یادگیری ماشین در فیزیک
امتحان پایانی: پیشبینی آلودگی هوای تهران

علی ستاره کوکب
شماره دانشجویی: ۹۵۱۰۰۴۹۱

۳ تیر ۱۴۰۰

فهرست مطالب

۱ خلاصه

در این تمرین می خواهیم آلودگی هوای تهران را با استفاده از الگوریتم های یادگیری ماشین پیش بینی کنیم. داده ای که با آن کار می کنیم از دو بخش تشکیل شده است. بخش اول شامل داده های هواشناسی ساعتی مانند دما ی هوا، وزش باد، میزان بارندگی و غیره می باشد. بخش دوم داده شامل داده های مربوط به میزان آلاینده های هوا مانند CO، NOx و جز آن می باشد. بخش های این تمرین بدین صورت طبقه بندی شده اند: در بخش ۳ به نحوه گردآوری داده ها و توصیف داده ها می پردازیم. در بخش ۴ اندکی به ساختار داده های موجود می پردازیم و ارتباط میان فیچر های مختلف را بررسی می کنیم. در بخش ۵ به روش کلاسیک یادگیری ماشین مانند random forrest می پردازیم و در بخش آخر ۶ به روش شبکه عصبی برای حل این مساله می پردازیم.

۲ بیان مساله

مساله ی ما در این تمرین پیش بینی آلودگی هوا ی تهران می باشد. ابتدا باید منظورمان از آلودگی را بصورت دقیق مشخص کنیم. معیار های مختلفی برای سنجش آلودگی هوا وجود دارد. ما در اینجا از غلظت ذرات pm_{10} و ذرات $pm_{2.5}$ استفاده می کنیم. این انتخاب به دو دلیل انجام شده است. علت اول آن است که این دو آلاینده بخش اصلی آلودگی هوا ی بیشتر کلانشهر ها از جمله تهران می باشند. دلیل دوم نیز در دسترس بودن بیشتر داده های این دو آلاینده نسبت به آلاینده های دیگر در ایستگاه های سنجش آلودگی هوای تهران می باشد که در بخش ۳ بیشتر بدان می پردازیم. فیچر هایی که برای حل این مساله از آنها نیز استفاده می کنیم، داده های هواشناسی ساعتی مربوط به تهران می باشد. بنابراین مساله ما پیشبینی غلظت آلاینده های $pm_{2.5}$ و pm_{10} در زمان $t+1$ با داشتن داده های هواشناسی و غلظت این دو آلاینده در زمان t می باشد. t در مساله ما برابر ساعت است.

۳ داده

گام اساسی در حل این مساله یافتن داده های مورد نیاز می باشد. همانطور که گفتیم داده های مورد نیاز ما به دو بخش تقسیم می شوند: داده های هواشناسی و داده های آلاینده های هوا. داده های هواشناسی تهران را با استفاده از api سایت www.worldweatheronline.com[۱] بدست می آوریم. این موسسه از معتبر ترین موسساتی است که به انتشار داده های هواشناسی سراسر دنیا می پردازد. برای استفاده ی کامل از خدمات این api باید هزینه پرداخت کرد اما خدمات محدود آن برای ۶۰ روز بصورت رایگان قابل استفاده می باشد. برای گرفتن اطلاعات از این api ابتدا در این سابت ثبت نام می کنیم. پس از ثبت نام یک کلید منحصر به فرد برای استفاده از api برای ما تولید می شود که با استفاده از این کلید می توان ۵۰۰ درخواست در روز به این api ارسال کرد. با استفاده از این کلید و کتاب خانه request در پایتون، اطلاعات ساعتی هواشناسی شهر تهران (فرودگاه مهرآباد) را از تاریخ 1/1/2010 تا 5/31/2021 دانلود می کنیم. اطلاعات

