|  |  |  |
| --- | --- | --- |
|  | به نام خدا |  |
| **دانشگاه تهران**  **دانشکده‌ مهندسی برق و کامپیوتر**  **شبکه های عصبی و یادگیری عمیق**  **مینی‌پروژه سری ۲** | | |

|  |  |
| --- | --- |
| مهسا قزوینی نژاد - نازنین صبری | نام و نام خانوادگی |
| ۸۱۰۱۹۸۳ - ۸۱۰۱۹۸۳۱۲ | شماره‌ دانشجویی |
| ۱۳ خرداد ۱۳۹۹ | تاریخ ارسال گزارش |

­

**فهرست گزارش سوالات**

[سوال ۱ – طراحی شبکه‌های عصبی 2](#_Toc41727095)

[سوال ۲ – نقصان دادگان 2](#_Toc41727096)

[سوال 4 – عنوان سوال 3](#_Toc41727097)

# سوال ۱ – طراحی شبکه‌های عصبی

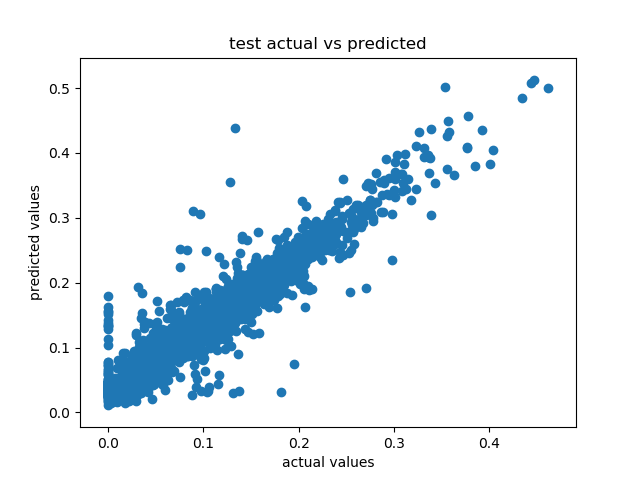
سوال ۱

برای این سوال ابتدا در شکل ۱.۱ چگونگی تقسیم داده‌ها را نشان می‌دهیم. همانطور که می‌توان دید در کد با شروع از نمونه ۱۲ ام، ۱۱ نمونه قبلی را به عنوان ویژگی‌های ورودی در X (ورودی مدل) ریخته و مقدار آلودگی در ساعت دوازدهم در Y (مقداری که مدل پیش‌بینی می‌کند) قرار می‌گیرد.

