



دانشگاه صنعتی امیرکبیر

(پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر و فناوری اطلاعات

پروژه کارشناسی

گرایش معماری سیستم‌های کامپیوترا

طراحی و پیاده‌سازی سامانه تحلیل رفتار راننده با استفاده از سیگنال‌های آینرسی

نگارش

نیما توسلی

استاد راهنما

دکتر احمد نیکآبادی

شهریور ۱۳۹۹

بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِيْمِ

# صفحه فرم ارزیابی و تصویب پایان نامه- فرم تأیید اعضاء کمیته دفاع

در این صفحه فرم دفاع یا تایید و تصویب پایان نامه موسوم به فرم کمیته دفاع- موجود در پرونده آموزشی- را قرار دهید.

## نکات مهم:

- نگارش پایان نامه/رساله باید به زبان فارسی و بر اساس آخرین نسخه دستورالعمل و راهنمای تدوین پایان نامه های دانشگاه صنعتی امیرکبیر باشد.(دستورالعمل و راهنمای حاضر)
- رنگ جلد پایان نامه/رساله چاپی کارشناسی، کارشناسی ارشد و دکترا باید به ترتیب مشکی، طوسی و سفید رنگ باشد.
- چاپ و صحافی پایان نامه/رساله بصورت پشت و رو(دورو) بلامانع است و انجام آن توصیه می شود.



دانشگاه صنعتی امیرکبیر  
(پلی‌تکنیک تهران)

به نام خدا

تعهدنامه اصالت اثر

تاریخ: شهریور ۱۳۹۹

اینجانب نیما توسلی متعهد می‌شوم که مطالب مندرج در این پایان‌نامه حاصل کار پژوهشی اینجانب تحت نظرارت و راهنمایی استادی دانشگاه صنعتی امیرکبیر بوده و به دستاوردهای دیگران که در این پژوهش از آنها استفاده شده است مطابق مقررات و روال متعارف ارجاع و در فهرست منابع و مأخذ ذکر گردیده است. این پایان‌نامه قبلاً برای احراز هیچ مدرک هم‌سطح یا بالاتر ارائه نگردیده است.

در صورت اثبات تخلف در هر زمان، مدرک تحصیلی صادر شده توسط دانشگاه از درجه اعتبار ساقط بوده و دانشگاه حق پیگیری قانونی خواهد داشت.

کلیه نتایج و حقوق حاصل از این پایان‌نامه متعلق به دانشگاه صنعتی امیرکبیر می‌باشد. هرگونه استفاده از نتایج علمی و عملی، واگذاری اطلاعات به دیگران یا چاپ و تکثیر، نسخه‌برداری، ترجمه و اقتباس از این پایان‌نامه بدون موافقت کتبی دانشگاه صنعتی امیرکبیر ممنوع است. نقل مطالب با ذکر مأخذ بلامانع است.

نیما توسلی

امضا

تکیه بر تقوی و دانش در طریقت کافر است  
راه را کر صد هزار دنگل باید ش

(حافظ)

# سپاس‌گزاری

به نشانهٔ حداقل سپاس، از بزرگواران و عزیزان؛  
از پدر و مادر مهربانم که همواره انسانیت را به من می‌آموزند؛  
از جناب آقای دکتر احمد نیکآبادی، به پاس تقبل راهنمایی‌های ارزنده‌شان بعنوان استاد راهنما  
که در طول مسیر پژوهش با بزرگواری و برداری رهنمون من شده‌اند؛  
از محمد قاسمپور گرامی که همواره مدیون محبت و راهنمایی‌هایش هستم؛  
و از سایر دوستان عزیزم که در کنارشان این مرحله را طی نمودم.

نیما توسلی  
۱۳۹۹ شهریور

## چکیده

نحوه انجام اعمال پایه رانندگی مانند شتاب، ترمز، و چرخش نمایانگر رفتار راننده در طول رانندگی است که بعنوان یک عامل مهم در سامانه ماشین-راننده-محیط در نظر گرفته می‌شود. عوامل متعددی مانند خستگی، حواس‌پرتی، تجربه، و شرایط محیط بر روی این رفتار تاثیرگذار هستند. رفتارهای نادرست و پرخطر علاوه بر افزایش احتمال رخ دادن حوادث و تصادفات رانندگی، الگوی مصرف سوخت و استهلاک خودرو را نیز در دراز مدت تحت تاثیر قرار می‌دهند. سامانه‌های پایش، تحلیل، و تشخیص رخدادهای رانندگی می‌توانند نقش مهمی درخصوص امنیت جاده‌ای و اثرات محیط‌زیستی خودروها ایفا کنند. هدف این پژوهه، طراحی و پیاده‌سازی سامانه تشخیص و تحلیل رخدادهای پایه رانندگی است که راننده در طول عمل رانندگی انجام می‌دهد. از داده‌های حرکتی که توسط حسگرهای مختلف جمع‌آوری می‌شوند بعنوان ورودی و مدل‌های یادگیری ماشین مبتنی بر شبکه عصبی بازگشتی بعنوان مدل تشخیص رخداد سامانه استفاده شده است. پس از انجام پیش پردازش بر روی داده‌ها مانند هموارسازی و انتقال دستگاه مختصات، مدل می‌تواند رخدادهای مهم مانند ترمز، شتاب، و چرخش را با دقت حدود ۷۷٪ تشخیص دهد. پس از تعیین رخدادها، پارامترهای مرتبط با هر عمل مانند مدت زمان و انرژی سیگنال رخداد بررسی شده و پس از مقایسه با توزیع رخدادهای یک رانندگی خوب که توسط دو راننده با سابقه آزمایش شده است، نمره‌ای جهت سنجش کیفیت رانندگی تعیین می‌گردد. این کار تحقیقاتی، به‌واقع، می‌تواند مقدمه توسعه نرم‌افزار یا سخت‌افزارهایی که قادر به شناسایی و تحلیل رفتار رانندگان هستند را به خوبی فراهم کند.

## واژه‌های کلیدی:

حسگر اینرسی، خودرو، تحلیل رانندگی، پردازش سیگنال، یادگیری ماشین، شبکه عصبی بازگشتی

# فهرست مطالب

صفحه	عنوان
۱	۱ تعریف مسئله، راه حل پیشنهادی و تعریف نیازمندی‌ها
۲	۱-۱ مقدمه
۴	۲-۱ تعریف مسئله
۵	۳-۱ راه حل پیشنهادی
۵	۴-۱ نیازمندی‌های پروژه
۵	۱-۴-۱ جمع‌آوری داده
۶	۲-۴-۱ مدل تشخیص رخداد
۷	۲ اجزا و تکنولوژی‌ها
۸	۱-۲ مقدمه
۸	۲-۲ داده حرکتی و واحد اندازه‌گیری اینرسی
۹	۱-۲-۲ شتاب‌سنج
۹	۲-۲-۲ چرخش‌نما
۱۱	۳-۲-۲ مغناطیس‌سنج
۱۳	۴-۲-۲ حسگر IMU
۱۳	۵-۲-۲ آردوبینو
۱۵	۶-۲-۲ راهاندازی حسگر IMU
۱۷	۷-۲-۲ نرم‌افزار SensorLog
۱۸	۳-۲ یادگیری ماشین
۱۹	۱-۳-۲ یادگیری با نظارت
۱۹	۲-۳-۲ یادگیری بدون نظارت
۲۰	۳-۳-۲ یادگیری تقویتی
۲۰	۴-۳-۲ مسئله دسته‌بندی
۲۱	۵-۳-۲ شبکه‌های عصبی مصنوعی
۲۴	۶-۳-۲ شبکه LSTM
۲۸	۴-۲ کتابخانه‌های نرم‌افزاری