هواشناسی ذخیره شده برای هر ساعت شامل دمای هوا، سرعت باد، جهت باد، رطوبت، فشار، پوشش ابری، میزان بارندگی، شاخص UV و نقطه شبنم [۱] می باشد. این api داده ها را با فرمت JSON ذخیره می کند که پس از پردازش آنها را بصورت فایل CSV در می آوریم. همچنین با استفاده از ابزار datetimeindex در کتابخانه پانداز، اندیس تمام دیتاست ها را بصورت تاریخ همراه با ساعت آن در می آوریم. خوبی استفاده از این ابزار آن است که به سادگی می توان داده های مربوط به سال، ماه یا ساعت های مختلف را فراخوانی کرد.
داده های مربوط به غلظت آلاینده های هوا ی تهران را نیز از سایت شرکت کنترل کیفیت هوا ی تهران [۲] بدست می آوریم. در این سایت اطلاعات ساعتی تمام ایستگاه های تهران در قالب فایل اکسل قابل دانلود است. ابتدا فایل داده های ساعتی تمام ۲۳ ایستگاه تهران را از تاریخ 1/1/2010 تا 5/31/2021 دانلود می کنیم و آنها را ذخیره سازی می کنیم. دقت کنید که برخی ایستگاه ها جدید تر هستند و اطلاعات تا سال 2010 در آنها موجود نیست. پس از دانلود این ۲۳ فایل نوبت جست و جو در این فایل ها و یافتن مناسب ترین ایستگاه برای استفاده از داده های آن است. منظور از داده مناسب، داده ای است که کمترین اطلاعات از دست رفته را داشته باشد. در هر یک از این فیل ها اطلاعات ساعتی مربوط به ۸ آلاینده وجود دارد. نکته ی بسیار مهمی که در این داده ها وجود دارد و باید بدان توجه کنیم آن است که برخی از داده های از دست رفته در این دیتاست ها حتی بصورت nan نیز موجود نمی باشند؛ بلکه سطر مربوط به آن داده به کلی ناموجود است. برای مثال ممکن است سطر یک مربوط به داده ی ساعت ۱ روز ۲۳ فروردین باشد و سطر بعدی مربوط به روز ۱ تیر باشد و در این بین تعداد روز و ساعت به کلی حتی بصورت nan ناموجود باشند. بنابراین اولین گام آن است که تمام ساعت های ناموجود در این ۲۳ فایل را بیابیم و مقدار آنها را بصورت nan در فایل داده اضافه کنیم. برای این کار ابتدا تاریخ ها را از شمسی به میلادی تبدیل می کنیم . این کار را با استفاده از کتابخانه jdatetime انجام می دهیم. سپس مشابه قبل، اندیس ها را با استفاده از ابزار datetimeindex بصورت تاریخ همراه با ساعت در می آوریم. سپس با استفاده از دستور date range در پانداز، زمان های ناموجود در هر یک از فایل ها با مقدار nan اضافه می کنیم. اکنون تمام فایل ها کامل می باشند. حال برای هر یک از این داده ها و هر یک از این ۸ آلاینده، درصد داده های موجود به نسبت کل داده ها را محاسبه می کنیم که در شکل ۱ می بینید. همانطور که در شکل ۱ دیده می شود، داده های مربوط به دو آلاینده CO و SO_2 ایستگاه منقطه ۲۲، کمترین میزان از دست رفتگی داده را دارا می باشند. پس از آن، داده های مربوط به دو آلاینده pm_{10} و $pm_{2.5}$ به ترتیب برای ایستگاه شریف و ایستگاه صدر در منطقه ۳ می باشند. به علت آنکه این دو آلاینده نقش بیشتری در آلودگی هوای تهران دارند، ما این دو آلاینده و اطلاعات این دو ایستگاه را به عنوان متغیر هدفمان برای پیش بینی انتخاب می کنیم.
همانطور که در شکل ۱ دیده می شود، داده های مربوط به دو آلاینده $pm_{2.5}$ و pm_{10} مربوط به دو ایستگاه گفته شده، به ترتیب دارای ۱۴ و ۱۰ درصد از دست رفتگی داده می باشند که ممکن است در کار پیش بینی ما اثرات منفی داشه باشند. با جست و جو بیشتر در داده های این دو ایستگاه، متوجه می شویم که اگر بازه زمانی که آن را بررسی می کنیم را محدود تر کنیم و تنها