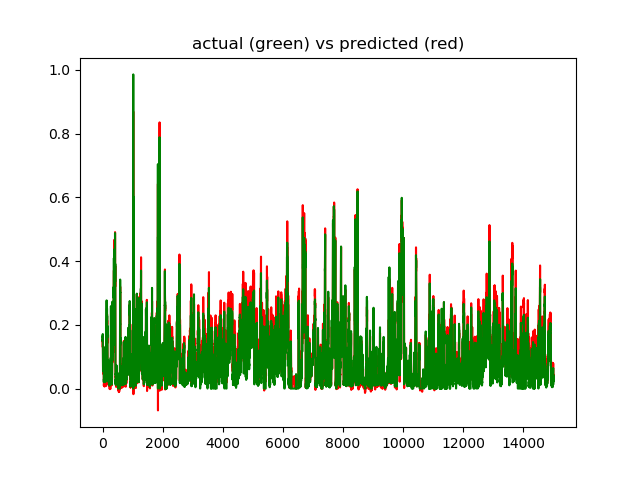


شکل ۱.۱: تابع جداکننده‌ی داده‌ها به مجموعه‌ی ورودی و خروجی مدل

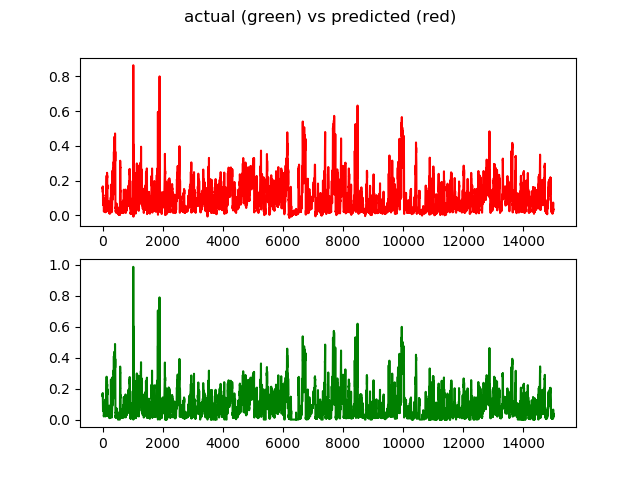
برای نمایش دقت مدل نیز ۳ شکل می‌سازیم. شکل ۱.۲ نمونه‌ی اول است که در آن مقدار حقیقی و مقدار پیش‌بینی شده برای زمان واحدی را به صورت scatter نمایش می‌دهد. در حالت ایده‌آل دوست داریم که نمودار به صورت خطی گذرنده از مبدا با شیب ۴۵ درجه باشد، یعنی مقدار پیش‌بینی شده و مقدار حقیقی دقیقا یکی باشند. شکل ۱.۳ یک نمونه دیگر برای بررسی میزان مطلوب بودن مدل است که در آن نمودار واقعی و پیش‌بینی شده با رنگ‌های مختلف به صورت سری زمانی روی هم رسم می‌شوند، اما همان‌طور که می‌توان دید این روش نمایش خیلی گویا نیست و تشخیص دو نمودار از هم دشوار است. به همین دلیل به سراغ شکل ۱.۴ می‌رویم که همان ایده‌سری زمانی‌های قبل است اما این بار با این تفاوت که به جای کشیدن آن‌ها روی هم آن‌ها را زیر هم می‌کشیم. این بار علاقه داریم که دو نمودار کاملا شبیه هم باشند. در این نمودارها باید به بازه‌ی تغییرات محورهای عمودی دقت کرد.



شکل ۱.۲: نمودار scatter plot برای بررسی میزان مطلوب بودن پیش‌بینی‌های انجام شده



شکل ۱.۳: نموداری برای بررسی میزان مطلوب بودن پیش‌بینی‌ها با رسم آن‌ها به شکل time series روی یک نمودار



شکل ۱.۴: نسخه‌ای مشابه شکل ۱.۳ با این تفاوت که نمودارها را زیر یکدیگر رسم می‌کنیم (به جای روی یک نمودار)

سوال ۲

در این بخش به طراحی شبکه‌های عصبی خواسته شده می‌پردازیم. از آنجا که هیچ چیزی درباره‌ی این مدل‌ها و چگونگی طراحی آن‌ها بیان نشده است، طراحی هر یک را به صورت دلخواه و به صورت ساده انجام می‌دهیم. شکل‌های ۱.۵، ۱.۶ و ۱.۷ به ترتیب هر یک از مدل‌های RNN، GRU و LSTM را نمایش می‌دهند. تعداد ایپاک‌های آموزش را برابر با ۲۰ قرار می‌دهیم در این صورت داریم. برای مدل‌های گزارش شده در این قسمت تابع بهینه‌سازی adam و تابع هزینه (خطا) MAE را در نظر می‌گیریم.

برای هر یک از حالات نمودارهای میزان loss در حین آموزش را رسم می‌کنیم تا ببینیم خطا به چه شکل کاهش پیدا می‌کند و در کدام نمودار سریع‌تر به حداقل مقدار خود می‌رسد و تقریبا ثابت می‌شود. مدت زمان آموزش هر یک را نیز به کمک توابع time در پایتون محاسبه می‌کنیم. نتایج زمان‌های اجرا در جدول ۱.۱ نمایش داده شده است و نمودارهای میزان خطا در شکل‌های ۱.۸، ۱.۹ و ۱.۱۰ گزارش شده‌اند.



