف معماری شبکه MLP می توان باتوجه به مشاهدات گفت که مدل در ابتدا که تعداد لایه های کم یا تعداد نورون های کمی دارد عملکرد ضعیفی دارد. به علاوه با افزایش بیش از اندازه پیچیدگی شبکه کاهش می یابد. و در بین این دو حالت مدل مناسب وجود دارد. دا واقع باید بین **complexity** و **simplicity** مدل یک مصالحه وجود داشته باشد. اما بر خلاف شبکه MLP، شبکه RBF با افزایش تعداد نورون های لایه مخفی عملکرد شبکه به طور مداوم بهبود پیدا کرده است.



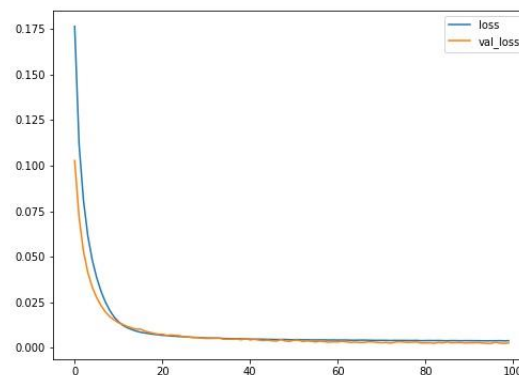
شکل ۲۶: نمودار خطای MLP ۳ لایه با ۵ نورون



شکل ۲۵: نمودار خطای KNN-RBF با ۵ نورون



شکل ۲۷: نمودار خطای MLP ۳ لایه با ۵۰ نورون



شکل ۲۷: نمودار خطای KNN-RBF با ۵۰ نورون

همان طور که در شکل های بالا مشاهده می کنید، شبکه RBF با تعداد پارامترهای کمتر می تواند به نتایج بهتری از شبکه MLP برسد اما شبکه MLP با عمق و معماری مناسب عملکرد بهتری نسبت به شبکه RBF دارد. می توان گفت شبکه RBF به نورون های کمی برای حل مساله نیاز دارد و می توان یک پاسخ نزدیک به بهترین پاسخ با **generalization** بالا بدهد و افزایش تعداد نورون های شبکه تاثیر مثبتی به طور مداوم دارد اما اثر بهبود پیچیده کردن شبکه RBF کم تر از شبکه MLP است. از جهتی شبکه MLP سرعت همگرایی بهتری دارد و با وجود اینکه با تعداد نورون های کم عملکرد ضعیفی دارد، می توان با افزایش پیچیدگی شبکه به نتایج بهتری هم از نظر خطای داده های تست، سرعت همگرایی و **generalization** رسید.

OLS RBF

در برداشت دیگری از OLS، مجموعه رگرسورها که هر یک از در نظر گرفتن یکی از داده‌ها به عنوان میانگین در نظر گرفته شده اند. یک ماتریس ایجاد می‌کند که با تجزیه ماتریس به کمک روش گرام اشمیت بردارهای متعامد این فضا را می‌یابد حال میزان کاهش خطا محاسبه می‌شود و از بین رگرسورها آن‌هایی که دارای بیشترین اثر در کاهش خطا هستند محاسبه می‌شود [1].