[۱] dew point

	O3 ppb	CO ppm	NO ppb	NO2 ppb	NOx ppb	SO2 ppb	PM 10 ug/m3	PM 2.5 ug/m3
c-14_اتوبان_محلاتی_منطقه.csv	0.461041	0.482079	0.577614	0.577865	0.577865	0.525345	0.744385	0.0
c-1_اقدسیه_منطقه.csv	0.653377	0.843455	0.710362	0.712108	0.712045	0.468549	0.678806	0.770483
c-_بهاران.csv	0.022909	0.088125	0.09329	0.093396	0.093258	0.092713	0.084369	0.0
c-6_تربیت_مدرس_منطقه.csv	0.694069	0.764969	0.771031	0.774237	0.774237	0.710486	0.84283	0.819116
c-3_دروس_منطقه.csv	0.166468	0.296738	0.198798	0.200972	0.200669	0.144835	0.061449	0.171727
c-7_ستاد_بحران_منطقه.csv	0.675105	0.829214	0.783483	0.786351	0.786341	0.664903	0.712491	0.756668
c-18_شاد آباد_منطقه.csv	0.604638	0.555448	0.796072	0.801537	0.801537	0.60304	0.875751	0.785529
c-2_شریف_منطقه.csv	0.684578	0.620207	0.732207	0.732358	0.732358	0.612889	0.898876	0.817409
c-2_شهرداری_منطقه.csv	0.61737	0.817101	0.487643	0.487643	0.487643	0.431605	0.749569	0.720116
c-4_شهرداری_منطقه.csv	0.477112	0.604078	0.541677	0.54519	0.545107	0.406504	0.265809	0.506932
c-10_شهرداری_منطقه.csv	0.355123	0.284243	0.21513	0.215391	0.214753	0.502854	0.201746	0.46813
c-11_شهرداری_منطقه.csv	0.345546	0.640056	0.524707	0.524728	0.524728	0.513666	0.249362	0.476014
c-16_شهرداری_منطقه.csv	0.558919	0.640747	0.625711	0.629151	0.625951	0.573766	0.642231	0.167127
c-19_شهرداری_منطقه.csv	0.463666	0.479883	0.339293	0.347407	0.347166	0.468423	0.582026	0.123913
c-21_شهرداری_منطقه.csv	0.631823	0.448625	0.658382	0.658395	0.658395	0.681954	0.74234	0.724505
c-22_شهرداری_منطقه.csv	0.69335	0.951667	0.65524	0.65524	0.65524	0.932269	0.786439	0.861684
c-3_صدر_منطقه.csv	0.719905	0.773974	0.813478	0.813478	0.813478	0.67377	0.0	0.833787
c-15_مسعودیه_منطقه.csv	0.62889	0.813864	0.649352	0.649362	0.649362	0.618716	0.586366	0.562202
c-9_میدان_فتح_منطقه.csv	0.600176	0.632008	0.709862	0.712477	0.710333	0.48365	0.855491	0.000094
c-22_پارک_رز_منطقه.csv	0.26973	0.485623	0.338404	0.349069	0.349069	0.396424	0.534253	0.381943
c-_پونک.csv	0.677907	0.814262	0.786721	0.787003	0.786784	0.63762	0.807696	0.566343
c-13_پیروزی_منطقه.csv	0.607497	0.606357	0.684163	0.690831	0.689114	0.658165	0.775892	0.677249
c-8_گلبرگ_منطقه.csv	0.575355	0.816133	0.815171	0.815182	0.815161	0.591018	0.718078	0.68669

شکل ۱: مقایسه میزان داده های موجود به نسبت کل داده ها در اطلاعات ۲۳ ایستگاه سنجش آلاینده های هوا ی تهران. با استفاده از این جدول می توان بهترین ایستگاهی را که داده های کامل تری دارد انتخاب کرد.

داده های مربوط به سال های ۲۰۱۷ تا ۲۰۲۱ را نگه داریم، این از دست رفتگی برای pm_{10} به ۳ درصد و برای $pm_{2.5}$ به ۵ درصد می رسند که بسیار خوب می باشند. بنابراین ما در الگوریتم یادگیری مان از داده های مربوط به این سال ها استفاده می کنیم. برای پر کردن این چند درصد باقی مانده نیز از KNNImputer در کتابخانه sklearn استفاده می کنیم. بدین ترتیب کار ساختن داده های مورد نیازمان به پایان می رسد.