شکل ۱.۵: کد مدل LSTM

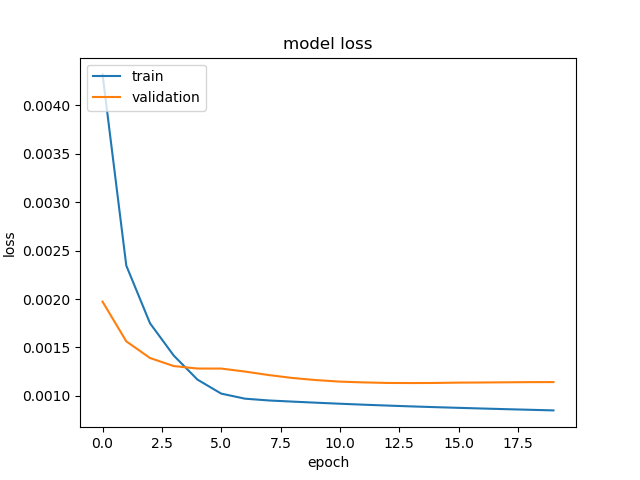


شکل ۱.۶: کد مدل GRU

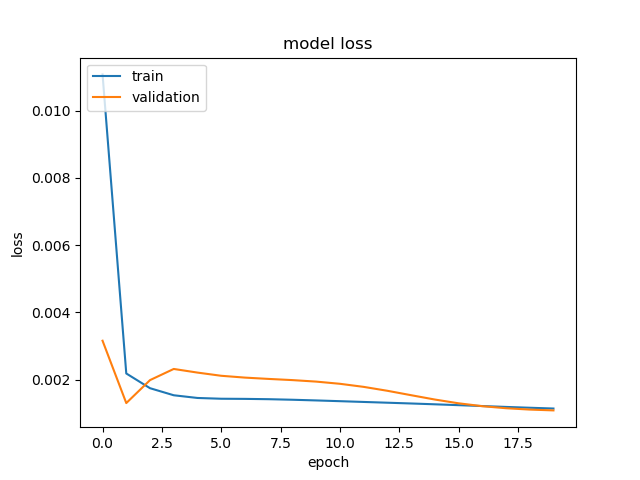


شکل ۱.۷: کد مدل RNN

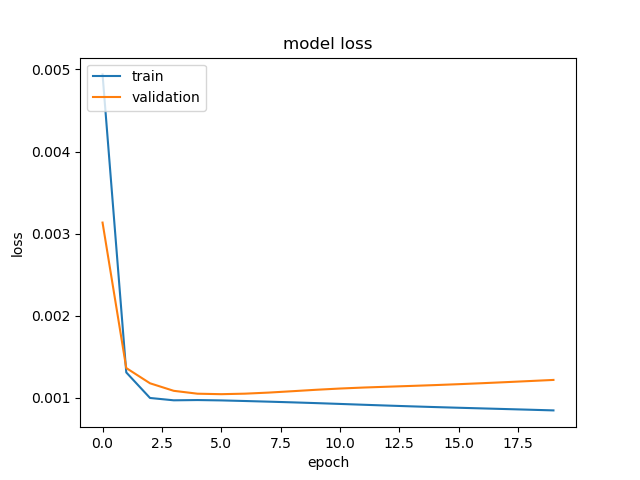
|  |  |
| --- | --- |
| **Model** | **Run time (ms)** |
| LSTM | 168.30 |
| GRU | 120.23 |
| RNN | 44.79 |



شکل ۱.۸: نمودار تغییرات میزان خطا برای مدل LSTM



شکل ۱.۹: نمودار تغییرات میزان خطا برای مدل RNN



شکل ۱.۱۰: نمودار تغییرات میزان خطا برای مدل GRU

با توجه به این نتایج می‌توان دید که مدل RNN کمتر مدت زمان آموزش را دارد اما میزان خطای آن به اندازه‌ی دو مدل دیگر کاهش پیدا نمی‌کند. بیشتر مدت زمان آموزش نیز متعلق به LSTM است که عملکرد آن مشابه GRU اما اندکی بهتر است.