۲۸	TensorFlow	۱-۴-۲
۲۹	Keras	۲-۴-۲
۲۹	PyTorch	۳-۴-۲
۲۹	Caffe	۴-۴-۲
۲۹	scikit-learn	۵-۴-۲
۳۱	جمع‌بندی	۵-۲
۳۲	۳ طراحی و پیاده‌سازی	
۳۳	۱-۳ مقدمه	
۳۴	۲-۳ داده حسگرهای حرکتی	
۳۴	۱-۲-۳ رخدادهای ترمز و شتاب	
۳۶	۲-۲-۳ رخدادهای چرخش	
۴۰	۳-۳ جمع‌آوری داده	
۴۰	۱-۳-۳ هماهنگ سازی دستگاه مختصات	
۴۴	۴-۳ مدل تشخیص رخداد	
۴۴	۱-۴-۳ طراحی مدل LSTM	
۴۶	۲-۴-۳ دسته‌بند ترمز و شتاب	
۴۷	۳-۴-۳ دسته‌بند چرخش به راست و چپ	
۴۸	۴-۴-۳ دسته‌بند دور زدن	
۴۸	۵-۳ پس پردازش داده‌ها	
۵۱	۳-۵-۳ میزان ناگهانی (شدید) بودن عمل	
۵۱	۶-۳ ارزیابی و انتخاب مدل دسته‌بند	
۵۲	۱-۶-۳ معیار امتیاز اف-۱	
۵۳	۲-۶-۳ معیار نسبت اشتراک به اجتماع	
۵۴	۳-۶-۳ بررسی خطای دسته‌بندی	
۵۷	۷-۳ برنامه بررسی نوع رانندگی	
۶۲	۸-۳ جمع‌بندی	
۶۳	۴ جمع‌بندی، نتیجه‌گیری و پیشنهادات	

## فهرست مطالب

۶۴	۱-۴ جمع‌بندی و نتیجه‌گیری
۶۵	۲-۴ کارهای آینده
۶۷	منابع و مراجع

# فهرست اشکال

صفحه

شکل

۱-۲ دستگاه مختصات تلفن همراه	۸
۲-۲ داده حسگر شتاب سنج ساکن بر روی سطح صاف	۱۰
۳-۲ چرخش نما-جیمبال	۱۱
۴-۲ داده حسگر چرخش نما هنگام چرخش حول محور X	۱۲
۵-۲ داده حسگر مغناطیس سنج هنگام تغییر جهت	۱۲
۶-۲ حسگر IMU ۹ محوره [۱]	۱۴
۷-۲ برد آردوینو UNO [۲]	۱۵
۸-۲ نحوه اتصال حسگر IMU با میکروکنترلر آردوینو [۴]	۱۶
۹-۲ نمونه خروجی در محیط توسعه آردوینو که میزان چرخش، جهت مغناطیسی، ارتفاع، و دما را گزارش می‌کند [۴]	۱۶
۱۰-۲ طرح یک نورون زیستی [۵]	۲۲
۱۱-۲ مدل محاسباتی ساده شده نورون [۵]	۲۲
۱۲-۲ شکل ساختار شبکه عصبی کاملاً متصل در مدل پیشخور [۵]	۲۳
۱۳-۲ ساختار فشرده بخشی از یک شبکه عصبی بازگشتی و وجود حلقه در آن ساختار	۲۳
۱۴-۲ ساختار بازشده حلقه بخشی از شبکه عصبی بازگشتی	۲۳
۱۵-۲ معماری داخلی یک سلول LSTM	۲۵
۱۶-۲ وضعیت سلول در یک سلول LSTM	۲۵
۱۷-۲ دروازه فراموشی یک سلول LSTM	۲۶
۱۸-۲ دروازه ورودی یک سلول LSTM	۲۶
۱۹-۲ بروزرسانی وضعیت سلول یک سلول LSTM	۲۷
۲۰-۲ دروازه خروجی یک سلول LSTM	۲۸
۱-۳ معماری کلی پروژه	۳۳
۲-۳ دستگاه مختصات تلفن همراه	۳۴
۳-۳ رخداد ترمز	۳۵
۴-۳ رخداد شتاب	۳۵

۳۶	۵-۳ داده حسگر چرخش نما هنگام انجام رخداد ترمز
۳۶	۶-۳ داده حسگر مغناطیس سنج هنگام رخداد ترمز
۳۷	۷-۳ داده حسگر چرخش نما هنگام رخداد چرخش به راست
۳۷	۸-۳ میزان چرخش هنگام رخداد چرخش به راست
۳۷	۹-۳ داده حسگر چرخش نما هنگام رخداد چرخش به چپ
۳۸	۱۰-۳ میزان چرخش هنگام رخداد چرخش به چپ
۳۸	۱۱-۳ داده حسگر چرخش نما هنگام رخداد دور زدن
۳۹	۱۲-۳ میزان چرخش هنگام رخداد دور زدن
۳۹	۱۳-۳ جهت حرکت نسبت به قطب شمال مغناطیسی، چرخش به راست
۴۰	۱۴-۳ جهت حرکت نسبت به قطب شمال مغناطیسی، دور زدن
۴۱	۱۵-۳ مسیر انتخابی ۱
۴۲	۱۶-۳ مسیر انتخابی ۲
۴۲	۱۷-۳ مسیر انتخابی ۳
۴۳	۱۸-۳ دستگاه مختصات خودرو و تلفن همراه
۴۵	۱۹-۳ ساختار ورودی شبکه LSTM
۴۶	۲۰-۳ معماری مدل دسته بند ترمز و شتاب
۴۶	۲۱-۳ نمونه ورودی به مدل دسته بند ترمز و شتاب
۴۷	۲۲-۳ نمونه پاسخ ورودی به مدل دسته بندی ترمز و شتاب
۴۷	۲۳-۳ نمونه ورودی به مدل دسته بندی چرخش
۴۷	۲۴-۳ نمونه پاسخ ورودی به مدل دسته بندی چرخش
۴۸	۲۵-۳ معماری مدل دسته بند دور زدن
۴۸	۲۶-۳ نمونه ورودی به مدل دسته بندی دور زدن
۴۸	۲۷-۳ نمونه پاسخ ورودی به مدل دسته بند دور زدن
۴۹	۲۸-۳ مقایسه انرژی سیگنال های رخداد شتاب و ترمز
۵۰	۲۹-۳ مقایسه طول زمانی رخداد شتاب و ترمز
۵۰	۳۰-۳ مقایسه انرژی سیگنال های رخداد چرخش
۵۰	۳۱-۳ مقایسه طول زمانی سیگنال های رخداد چرخش
۵۱	۳۲-۳ تعریف حدود آستانه ناگهانی بودن رخدادها