۴ گشت و گذار در داده ها

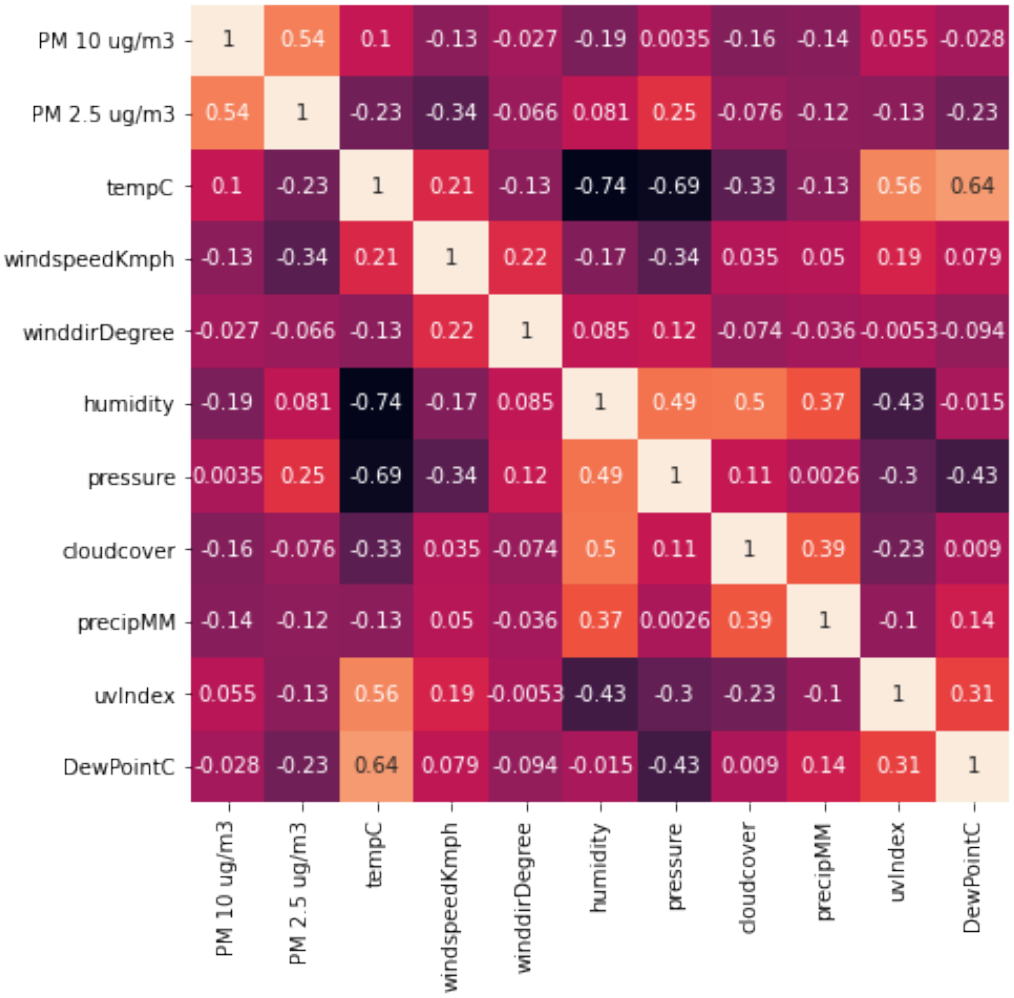
اکنون که داده های مان کامل شده است، خوب است اندکی در آن به گشت و گذار بپردازیم. در شکل ۲ هم بستگی میان ستون های مختلف داده را می بینید. همانطور که در شکل هم پیداست به نظر نمی رسد هم بستگی قابل توجهی میان ویژگی های مختلف داده وجود داشته باشد. در شکل ۳ و ۴ نیز به ترتیب تغییرات دو آلاینده pm_{10} و $pm_{2.5}$ را می بینید. همانطور که دیده می شود در بعضی زمان ها غلظت این دو آلاینده بصورت چشم گیری از افزایش یافته است. در شکل ۵ نیز نمودار دمای هوا ی تهران را در از سال ۲۰۱۰ تا ۲۰۲۱ مشاهده می کنید. همانطور که مشاهده می کنید زمستان های تهران در طی این ده سال تقریبا ۱۰ درجه سانتی گراد گرم شده است. در شکل ۶ نیز نمودار تعداد از ستون های داده را برحسب یکدیگر می بینید. در دو بخش بعدی به مساله پیش بینی غلظت این دو آلاینده می پردازیم. در تمامی روش های گفته داده ها را با استفاده از min-max scaler مقیاس کرده ایم.

۵ مدل های سنتی

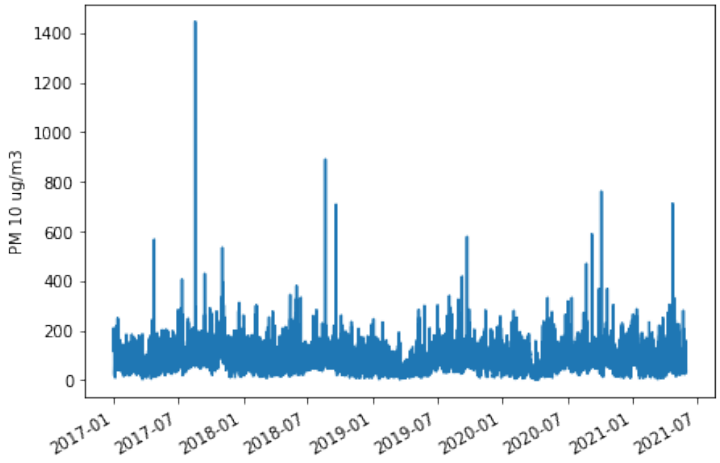
۱.۵ Random Forest

در این قسمت به نتایج بدست آمده برای پیش بینی غلظت دو آلاینده گفته شده می پردازیم. از این مدل ها به عنوان مدل های پایه ای[۲] استفاده می کنیم. در این مدل ها، از ۹ ویژگی گفته شده آب و هوایی به عنوان فیچر استفاده می کنیم. اولین مدل رگرسور، random forest می باشد. این رگرسور شامل ۲۰۰ درخت به عمق ۳۰ می باشد. از ۲۰ درصد پایانی داده ها به عنوان داده تست و از بقیه برای آموزش استفاده کرده ایم. نتایج بدست آمده بر روی داده های pm_{10} می بینید در شکل ۷ می بینید. همانطور که می بینید این مدل نتوانسته به خوبی داده های تست را پیشبینی کند. همچنین میزان همبستگی میان داده های تست و پیشبینی ۳۳ درصد می باشد که عدد بسیار پائینی است. همچنین مقدار rmse بر روی داده تست برابر ۴۱ می باشد.

