پروژه‌ی پایانی درس شناسایی الگو

عنوان: پیش‌بینی زلزله به کمک علم نجوم و هوش مصنوعی

استاد: آقای دکتر آبین

استادیار: آقای مهندس سیمچی

ارائه‌دهنده: پدرام یزدی پور

شماره دانشجویی: ۹۹۴۴۳۲۴۱

بهمن ۱۴۰۰

مقدمه

پیش‌بینی قطعی زلزله تاکنون غیرممکن بوده است. با پیشرفت روزافزون تکنولوژی و به ویژه هوش مصنوعی می‌توان راه حل‌های جدید را آزمود. در این پروژه لیست تمام زلزله‌های ایران از سال ۱۹۰۰ میلادی تا ۲۰۲۰ با جزییات جمع‌آوری شده است اعم از تاریخ و ساعت دقیق، مختصات جغرافیایی، عمق کانون زلزله در زمین و بزرگی آن. قصد داریم با کمک کتابخانه‌ی Solar System در زبان پایتون، ویژگی‌های نجومی هر تاریخ را استخراج کرده و سپس با استفاده از یک مدل مبتنی بر SVM، امکان پیش‌بینی زلزله‌های بالای ۴.۵ ریشتر را ارزیابی نماییم. در نهایت، دقت مدل را با معیارهای متفاوت اعلام خواهیم کرد. مطمئنا پیش‌بینی زلزله می‌تواند جان هزاران انسان را نجات داده و از خسارتهای میلیاردی جلوگیری کند.

هدف پروژه

هدف ما پیش‌بینی زلزله‌های بالای ۴.۵ ریشتر در ایران به کمک ویژگی‌های جمع‌آوری شده حاصل از موقعیت نجومی سیارات منظومه‌ی شمسی است. علت انتخاب این ویژگی‌ها می‌تواند به قانون جهانی گرانش نیوتن مربوط باشد که بنابر آن، هر دو جسمی به یکدیگر نیروی جاذبه وارد می‌کنند که مقدار این نیرو با فاصله‌ی آنها رابطه‌ی معکوس دارد.

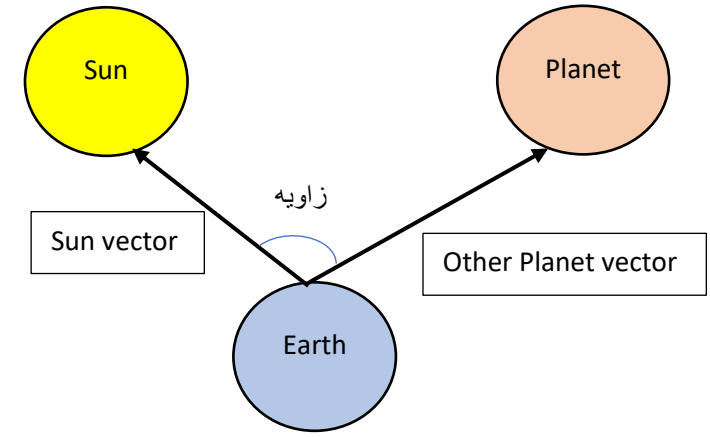
پیش‌پردازش داده‌ها

داده‌ها در دو برگ مختلف یک فایل اکسل جمع شده‌اند. ابتدا هر داده‌ای که قسمتی از آن حاوی مقادیر Null باشد را حذف می‌کنیم. برای جمع‌آوری ویژگی‌ها، نیاز به سال، ماه، روز، ساعت و دقیقه‌ی وقوع زلزله داریم پس هر دو برگ را طوری تغییر می‌دهیم که علاوه بر این موارد شامل مختصات جغرافیایی محل وقوع زلزله و همچنین بزرگی آن باشد. وقتی داده‌ها در یک دیتافریم منظم شدند، هر سطر از داده‌ها را به عنوان ورودی به کتابخانه‌ی نجومی مذکور داده و خروجی شامل مختصات سه‌بعدی تمام سیارات منظومه‌ی شمسی را دریافت می‌کنیم؛ اینها بخشی از ویژگی‌های نهایی مورد نظر ما برای پیش‌بینی زلزله هستند.