۳-۳۳-۳ مدل پیشنهادی اعتبارسنجی متقابل	۵۲
۳-۳۴-۳ رابطه نسبت اشتراک به اجتماع	۵۴
۳-۳۵-۳ نمونه ای از پیش‌بینی رخداد	۵۴
۳-۳۶ نمودار سیگنال حسگر شتاب‌سنج که بصورت اشتباه بعنوان رخداد ترمز دسته‌بندی شده است	۵۵
۳-۳۷-۳ رخدادهای شتاب/ترمز انجام شده	۵۵
۳-۳۸-۳ رخدادهای چرخش انجام شده	۵۶
۳-۳۹-۳ رخدادهای دور زدن انجام شده	۵۶
۳-۴۰ نمودار Histogram پیش‌امتیاز رانندگی خوب	۵۸
۳-۴۱ نمودار تابع توزیع تجمعی پیش‌امتیاز رانندگی خوب	۵۹
۳-۴۲-۳ مشخص نمودن مقدار تابع توزیع تجمعی دو نمونه داده	۶۰
۳-۴۳-۳ توزیع رخدادها در نمونه داده [۶]	۶۰
۳-۴۴-۳ توزیع رخدادها در نمونه داده تهییه شده	۶۱
۳-۴۵-۳ پیش-امتیاز به ازای هر ۵ رخداد داده [۶]	۶۱
۳-۴۶-۳ امتیاز نسبت داده شده به رانندگی داده [۶]	۶۱
۳-۴۷-۳ پیش-امتیاز به ازای هر ۵ رخداد داده ضبط شده	۶۲
۳-۴۸-۳ امتیاز نسبت داده شده به رانندگی داده ضبط شده	۶۲

## فصل اول

# تعریف مسئله، راه حل پیشنهادی و تعریف نیازمندی‌ها

## ۱-۱ مقدمه

عمل رانندگی از مجموعه‌ای از رخدادهای پایه مانند ترمز کردن، شتاب گرفتن، دور زدن و ... تشکیل شده است که سرنشینان خودرو با مشاهده نحوه انجام هر کدام از این اعمال و میزان تکرار آن‌ها می‌توانند به درک قابل قبولی از کیفیت رانندگی و رفتار راننده دست یابند. عنوان مثال، استفاده از ترمزهای ناگهانی حتی با فواصل زمانی زیاد یا تعداد زیادی از ترمزهای معمولی در مدت زمان کوتاه، نمایانگر رفتار نامطلوب راننده می‌باشد و تکرار این‌گونه رفتارها در رانندگی یک فرد منجر به بروز مشکلات مختلفی خواهد شد.

در کشورهای بسیاری موضوع تصادفات رانندگی عنوان یکی از عوامل اصلی مرگ و میر شناخته می‌شود و نگرانی از این مسئله در حال افزایش است. علاوه بر موضوع امنیت، نحوه رانندگی افراد ارتباط مستقیم با الگوی مصرف سوخت و انرژی در خودروها دارد که عنوان یک چالش برای محیط زیست و صنعت خودروسازی مطرح است. در حوزه ماشین‌های خودران، که هدف نهایی آن خودکارسازی تمام عملکردهای یک خودرو بدون دخالت انسان است، در اختیار داشتن درکی درست از رفتار راننده در شرایط مختلف، الگوی مناسبی برای پیش بردن این تکنولوژی می‌باشد و می‌تواند توسعه سامانه‌های دستیار راننده در سطوح مختلف را تسريع کند. از داده مربوط به نوع و رفتار رانندگی رانندگان می‌توان به نمایه و امضای رانندگی یکتاً هر فرد دست یافت که این اطلاعات در شرکت‌های بیمه برای بهبود فرآیندهای ارزیابی ریسک و اختصاص دادن بیمه خودروها، همچنین در شرکت‌های تاکسی‌رانی و کرایه خودرو جهت ارزیابی رانندگان و تشخیص تخلفات استفاده می‌شود. بنابراین، با درنظر گرفتن موارد بالا توسعه راهکارهای بررسی رخدادها، سبک، و رفتار رانندگی رانندگان برای محققان موضوع جالبی است که کاربردهای متعددی را فراهم می‌نماید.

برای توسعه سامانه‌های تحلیل و بررسی رانندگی روش‌های مختلفی ارائه شده است که می‌توان آن‌ها را به ۲ دسته اصلی تقسیم نمود: [۱۳]:

۱. روش‌های مبتنی بر داده‌های حرکتی ماشین (ماشین-محور)
۲. روش‌های مبتنی بر داده‌های مربوط به راننده (راننده-محور)

در روش‌های ماشین-محور، از حسگرهای مختلفی در ارتباط با دینامیک ماشین برای دسته‌بندی سبک و رفتار راننده در طول سفر استفاده می‌شود. داده‌های این روش از منابع مختلفی مانند

## فصل اول: تعریف مسئله، راه حل پیشنهادی و تعریف نیازمندی‌ها

حسگر اینرسی (شتاب سنج<sup>۱</sup> و چرخش‌نما<sup>۲</sup>، سرعت‌سنج، دوربین و ... تهیه می‌گردد. اما در رویکرد دوم، از داده‌های مرتبط با شخص راننده استفاده می‌شود: داده‌های مربوط به ضربان قلب، ردیابی نگاه راننده، و موقعیت قرارگیری سر و بدن فرد. در روش‌هایی نیز می‌توان از ترکیب رویکردهای بیان شده استفاده نمود.

بطور کلی، روش‌های ماشین-محور بدلیل سهولت ثبت و پردازش داده‌ها و وجود روش‌های مختلف جهت جمع‌آوری داده، بیشتر مورد توجه قرار گرفته‌اند. این روش‌ها را می‌توان براساس منبع‌شان به ۲ دسته داخلی و خارجی تقسیم نمود. روش‌های داخلی از داده‌های گذرگاه CAN<sup>۳</sup> استفاده می‌نمایند. گذرگاه CAN استانداردی است که در وسایل نقلیه استفاده شده و اجازه می‌دهد تا میکروکنترلرها و دستگاه‌ها بدون سامانه میزبان با یکدیگر ارتباط برقرار کنند. از طریق درگاه OBD<sup>۴</sup> در خودروها می‌توان به اطلاعاتی مانند سرعت، وضعیت چراغ‌ها، وضعیت پدال‌ها و ... دسترسی داشت. روش خارجی از دستگاه‌های بیرونی مانند حسگرهای اینرسی یا جعبه تلماتیک استفاده می‌کند.

همانطور که اشاره شد با استفاده از دستگاه‌های مختلفی که در ادامه توضیحات تکمیلی آن نیز آورده خواهد شد، می‌توان داده‌های حرکتی را جمع‌آوری کرد. در این پژوهه از حسگرهای اینرسی بعنوان منبع جمع‌آوری داده‌های حرکتی و همچنین بعنوان ورودی الگوریتم‌های یادگیری ماشینی جهت تشخیص رخدادهای رانندگی استفاده شده است.

این پژوهه همانند اکثر پژوهه‌های داده-محور و مبتنی بر یادگیری، دارای ۲ فاز آموزش و آزمایش است. هدف فاز آموزش توسعه مدل‌های یادگیری و تعیین پارامترهای مرتبط با مدل‌ها است. در فاز آزمایش از مدل‌هایی که تحت آموزش قرار گرفته‌اند جهت دسته‌بندی داده‌های جدید استفاده خواهد شد. برای فاز آموزشی پژوهه، نیاز است تا به داده‌های برچسب‌گذاری شده دسترسی داشته باشیم که این عمل توسط ضبط صدای راننده هنگام انجام رخدادهای رانندگی انجام شده است و به این ترتیب شروع و پایان رخدادها مشخص می‌شود. همچنین، سعی شده است تا تنوع قابل قبولی از رخدادهای مورد نظر شتاب، ترمز، و چرخش در نظر گرفته شود تا قابلیت تعمیم‌دهی مدل‌ها افزایش یابد.