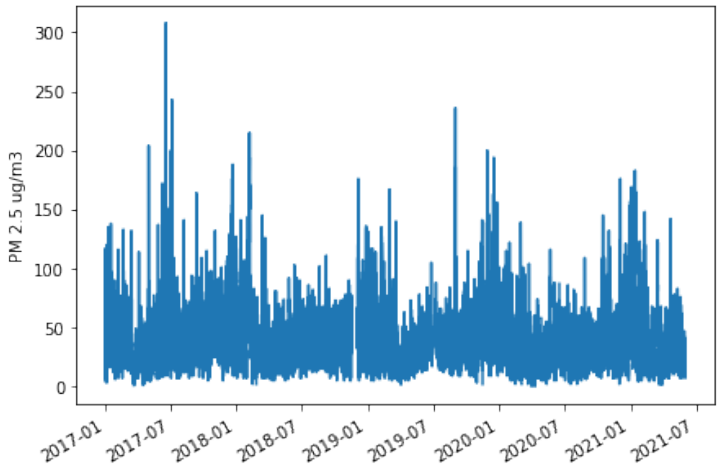
[۲] baseline



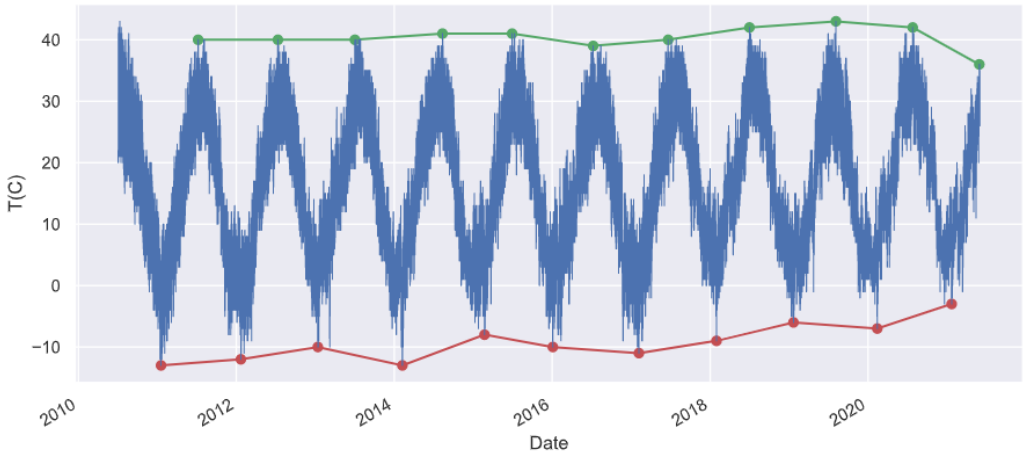
شکل ۲: ماتریس همبستگی میان ویژگی های مختلف داده. همانطور که دیده می شود، همبستگی قابل توجهی میان داده های هواشناسی و دو آلاینده مورد بررسی دیده نمی شود.



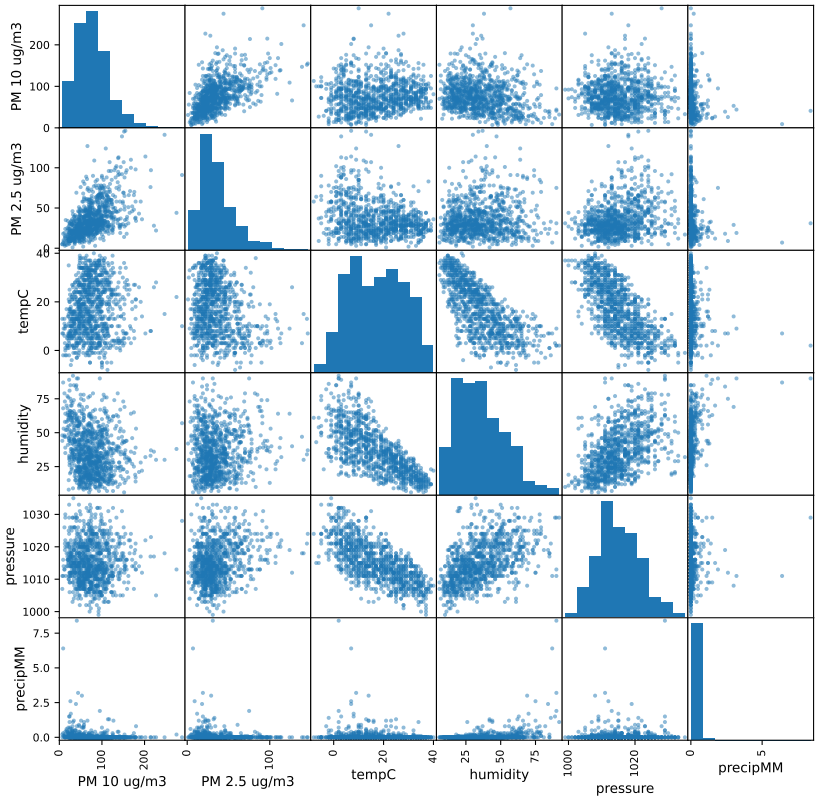
شکل ۳: نمودار غلظت pm_{10} برحسب زمان.