محاسبه‌ی ویژگی‌ها

همانطور که قبلا توضیح داده شد، پس از افزودن مختصات سه بعدی سیارات، یک ویژگی دیگر را هم برای بهبود احتمالی عملکرد مدل خواهیم افزود. قصد داریم زاویه‌ی بین بردارهای زمین-خورشید و زمین-سیاره‌ی ثالث را به ازای تمام سیارات منظومه‌ی شمسی محاسبه کرده و به جدول خود بیافزاییم.

شکل ۱ نحوه‌ی محاسبه‌ی زاویه بین زمین-خورشید و زمین-سیاره‌ی ثالث را نشان می‌دهد.



شکل ۱: محاسبه‌ی زاویه بین یک سیاره با خورشید و زمین

از ریاضیات رابطه‌ی ضرب داخلی دو بردار را به یاد داریم:

$$\theta = \cos^{-1}\left(\frac{\mathbf{u}\cdot\mathbf{v}}{|\mathbf{u}||\mathbf{v}|}\right)$$

مختصات زمین و خورشید و سیاره‌ی دیگر را داریم پس دو بردار را هم داریم و می‌توانیم ضرب داخلی هم انجام دهیم و در نهایت زاویه را محاسبه می‌کنیم.

مدل به کار رفته

از یک SVM با کرنل گاوسی استفاده می‌کنیم. کرنل گاوسی برای مواقعی که پیش‌دانسته‌ای درباره‌ی دادگان نداریم گزینه‌ی مناسبی است. دو پارامتر برای تنظیم داریم که باید به صورت دستی انتخاب کنیم. پارامتر C که مربوط به پنالتی اشتباه دسته‌بندی کردن داده‌ها در فرآیند آموزش است و پارامتر گاما که مربوط به واریانس هر کرنل گاوسی است. با کمی سعی و خطا مقادیر ۱۰۰۰ و ۱۰ به ترتیب برای C و گاما(راست به چپ) انتخاب می‌شوند. اگر مقادیری جز این انتخاب شوند دقت مدل افت محسوسی خواهد داشت. مقادیر بسیار بزرگ برای C موجب تمرکز مدل برای درست دسته‌بندی کردن تمام نمونه‌ها و در نتیجه بیش‌برازش خواهد شد و مقادیر خیلی کوچک آن نیز باعث تضعیف مدل و خطر کم‌برازش مدل می‌شود. البته با کرنل خطی نیز که شباهت را

بر اساس ضرب داخلی می‌سنجد، مدل را ارزیابی خواهیم نمود. علت انتخاب مدل SVM این است که این مدل همواره پاسخ بهینه‌ی سراسری را می‌یابد و ضمنا در در این درس برای نخستین بار با آن آشنا شده‌ایم.

خلاصه‌ی توضیح کد

محاسبه‌ی زاویه‌ی سیاره با زمین و خورشید

```
def  angle_calculator(planet_pos,sun_pos)
```

افزودن ویژگی‌های مختصات سه‌بعدی کرات و زاویه به دیتافریم اصلی

```
def long_lat_teta_calculator(df, planet_names)
```

افزودن ویژگی‌های مربوط به ماه (قمر سیاره‌ی زمین)

```
def moon_features(df)
```

اجرای الگوریتم ماشین بردار پشتیبان

```
clf = svm.SVC(kernel='rbf',C = ۱۰۰۰, gamma = ۱۰)
clf.fit(X_train, y_train)
predicted = clf.predict(X_test)
```

پیکربندی سیستم مورد استفاده

کد این پروژه در آزمایشگاه آنلاین گوگل اجرا شده است و هم لینک هم کل کد منبع در فایل پروژه آپلود شده است.

نتایج

در حالت اول بلافاصله پس از نرمالسازی داده‌ها اقدام به آموزش مدل می‌کنیم:

جدول ۱: دقتها بدون استفاده از کاهش ابعاد

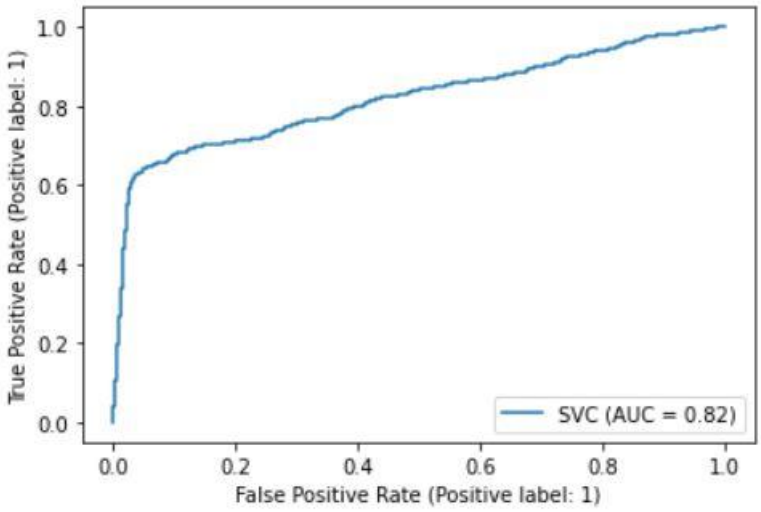
	Accuracy	Precision	Recall	F۱
RBF	۹۵.۶	۸۱.۴	۷۴.۰	۷۷.۱
Linear	۹۵.۰	۸۰.۰	۶۸.۱	۷۲.۳

در حالت دوم ابتدا الگوریتم PCA را برای کاهش ابعاد به کار می‌بریم و با استفاده از ۱۸ مولفه‌ی اصلی اول که حدود ۸۴ واریانس را حفظ می‌کنند(برای کاهش زمان اجرا و همچنین کاهش فضای جستجو) مدل را ارزیابی می‌کنیم:

جدول ۲: دقتها با استفاده از کاهش ابعاد

	Accuracy	Precision	Recall	F۱
RBF	۹۵.۰	۸۱.۰	۷۵.۰	۷۷.۰
Linear	۹۵.۰	۸۰.۰	۷۳	۷۶

قابل پیش‌بینی بود که مدل کرنل گاوسی بتواند به دقتهای بهتری برسد و در واقع داده‌های ما حالت خوشه‌ای دارند. در مرحله‌ی بعد، نمودار ROC را رسم می‌کنیم:

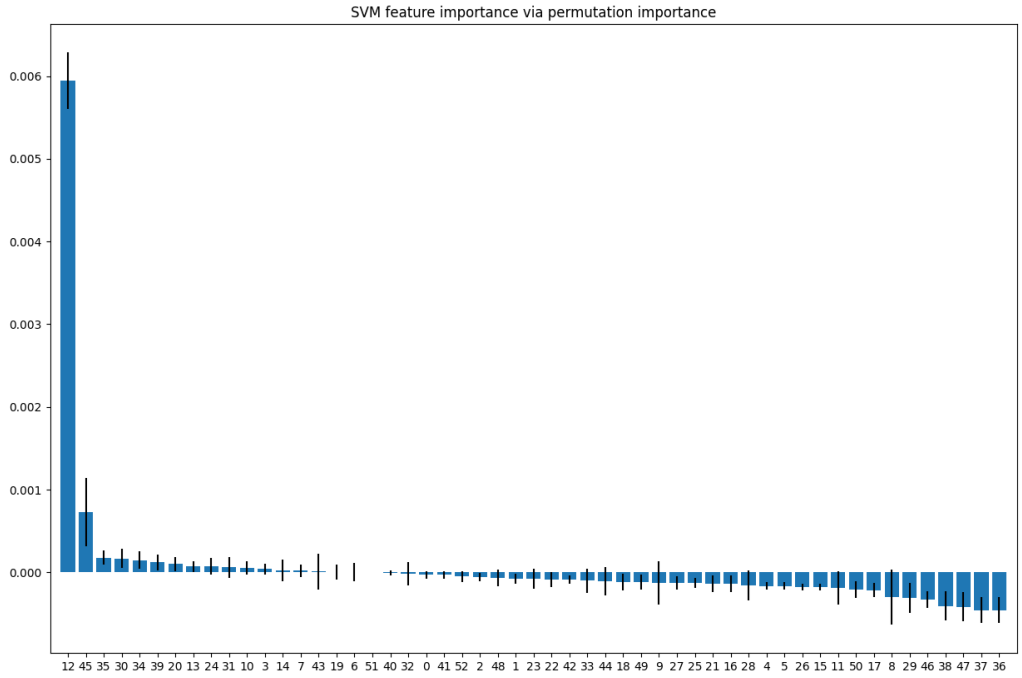


شکل ۲: نمودار ROC

در نمودار فوق، مقدار مساحت زیر نمودار ۰.۸۲ کل فضاست که می‌دانیم مقادیر بین ۰.۸ تا ۰.۹ عالی هستند. در واقع مقدار ۰.۵ متعلق به دسته‌بند تصادفی است. در نمودار فوق مشاهده می‌شود که تا نرخ تشخیص صحیح ۶۰ درصد، مقادیر نرخ تشخیص نادرست تقریبا ثابت هستند اما پس از آن، تقریبا به ازای هر مقداری که نرخ تشخیص درست بیشتر شود باید هزینه‌ی افزایش ناخواسته‌ی نرخ تشخیص نادرست را هم بپردازیم.