پس از جمع‌آوری داده‌ها، نیاز است تا با پیش‌پردازش‌هایی بتوان آن‌ها را برای عمل یادگیری

<sup>1</sup>Accelerometer

<sup>2</sup>Gyroscope

<sup>3</sup>Controller Area Network

<sup>4</sup>On-board diagnostics

## فصل اول: تعریف مسئله، راه حل پیشنهادی و تعریف نیازمندی‌ها

آمده نمود. یکی از این پیش پردازش‌ها تبدیل دستگاه مختصات داده از تلفن‌همراه به دستگاه مختصات خودرو است. این عمل از این بابت اهمیت دارد که ممکن است جهت تلفن‌همراه هنگام ضبط داده در وضعیت‌های مختلفی قرار گیرد. عمل دیگری که در فاز پیش پردازش استفاده می‌شود، هموار سازی داده جهت کاهش تاثیر نویز بر روی داده است.

قسمت اصلی فاز آموزش پروژه از مدل‌های یادگیری مبتنی بر شبکه‌های عصبی تشکیل شده است که در بخش پیاده‌سازی به تفصیل توضیح داده خواهد شد. از مدل‌های یادگیری جهت دسته‌بندی داده‌ها و تشخیص رخدادها استفاده می‌شود. پس از تشخیص رخدادهای رانندگی، پارامترهای مرتبط با هر رخداد را از داده استخراج نموده تا با توجه به معیارهای ارائه شده، میزان ناگهانی (شدید) بودن هر رخداد مشخص شود. بر اساس توزیع رخدادها و میزان شدت‌شان، نمره‌ای به هر راننده در طول یک مسیر اختصاص داده می‌شود که نشانگر کیفیت رانندگی است. همانطور که در ابتدا اشاره شد، سعی شده است تعریف کیفیت به همان درک و ارزیابی‌ای که سرنشینان خودرو از رانندگی راننده دارند، نزدیک باشد که در قسمت برنامه بررسی نوع رانندگی کامل‌تر توضیح داده خواهد شد.

این نوشته ابتدا به تعریف مسئله و راه حل پیشنهادی می‌پردازد. سپس تکنولوژی‌ها و مدل‌های یادگیری ماشین‌ای که بر بستر آنان توسعه داده شده است، معرفی شده و پس از آن پیاده‌سازی راه حل پیشنهادی به‌طور کامل شرح داده خواهد شد. در انتها نیز به ارائه نتایج این پروژه و همچنین کارهایی که در آینده جهت بهبود سامانه فعلی می‌توان انجام داد پرداخته خواهد شد.

### **۲-۱ تعریف مسئله**

با توجه به چالش‌های مطرح شده در ارتباط با خودرو و صنایع مرتبط با آن، نیازمند روش‌هایی هستیم تا با نظارت پیوسته بر رانندگی افراد بتوانند اطلاعات مناسبی را در خصوص رفتار راننده، کیفیت رانندگی، و نمایه امضای شخصی رانندگی افراد گزارش دهند. مسئله مطرح شده در این پروژه به‌طور خاص، ارائه راه حل‌ای جهت تعیین نوع و کیفیت رانندگی راننده با در اختیار داشتن داده رانندگی افراد می‌باشد که این داده بصورت سیگنال‌های حرکتی خودرو در اختیار می‌باشد.

## ۳-۱ راه حل پیشنهادی

یک روش موثر که می‌تواند در تامین امنیت جاده و خیابان‌هایی که محل تردد خودروها هستند نقش قابل توجهی ایفا کند، توسعه سامانه‌های دستیار راننده است. این سامانه‌ها قادر به تشخیص رفتار پرخطر و گزارش نمودن کیفیت رانندگی هستند که می‌تواند باعث ارتقا سطح رانندگی افراد نیز بشود. با پیشرفت چشمگیر هوش مصنوعی و یادگیری ماشین در زمینه شناسایی الگو و دسته بندی داده‌ها، می‌توان از بستری که توسط این زمینه فراهم شده است، برای پردازش داده‌ها در کاربردهای مختلف استفاده نمود. راه حل پیشنهادی برای توسعه سامانه تشخیص رفتار راننده، اجرای الگوریتم‌های یادگیری ماشین و شبکه‌های عصبی بر روی داده‌های اینرسی و حرکتی خودروها می‌باشد. این سامانه با استفاده از الگوریتم‌های دسته‌بند که توسط شبکه‌های عصبی توسعه داده شده‌اند، داده‌های حرکتی خودرو را بعنوان ورودی دریافت نموده و رخدادهای مهم یک دوره رانندگی را تشخیص می‌دهد. پردازشی که بر روی رخدادها انجام می‌شود، می‌تواند نمره‌ای را (بعنوان کیفیت) در هر واحد از رانندگی که در ادامه توضیح داده خواهد شد اعلام کند. این سامانه همچنین اطلاعاتی را در خصوص مدت زمان، میزان ناگهانی بودن، و تعداد رخدادهای رانندگی جهت بررسی‌های بیشتر ارائه می‌دهد.

## ۴-۱ نیازمندی‌های پروژه

با توجه به موارد مطرح شده در قسمت تعریف مسئله و راه حل پیشنهادی، نیازمندی‌هایی به وجود می‌آید که در ادامه به آن‌ها پرداخته شده است.

### ۱-۴-۱ جمع‌آوری داده

داده ورودی، یکی از مهم‌ترین بخش‌های این سامانه می‌باشد. برای دریافت و ثبت داده به حسگرهای شتاب‌سنج، مغناطیس‌سنج<sup>۵</sup>، و چرخ‌نمای نیاز داریم که بتوانند بطور پیوسته داده‌ها را ضبط کرده و برای پردازش برونو خط نگه‌داری کنند. داده‌ها را می‌توان با استفاده از حسگرهای مختلف و واحد اندازه‌گیری اینرسی<sup>۶</sup> جمع‌آوری نمود.