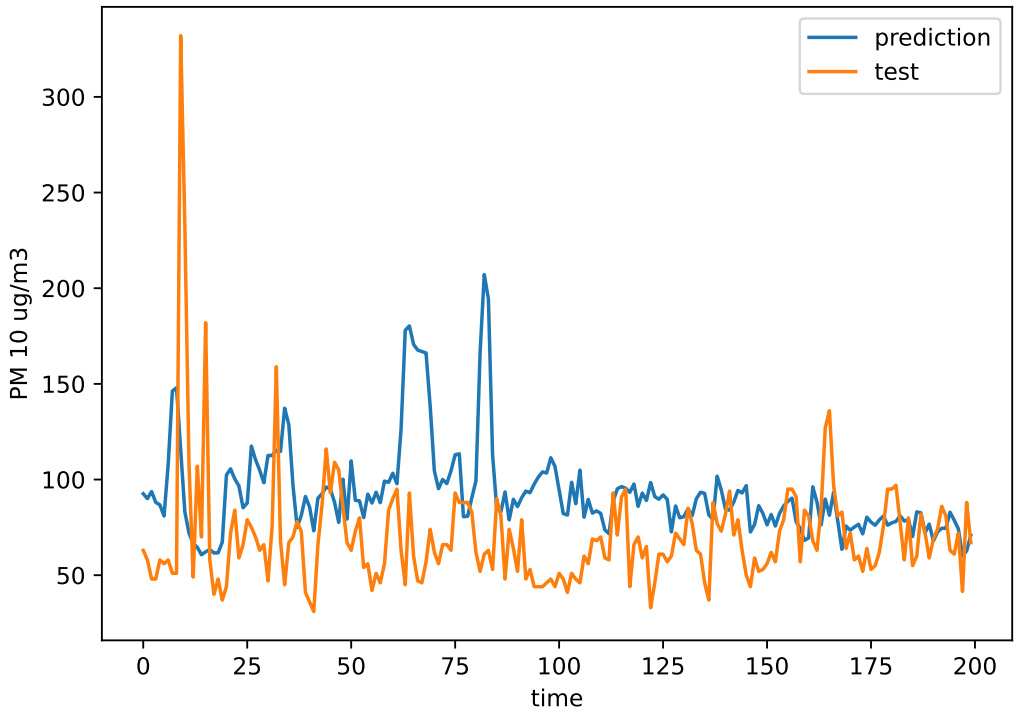
شکل ۴: نمودار غلظت $pm_{2.5}$ برحسب زمان.



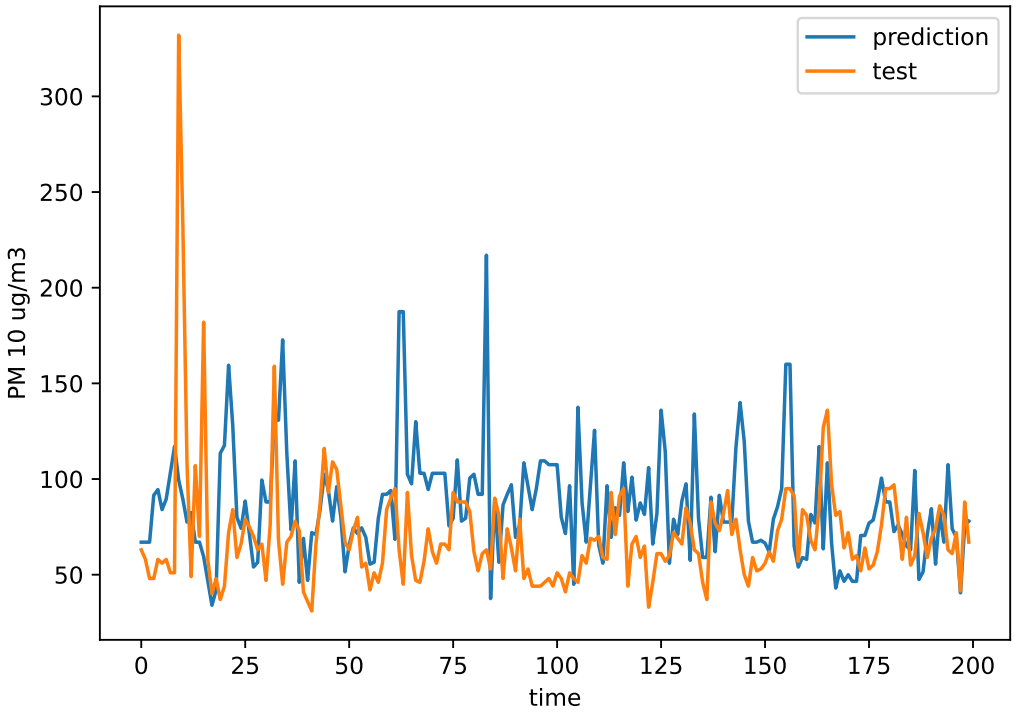
شکل ۵: نمودار دمای هوای تهران از سال ۲۰۱۰ تا ۲۰۲۱. همانطور که دیده می شود، زمستان های تهران بصورت چشمگیری گرم تر شده اند.