در مرحله‌ی بعد، وزن اهمیت هر ویژگی را به کمک الگوریتم Permutation Importance خواهیم سنجید. در این الگوریتم، به ازای هر ویژگی، چندین بار مقادیر یک ویژگی خاص، به صورت تصادفی به هم می‌ریزند و

بعد عملکرد مدل روی داده‌های جدید سنجیده می‌شود. معیار سنجش هم برای مسائل دسته‌بندی معمولا Accuracy است. هر ویژگی که باعث تغییرات بیشتری در معیار ارزیابی عملکرد مدل شود یعنی ویژگی تعیین‌کننده‌تری بوده است و اهمیت بالاتری دارد. در ادامه نمودار و جدول مربوط به این الگوریتم را که روی داده‌های تست(۲۰ درصد کل دادگان) ارزیابی شده آورده‌ایم:



شکل ۳: نمودار اهمیت هر ویژگی

راجع به نمودار فوق لازم است اشاره کنیم که ۵۳ ویژگی داریم که از صفر تا ۵۲ شماره دارند. برای اینکه بدانیم کدام ویژگی‌ها بیشترین اهمیت را دارند جدول مربوطه را هم چاپ می‌کنیم. نتایج به دست آمده طوری نیست که بتوانیم تفسیر علمی برای وقوع زلزله بیابیم زیرا نه در تخصص ماست نه جدول اهمیت ویژگی‌ها نتایج خیلی معناداری به دست داده‌اند؛ مثلا یکی از ابعاد سیاره‌ی نپتون جزء ویژگی‌های بسیار مهم تلقی شده که دلیلش مشخص نیست. یکی از دلایل می‌تواند ناکافی بودن دادگان مورد استفاده باشد.

جدول ۳: وزن اهمیت هر ویژگی

Feature	Weight
0.00594589238	tetaEris
0.000723416906	dicChiron
0.000178376771	laitNeptune
0.000168466951	dicSaturn
0.000148647309	longitNeptune
0.000118917848	dicPluto
9.90982063e-05	laitVenus
6.93687444e-05	longitSun
6.93687444e-05	dicMars
5.94589238e-05	longitUranus
4.95491032e-05	tetaCeres
3.96392825e-05	tetaVenus
1.98196413e-05	tetaUranus
1.98196413e-05	laitSun
9.90982063e-06	longitChiron
0.0	tetaSaturn
0.0	longitVenus
0.0	disM
-9.90982063e-06	longitCeres

جمع‌بندی

با توجه به نتایج توانستیم سه چهارم زلزله‌های بالای ۴.۵ ریشتر را به درستی تشخیص دهیم و همچنین حدود چهار پنجم از موارد مثبت اعلام شده کاملا درست بوده‌اند و در مجموع هم عملکرد مدل در ۹۵ درصد موارد صحیح بوده است. نتایج فوق نشان می‌دهد هنوز برای اینکه بتوانیم به نتایج این مدل اعتماد کنیم راه زیادی مانده اما می‌توان نتایج کسب شده را امیدوارکننده نیز دانست و برای مناطق زلزله‌خیز حتی مورد استفاده هم قرار داد. اگر امکان جمع‌آوری دادگان کشورهای همسایه نیز وجود داشت(به دلیل اینکه گسلهای فلات ایران محدود به محدوده‌ی جغرافیایی فعلی ایران نیستند) یا از ویژگی‌های گسترده‌تری نظیر وضعیت آب‌وهوا استفاده می‌شد، شاید دقت مدل بهبود می‌یافت. برای ارزیابی عملکرد مدل با دادگان زلزله‌های اخیر، خوشبختانه زلزله‌ی بالای ۴.۵ ریشتر رخ نداده بود و عملا هیچ داده‌ای با برچسب مثبت نیافیتم و بنابراین ارزیابی را به دادگان قبلی محدود کردیم چون این مدل در پیش‌بینی برچسب صفر عملکرد بهتری دارد(دادگان آموزشی با برچسب صفر بیشتر بوده‌اند).