<sup>5</sup>Magnetometer

<sup>6</sup>Inertial Measurement Unit

## ۲-۴-۱ مدل تشخیص رخداد

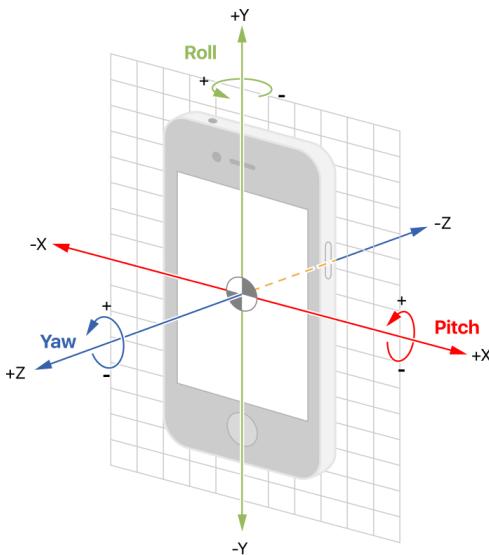
یکی از نیازمندی‌های پروژه جهت تحلیل مناسب رفتار و کیفیت رانندگی افراد، داشتن مدلی برای تشخیص رخداد و اعمال پایه رانندگی است. هسته اصلی مدل، الگوریتم‌های یادگیری ماشین و ساختار شبکه‌های عصبی بازگشتی هستند که با پردازش داده‌های جمع‌آوری شده به دسته‌بندی رخدادها و تشخیص آن‌ها در طول مسیر می‌پردازند. دسته‌های قابل تشخیص برای مدل توسعه داده شده عبارتند از: ترمز کردن، شتاب گرفتن، چرخش به راست و چپ، و دور زدن<sup>۷</sup> (چرخش یو شکل)

---

<sup>7</sup>U-Turn

## فصل دوم

# اجزا و تکنولوژی‌ها



شکل ۲-۱: دستگاه مختصات تلفن همراه

## ۱-۲ مقدمه

در این قسمت اجزا، تکنولوژی، و نرم‌افزارهای مورد استفاده در پروژه معرفی خواهند شد.

## ۲-۲ داده حرکتی و واحد اندازه‌گیری اینرسی

اولین گام در بررسی رفتار عوامل در حال حرکت، تحلیل داده‌های مربوط به حسگرهای حرکتی است. معمولاً این داده‌ها از ترکیب ۳ حسگر شتاب‌سنج، چرخش‌نما، و مغناطیس‌سنج بدست می‌آیند که به منظور تشخیص روند حرکت و نیروی اعمال شده استفاده می‌شود. ترکیب این حسگرها اطلاعات مفیدی در ارتباط با موقعیت و جهت حرکت عامل‌های مختلف در اختیار ما قرار می‌دهد. واحد اندازه‌گیری اینرسی، قطعه الکترونیکی است که شامل این ۳ حسگر بوده و با استفاده از آن‌ها جهت، سرعت، و نیروی گزارش را گزارش می‌دهد. این قطعه در کاربردهای مختلفی از جمله بازشناسی نوع فعالیت انسان<sup>۱</sup>، سامانه‌های ناوبری، تشخیص و تحلیل حرکت خودرو و ... مورد استفاده قرار می‌گیرد.

نمونه سیگنال و داده حسگرها برای توضیحات بیشتر اضافه شده است. از آنجاییکه داده‌ها مربوط به حسگرهای تعییه شده درون تلفن همراه می‌باشد، دستگاه مختصات استاندارد تلفن همراه جهت تفسیر اطلاعات بیان شده در شکل ۲-۳ مشخص شده است.

<sup>۱</sup>Human Activity Recognition

## ۱-۲-۲ شتاب‌سنج

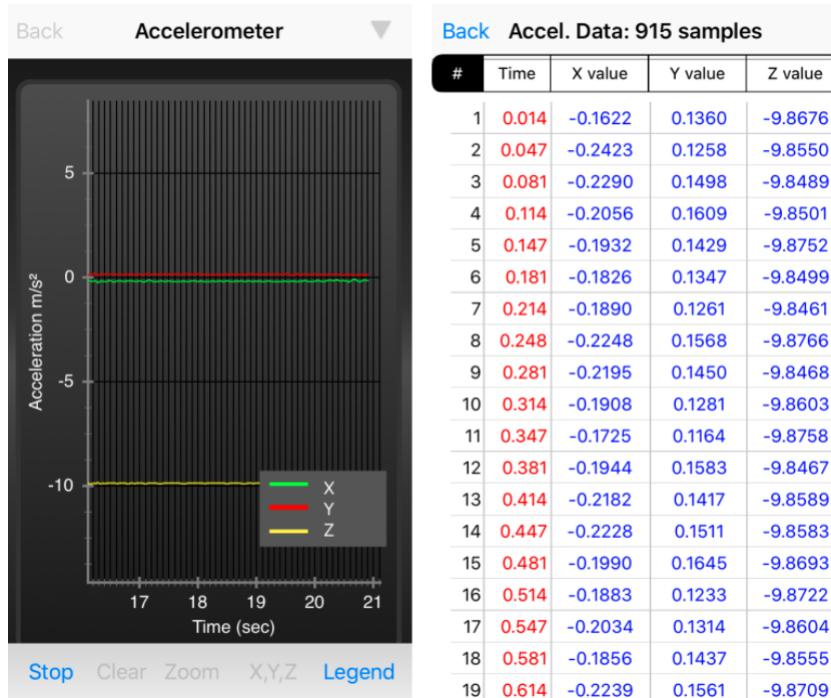
شتاب‌سنج دستگاه الکترومکانیکی است که می‌تواند نیروهای ایستا (جادبه) و پویا (ارتعاشات و حرکات) ناشی از شتاب را اندازه‌گیری کند. این قطعه اندازه‌گیری نیرو را در ۳ محور انجام می‌دهد. بطور مثال، شتاب‌سنج برای سطحی که ساکن نسبت به سطح زمین قرار گرفته است شتابی برابر  $g$  به سمت بالا را نشان خواهد داد که مقدار گرانش زمین حدود ۹.۸ متر بر مجدور ثانیه است. به طور مفهومی، یک شتاب‌سنج مانند یک جسم میراکننده روی یک فنر عمل می‌کند. هنگامی که شتاب‌سنج با شتابی حرکت می‌کند، جسم به اندازه‌ای جابجا می‌شود که نیروی وارد شده از فنر به جسم، جسم را با شتابی برابر شتاب بدنه شتاب‌سنج حرکت دهد. سپس با اندازه‌گیری میزان جابجایی مقدار شتاب اندازه‌گیری می‌شود. این حسگر در اکثر تلفن‌های هوشمند نیز تعبیه شده است تا امکانات متعددی را برای کاربران فراهم کند، مانند تغییر جهت دادن برنامه‌های کاربردی از حالت عمودی به افقی و نمایش سرعت در هنگام رانندگی. علاوه بر آن در صنایع مختلفی مانند پزشکی، حمل و نقل، ساختمان، و ... استفاده می‌شود.

در صورت قرار دادن این حسگر در خودرو بطوریکه جهت  $X$  در جهت حرکت خودرو قرار بگیرد، از میزان تغییرات در شتاب محور  $x$  می‌توان رخدادهای ترمز و شتاب گرفتن را تشخیص داد. نمونه سیگنال شتاب‌سنج برای حالتی که حسگر بر روی سطح صاف و ساکن قرار دارد در شکل ۲-۲ آمده است. همانطور که در نمودار و جدول مشخص شده است، برای محورهای  $Y$ ،  $Z$  مقدار شتاب بسیار ناچیز و حدود صفر است، در حالیکه شتاب در محور  $X$  حدود شتاب گرانش زمین است و بدلیل جهت، علامت آن منفی است. بسته به نوع حسگر، خروجی با واحد متر بر مجدور ثانیه یا بعنوان ضریبی از عدد  $g$  گزارش می‌شود.

## ۲-۲-۳ چرخش‌نما

شتاب‌سنج می‌تواند حرکت در جهت مشخص دهد، اما نمی‌تواند چرخش در حول محور خاصی را اندازه‌گیری کند. به همین دلیل برای تشخیص دقیق‌تر حرکت، چرخش، و موقعیت عامل موردنظر از حسگر چرخش‌نما استفاده می‌گردد. چرخش‌نما، دستگاهی است که از آن برای اندازه‌گیری جهت و سرعت زاویه‌ای استفاده می‌گردد. از نظر مفهومی، چرخی دور و پرسرعت دارای محور آزاد است که می‌تواند در هر جهتی قرار گیرد.