شکل ۶: نمودار داده های موجود برحسب یکدیگر.



شکل ۷: مقایسه بین پیش بینی مدل Random Forest و داده تست. همانطور که می بینید، این مدل به خوبی نتوانسته داده تست را پیشبینی کند.



شکل ۸: مقایسه بین پیش بینی مدل knn و داده تست. همانطور که می بینید، این مدل به خوبی نتوانسته داده تست را پیشبینی کند.

۲.۵ KNN

در این قسمت از رگرسور KNN با دو همسایه برای پیشبینی داده های تست استفاده کرده ایم. نتایج این پیشبینی را در شکل ۸ می بینید. همانطور که می بینید، این مدل نیز نتوانسته به خوبی پیش بینی را انجام دهد و میزان همبستگی میان پیشبینی ها و داده ی تست برای این مدل نیز برابر ۲۲ درصد می باشد. همچنین rmse برای این مدل برابر ۴۷ می باشد.

۶ شبکه عصبی

در این قسمت با استفاده از شبکه عصبی، سعی می کنیم غلظت دو آلاینده گفته شده را پیش بینی کنیم. همانطور که می دانیم شبکه های عصبی بازگشتی[۳] برای پیش بینی سری های زمانی ساخته شده اند. بنابراین ما نیز در این جا به سراغ استفاده از این نوع شبکه برای مساله خودمان رفته ایم. پس از آزمون و خطا های فراوان، ساختار شبکه ای که به آن رسیده ایم بصورت زیر می باشد:

[۳] recurrent neural network

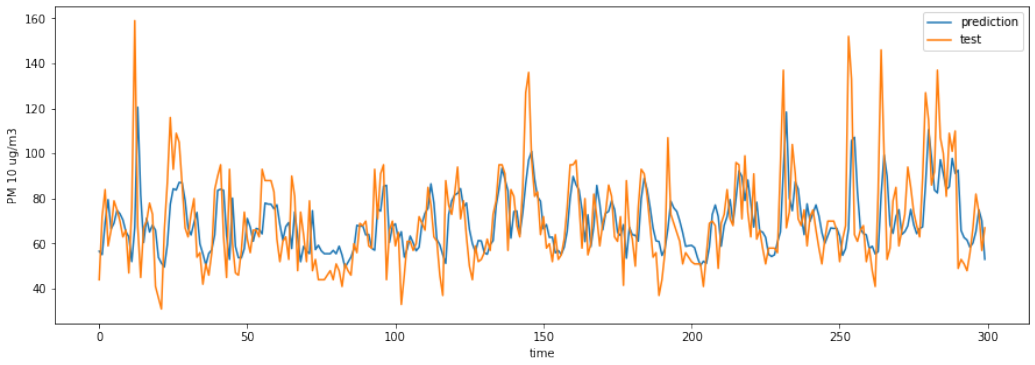
۱. لایه اول یک لایه convolutional با ۶۴ فیلتر و پنجره ای به اندازه ۳ می باشد. تابع فعالسازی این لایه نیز tanh است.

۲. لایه دوم و سوم یک لایه LSTM با ۳۲ راس و تابع فعالسازی relu می باشند.

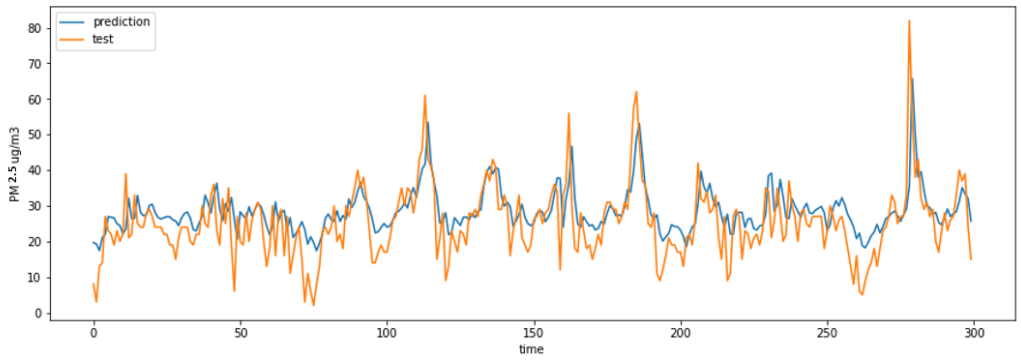
۳. لایه چهارم یک لایه Dense با ۳۲ راس و با تابع فعالسازی relu می باشد.