سوال ۳

برای این سوال کدی می‌زنیم که مقدار پارامترهای optimizer و loss را تغییر دهد و نتایج را بررسی می‌کنیم. از آنجا که تعداد نمودارهای این بخش خیلی زیاد است از آوردن آن‌ها در متن گزارش خودداری می‌کنیم، این گزارش‌ها در پوشه‌ی آپلود شده در پوشه‌ای به نام Generated files در فایل‌هایی با نامی‌هایی به فرمت زیر موجود اند:

Naming Format:

Q2\_basic\_<MODEL\_NAME>\_opt\_<OPTIMIZER\_NAME>\_loss\_<LOSS\_NAME>\_<TYPE\_OF\_PLOT\_BEING\_DRAWN>

پس از رسم این نمودارها برای یک مدل خاص (LSTM) حالت‌های مختلف را با یکدیگر مقایسه می‌کنیم. برای تمامی حالات شکل کلی تغییر خطا کاهشی است اما در برخی حالات شیب این کاهش یا میزان آن خیلی زیاد نیست، البته این مسئله الزاما به معنی بدتر بودن مدل نیست چون در برخی موارد نقطه‌ی شروع نیز با میزان خطای کمتری است و در نتیجه با اینکه میزان کاهش در طول یادگیری خیلی زیاد نبوده است در نهایت بهتر از مدل‌های دیگر عمل کرده‌است.

\*\* اینو نمی‌دونم دقیقا چطوری نتیجه گیری کنم \*\*

سوال ۴

برای این سوال

سوال ۵

همانطور که در شکل‌های ۱.۵، ۱.۶ و ۱.۷ دیدیم در هر یک از مدل‌ها عبارت شرطی if‌ ای قرار دارد که در صورتی که پارامتر داشتن dropout را True کنیم این لایه را اضافه می‌کند. در حالت عادی این مقدار False است اما در این بخش آن را اضافه می‌کنیم تا تاثیر را بررسی کنیم. شبکه‌ی LSTM را در این سوال بررسی می‌کنیم. در این مدل زمانی که dropout داریم از خطای اولیه‌ی بیشتری شروع می‌کنیم ولی در نهایت پس از ۲۰ ایپاک تقریبا به همان جایی می‌رسیم که مدل اولیه بدون داشتن dropout به آن می‌رسد و بهبود چشم‌گیری را مشاهده نمی‌کنیم.

علت این مسئله می‌توان میزان dropout انتخاب شده و کوچک بودن آن نیز باشد.

سوال ۶

برای این سوال

سوال ۷

برای انتخاب مهم‌ترین ویژگی‌ها باید از روش‌های feature selection انتخاب کنیم. برخی از این روش‌ها عبارتند از:

* یک روش می‌تواند محاسبه‌ی میزان Correlation بین دو به دوی ویژگی‌ها و حذف یکی از متغیر موجود در جفت‌های با correlation بالا و تکرار این کار تا رسیدن به تعداد ویژگی مورد نظر است. از آنجا که اگر correlation بالا باشد به این معنی است که دو متغیر تغییرات مشابهی دارند و روند تغییرات آن‌ها شبیه است، این مسئله باعث می‌شود که بررسی هم زمان دو ویژگی اطلاعات خیلی بیشتری از بررسی تنها یکی از آن‌ها به ما ندهد در نتیجه حذف یکی از آن‌ها باعث از دست رفتن اطلاعات زیادی نمی‌شود.
* روش دیگر می‌تواند استفاده از تست‌هایی مثل Chi-squared باشد. در این تست ایده این است که اگر ویژگی‌ای از مقدار مورد نظر (چیزی که می‌خواهیم پیش‌بینی کنیم) مستقل باشد اطلاعات زیادی به ما نخواهد داد، در نتیجه مشابه حالت قبلی با این کار می‌توانیم یکی یکی مستقل ترین‌ها را حذف کنیم تا به تعداد مورد نظر برسیم.

سوال ۸

برای این سوال

# سوال ۲ – نقصان دادگان

# سوال 4 – عنوان سوال