در ک ماهیت عملکرد چرخش‌نما این موضوع را مشخص می‌کند که چرا برای جهت‌یابی، از



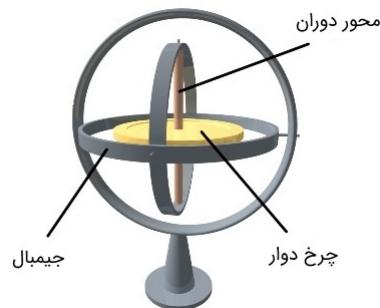
شکل ۲-۲: داده حسگر شتاب‌سنج ساکن بر روی سطح صاف

یک دیسک دور (که از طریق یک موتور تغذیه می‌شود) درون یک قاب فلزی (جیمبال) استفاده می‌شود. دیسک دور درون قاب فلزی نصب می‌شود تا هیچ‌گونه گشتاور خارجی به آن وارد نشود. بنابراین، جهت‌گیری آن، به جز یک مقدار ناچیز، تغییری نخواهد داشت. همین عامل موجب می‌شود چرخش‌نما در سیستم‌های ناوبری کشتی‌ها و قایق‌ها به وفور به کار رود. در این حالت، حتی اگر مسیر حرکت کشتی هم عوض شود، جهت‌گیری چرخش‌نما بدون تغییر می‌ماند. شکل ۳-۲ نشان‌دهنده سامانه چرخش‌نما-جیمبال می‌باشد.

کاربردهای چرخش‌نما شامل ناوبری و پایدارسازی وسایل نقلیه هوایی بدون سرنشین<sup>۲</sup>، زمانی که قطب‌های مغناطیسی کار نمی‌کنند (مانند موقعیت تلسکوپ هابل) یا به اندازه‌ی کافی دقیق نیستند، می‌باشد. به دلیل دقیق‌تر بودن این دستگاه، چرخش‌نماها همچنین در حفظ جهت در معدن کاری تونل‌ها هم به کار می‌روند. در تلفن‌های همراه نیز چرخش‌نما کاربرد دارد و تحت عنوان چرخش خودکار صفحه می‌توان آن را فعال نمود.

چرخش‌نماهای امروزی برخلاف چرخش‌نمای استفاده شده در هواپیماهای قدیمی، مکانیکی نیست و از محورهای تراز و دیسک‌های چرخنده در ساخت آن استفاده نمی‌شود. عنوان مثال،

<sup>2</sup>Unmanned Aerial Vehicles



شکل ۳-۲: چرخش‌نما-جیمبال

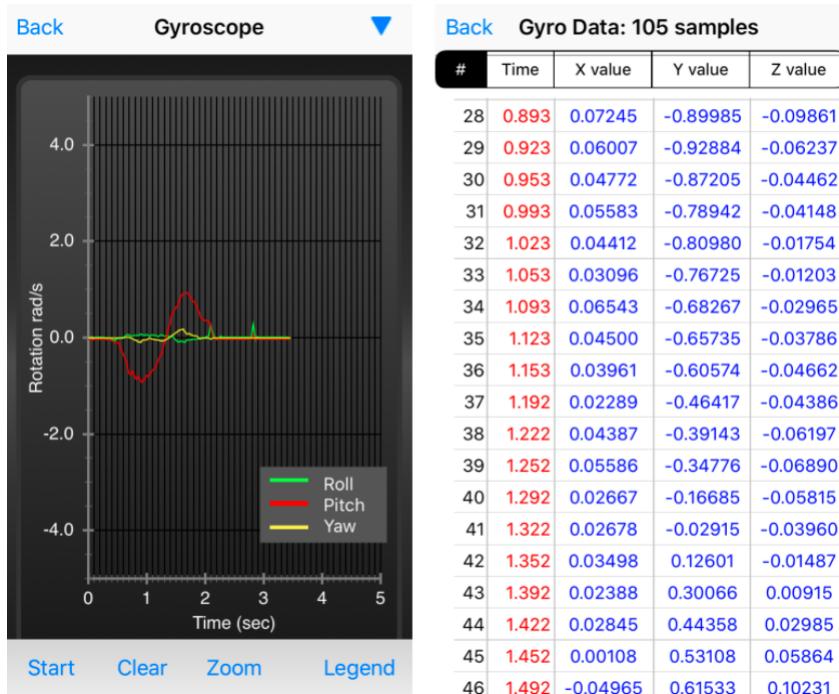
چرخش‌نمای تعبیه شده در تلفن‌های همراه از نوع میکروالکترومکانیکی است که بطور خلاصه به آن MEMS<sup>۳</sup> گفته می‌شود. این تکنولوژی جدید همان مفهوم چرخش‌نماهای قدیمی را در اندازه‌ای کوچک بر روی مدارهای الکترونیکی پیاده کرده است که می‌تواند در یک گوشی همراه قرار بگیرد. از این حسگر برای تشخیص رخدادهای گردش به چپ، راست، و دور زدن در این پروژه استفاده می‌شود.

شکل ۴-۲ نشان‌دهنده نمونه سیگنال چرخش‌نما برای حالتی که حسگر حول محور X چرخش داشته است. چرخش بر حسب رادیان بر ثانیه گزارش شده است که با استفاده از تبدیل واحد می‌توان به میزان زاویه چرخش نیز دست یافت. در حالت سکون تمام مقادیر برابر با صفر هستند و در صورت چرخش حول هر کدام از محورها مقادیر چرخش بدست خواهد آمد.