۴. لایه پنجم نیز لایه نهایی با یک راس و تابع فعالسازی linear می باشد.

برای جلوگیری از over fitting، میان لایه های ۲، ۳ و ۴ از dropout با نرخ 0.2 استفاده کرده ایم. علت استفاده از لایه convolutional در کنار لایه LSTM آن است که اگر به تنهایی از لایه LSTM استفاده کنیم، مشاهده می کنیم که مقداری عقب افتادگی میان داده های اصلی و پیش بینی هایمان داریم. به عبارتی بنظر می رسد که لایه LSTM پیش از حد نقاط جدید را مانند نقاط قبلی پیش بینی می کند و به همین خاطر همواره مقداری در پیش بینی هایمان به نسبت داده اصلی تاخیر داریم. از طرفی لایه convolutional در شناسایی الگو های موضعی خوب عمل می کند. بنابراین با استفاده از این دو لایه در کنار هم، هم الگوهای بلند مدت را با استفاده از لایه LSTM خواهیم داشت، و هم الگو های موضعی را با استفاده از لایه convolutional شناسایی می کنیم. البته تاخیر میان پیشبینی ها و داده اصلی در انیجا نیز کاملا از میان نمی رود اما کمتر از حالت استفاده تنها از لایه LSTM می باشد.
همانطور که می دانیم در شبکه های عصبی بازگشتی هر داده علاوه بر فیچر های معمول خود (در اینجا اطلاعات هواشناسی)، از داده های قبل تر از خود نیز می تواند به عنوان فیچر استفاده کند. انتخاب اینکه برای پیشبنی داده زمان $t+1$ به چند گام زمانی نگاه کنیم بستگی به مساله و یکی از هایپر پارامتر های مساله می باشد. ما در این جا برای هر داده، از ۲۰ داده قبلی آن، معادل ۲۰ ساعت، استفاده می کنیم. برای اینکه شکل داده هایمان را مناسب ورود به لایه LSTM بکنیم، از ابزار timeseriesgenerator در Keras استفاده می کنیم. این ابزار مقدار گام زمانی که به عقب نگاه می کنیم را می گیرد، و خودش داده ها را به شکل مناسب در می آورد. در این جا نیز از ۲۰ درصد داده برای تست و از بقیه برای آموزش استفاده کرده ایم. همچنین می توان در این تابع داده ها را بصورت batch نیز در آورد که ما در اینجا از اندازه batch ۳۲ استفاده کرده ایم. تابع هزینه ای که در این مساله از آن استفاده کرده ایم، تابع mse می باشد. همچنین نرخ یادگیری را نیز بر روی 0.001 قرار داده ایم. مدت زمان یادگیری این مدل برای آلاینده pm_{10} برابر ۲ دقیقه و ۳۴ ثانیه و برای $pm_{2.5}$ برابر ۱ دقیقه و ۳۴ ثانیه می باشد. لازم به ذکر است که ما از خاصیت callbacks برای متوقف کردن فرآیند آموزش در صورت عدم تغییر هزینه داده validation پس از ۵ epoch استفاده کرده ایم. این کار باعث صرفه جویی در زمان آموزش می شود. در شکل ۹ و ۱۰، نتایج پیشبینی مدل و داده های واقعی را می بنیدو همانطور که دیده می شود، مدل تقریبا توانسته روند های موجود در سری زمانی را تشخیص دهد. میزان همبستگی میان پیشبینی ها و داده ی واقعی برای pm_{10} برابر ۸۱ درصد و برای $pm_{2.5}$ برابر ۸۲ درصد می



شکل ۹: مقایسه سری زمانی پیشبینی شده توسط مدل و داده واقعی برای pm_{10}. همانطور که می بنید، مدل توانسته به خوبی الگو ها را تشخیص دهد.



شکل ۱۰: مقایسه سری زمانی پیشبینی شده توسط مدل و داده واقعی برای $pm_{2.5}$. همانطور که می بنید، مدل توانسته به خوبی الگو ها را تشخیص دهد.

باشد. همچنین rmse برای این دو داده به ترتیب ۲.۲۵ و ۶.۲۳ می باشد. همانطور که دیده می شود، این مدل بسیار عملکرد بهتری نسبت به دو مدل پایه ای در قسمت ۵ از خود نشان می دهند.

مراجع

[1] World weather online. https://www.worldweatheronline.com/.

[2] https://airnow.tehran.ir/.