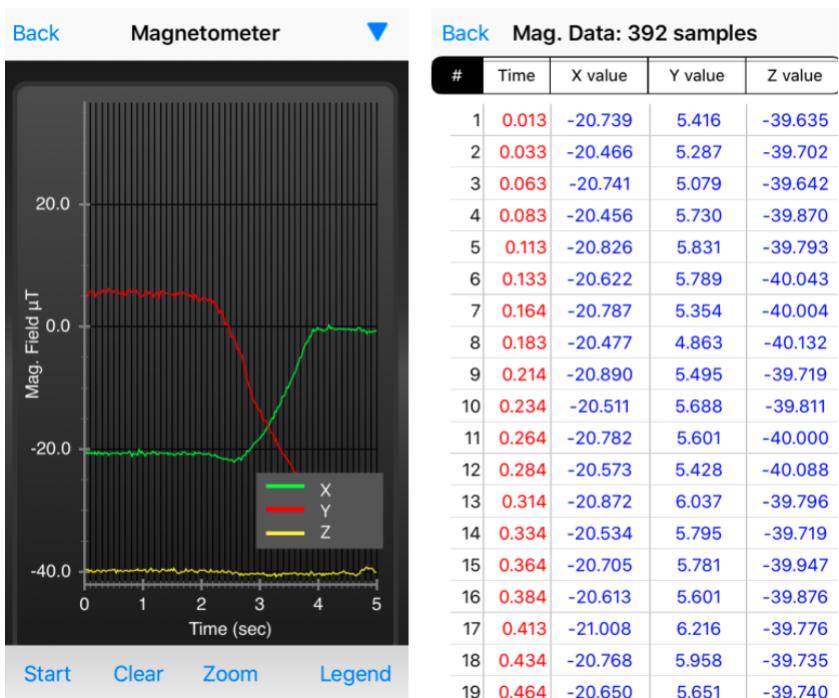
### ۳-۲-۲ مغناطیس‌سنج

سه گانه‌ی حسگرهایی که مسؤولیت شناسایی موقعیت یک عامل را بر عهده دارند با مغناطیس‌سنج کامل می‌شود. حسگر مغناطیس‌سنج، ابزاری است که جهت، قدرت و تغییر نسبی میدان مغناطیسی در یک مکان خاص را اندازه‌گیری می‌کند. قطب‌نما نمونه ساده یک مغناطیس‌سنج است که جهت یک میدان مغناطیسی را نشان می‌دهد. مغناطیس‌سنج‌ها به شکل گسترده‌ای در اندازه‌گیری میدان مغناطیسی زمین و در نقشه‌برداری ژئوفیزیکی به منظور شناسایی انواع مختلف ناهنجاری مغناطیسی و همچنین در مسائل نظامی جهت تشخیص زیردریایی‌ها به کار می‌روند. در عمل دور زدن و گردش، خودرو جهات متفاوتی را اتخاذ می‌کند و با تغییرات در اندازه‌گیری جهات جغرافیایی همراه است. بنابراین، این حسگر نیز می‌تواند به تشخیص رخدادهای دور زدن

<sup>3</sup>Microelectromechanical systems



شکل ۴-۲: داده حسگر چرخش‌نمای هنگام چرخش حول محور X



شکل ۵-۲: داده حسگر مغناطیس‌سنج هنگام تغییر جهت

و گردش کمک کند. نمونه داده مربوط به تغییرات این حسگر را می‌توان در شکل ۵-۲ مشاهده نمود. واحد اندازه‌گیری این داده بر حسب میکروتسلا است.

## ۴-۲-۲ حسگر IMU

حسگرهای اینرسی معمولاً از ترکیب شتاب‌سنج و چرخش‌نما که هر کدام از آن‌ها در ۳ محور،  $x$ ,  $y$ ,  $z$  متغیرها را اندازه‌گیری می‌کنند تشکیل می‌شود. ترکیب آن‌ها حسگرهای ۶ محوره (۶ درجه آزادی) را بوجود می‌آورد. تعداد درجه آزادی نشان‌دهنده تعداد بیشینه مدل‌های متفاوتی است که یک شی می‌تواند در فضای ۳ بعدی حرکت کند. بعنوان مثال ۶ درجه آزادی شامل ۳ محور در یک صفحه و ۳ محور چرخش حول هر کدام از آن محورها می‌باشد. با اضافه شدن مغناطیس‌سنج ۳ محوره می‌توان مدل‌های ۹ محوره نیز ایجاد کرد.

در شکل ۶-۲ مدل ۹ محوره حسگر IMU آورده شده است. این حسگر ساخت شرکت Adafruit [۱]، از ترکیب ۲ حسگر چرخش‌نمای مدل L3GD20H و شتاب‌سنج/مغناطیس‌سنج مدل LSM303 ساخته شده است. حسگر شتاب‌سنج شامل ساختارهای ریزماشین<sup>۴</sup> بر روی ویفر<sup>۵</sup> سیلیکون است که برای اندازه‌گیری شتاب و میدان مغناطیسی طراحی شده است. بازه خروجی شتاب سنج تا  $16g$  +/-<sup>6</sup> ، مغناطیس سنج تا  $8.1 \text{ gauss}$  +/-<sup>6</sup> و چرخش‌نما تا  $2000 \text{ dps}$  +/-<sup>6</sup> می‌باشد. حسگرهای IMU دارای ابعاد کوچکی هستند و بطور خاص، طول و عرض این حسگر به ترتیب ۳۸ و ۲۳ میلیمتر است و وزن آن ۲.۸ گرم می‌باشد. مشخصه دیگر این آن بهره‌مند بودن از گذرگاه I2C برای انتقال داده و ارتباط با دستگاه‌های جانبی است. در قسمت بعد نحوه راهاندازی و ارتباط این حسگر با استفاده از میکروکنترلر آردوینو<sup>۷</sup> شرح داده خواهد شد.

## ۵-۲-۲ آردوینو

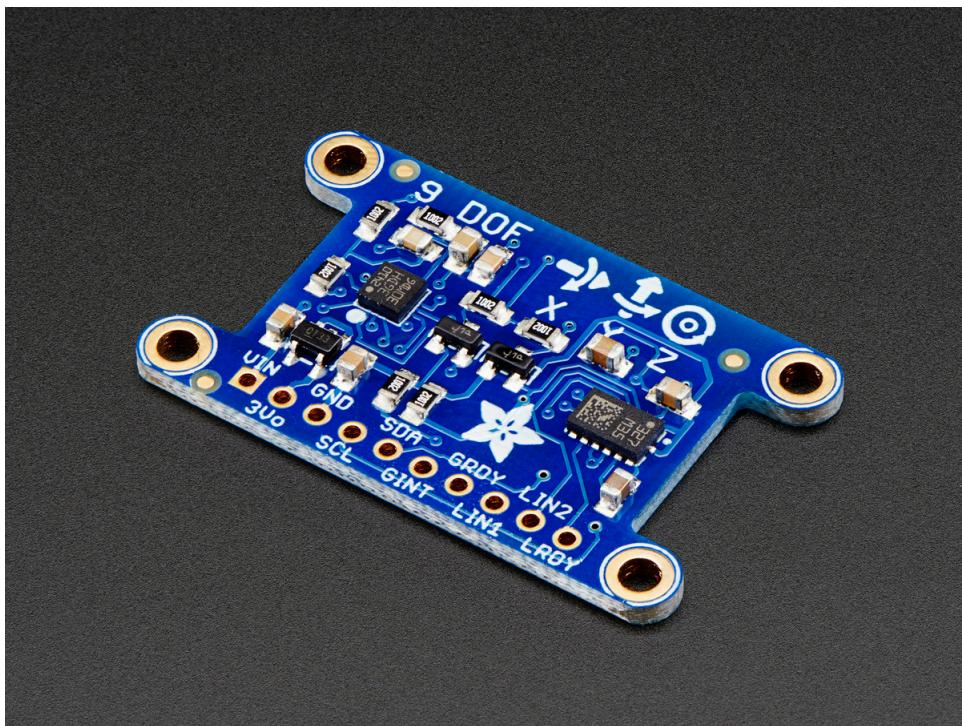
یکی از راههای دریافت داده از حسگرهای استفاده از میکروکنترلرها است. میکروکنترلرها نوعی ریزپردازنده هستند که از اجزای مختلفی مانند پردازنده، حافظه، درگاه‌های ورودی و خروجی و ... تشکیل شده‌اند و به تنها‌یی می‌توانند ابزارهای دیگر را کنترل کنند. میکروکنترلرهای آردوینو [۲] یکی از سخت‌افزارهای پراستفاده در این زمینه هستند و برای ساخت محصولات آزمایشی و آموزشی مناسب می‌باشند. مدار الکتریکی توصیف شده از این پلتفرم برای دریافت داده از حسگر استفاده می‌کند. در این قسمت سعی شده است تا در مورد آردوینو توضیح خلاصه

<sup>4</sup>micro-machined

<sup>5</sup>wafer

<sup>6</sup>Degree per Second

<sup>7</sup>Arduino



شکل ۲-۶: حسگر ۹ IMU محوره [۱]

ای آورده شود.

آردوینو یک پلتفرم سخت‌افزاری و نرم‌افزاری متن‌باز است که طراحی و ساخت سریع و آسان وسایل تعاملی دیجیتال را فراهم می‌کند. این پلتفرم شامل یک میکروکنترلر تک‌بردی متن‌باز است که قسمت سخت‌افزار آردوینو را تشکیل می‌دهد. علاوه بر آن، محیط یکپارچه توسعه نرم‌افزار آردوینو<sup>۸</sup> که به منظور برنامه‌نویسی برای برد‌های آردوینو طراحی شده است و یک بوت‌لودر نرم‌افزاری که بر روی میکروکنترلر بارگذاری می‌شود نیز جزو پلتفرم هستند. اغلب برد‌های آردوینو که تمام آن‌ها سخت‌افزار متن‌باز هستند بر پایه میکرو کنترلر AVR و تعداد کمی از برد‌های آردوینو بر پایه میکروکنترلرهای ARM طراحی شده‌اند. به عنوان مثال، برد آردوینو UNO، شکل ۷-۲، که پرکاربردترین برد آردوینو و برد پایه آردوینو در اکثر دوره‌های آموزش آردوینو است، بر پایه میکروکنترلر AVR ATmega328 ساخته شده است. برای توضیح راه‌اندازی حسگر اینرسی نیز از برد UNO استفاده شده است. این برد دارای رابط USB جهت بارگذاری برنامه و ارتباط با کامپیوتر، ۶ پین ورودی آنالوگ و همچنین ۱۴ پین ورودی/خروجی دیجیتال است که کاربر را قادر می‌سازند تا برد آردوینو را به قطعات، حسگرهای برد و مازول‌های دیگری

<sup>8</sup>Arduino Integrated Development Environment



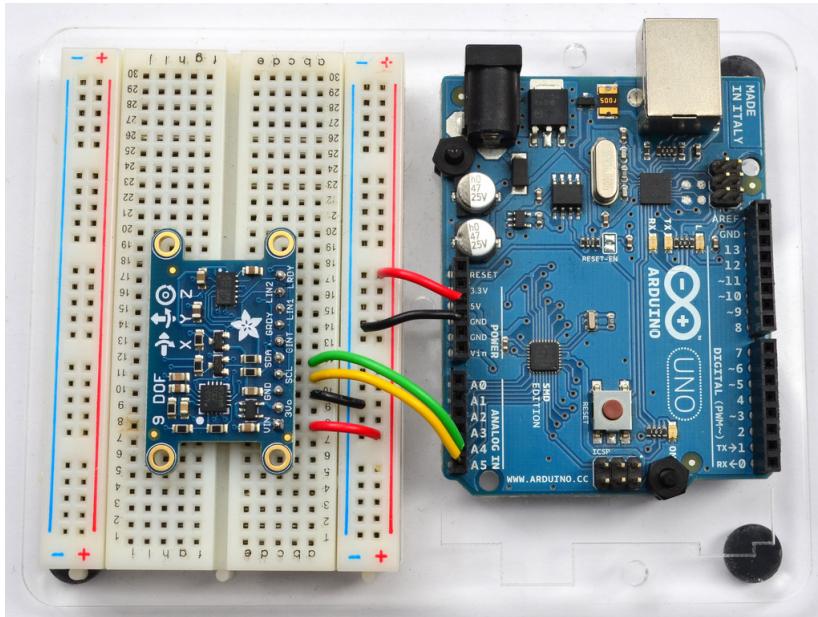
[۲] شکل ۲-۷: برد آردوینو UNO

متصل کند. تعداد ورودی خروجی‌های آنالوگ و دیجیتال در مدل‌های مختلف برد‌های آردوینو با توجه به میکروکنترلر اصلی استفاده شده بر روی برد متفاوت است.

## ۶-۲-۲ راهاندازی حسگر IMU

در این قسمت ارتباط بین حسگر و میکروکنترلر آردوینو مدل UNO در شکل ۸-۲ آورده شده است [۴]. همانطور که اشاره شد، حسگر برای انتقال داده از گذرگاه سریال I2C استفاده می‌کند که در برد آردوینو نیز پشتیبانی می‌شود. این پروتکل ارتباطی، داری ۲ خط جهت ارسال و دریافت داده می‌باشد که در برد آردوینو با ۲ پین به نام‌های SDA و SCL مشخص شده است. پین‌های اشاره شده به ترتیب مربوط به خطوط داده و کلاک هستند. برد Master با ارسال کلاک در فواصل زمانی معین باعث می‌شود تا داده در خط داده بین ۲ (یا چند) دستگاه منتفل شود. همچنین ۲ پین به همین نام در حسگر موجود است که در طراحی مدار باید به پین‌های میکروکنترلر متصل شوند. برای تامین ولتاژ و روشن شدن حسگر از خروجی‌های ۵ ولت یا ۳.۳ ولت و پایه GND برد استفاده شده است.

شرکت سازنده این حسگر، برنامه‌ای را بعنوان واسط بین حسگر و میکروکنترلر توسعه داده است که تا کمک آن بتوان به مقادیر جمع‌آوری شده توسط حسگرهای دسترسی داشت. همچنین، این برنامه با استفاده از داده‌های خام حسگرهای اطلاعات مرتبط با جهت و میزان چرخش را به واحد استاندارد تبدیل و گزارش می‌کند. بنابراین با اتصال برد آردوینو به کامپیوتر و بارگذاری برنامه مخصوص این حسگر، می‌توان خروجی حسگر را ثبت و ذخیره نمود. مثالی از خروجی



شکل ۲-۸: نحوه اتصال حسگر IMU با میکروکنترل آردوینو [۴]

```
Roll: -24.96; Pitch: -38.30; Heading: 356.23; Alt: 134.16 m; Temp: 21.68 C
Roll: -25.61; Pitch: -37.47; Heading: 356.12; Alt: 133.47 m; Temp: 21.69 C
Roll: -25.16; Pitch: -38.23; Heading: 355.89; Alt: 133.99 m; Temp: 21.70 C
Roll: -22.40; Pitch: -40.83; Heading: 353.99; Alt: 133.74 m; Temp: 21.72 C
Roll: -23.21; Pitch: -40.39; Heading: 354.90; Alt: 134.44 m; Temp: 21.74 C
Roll: -23.92; Pitch: -39.69; Heading: 354.64; Alt: 134.54 m; Temp: 21.75 C
Roll: -23.39; Pitch: -40.47; Heading: 353.93; Alt: 134.10 m; Temp: 21.76 C
Roll: -23.31; Pitch: -40.20; Heading: 354.02; Alt: 134.55 m; Temp: 21.77 C
Roll: -22.61; Pitch: -40.32; Heading: 354.48; Alt: 134.38 m; Temp: 21.79 C
Roll: -23.58; Pitch: -39.96; Heading: 354.29; Alt: 134.65 m; Temp: 21.80 C
Roll: -24.50; Pitch: -39.04; Heading: 354.62; Alt: 134.13 m; Temp: 21.81 C
Roll: -23.74; Pitch: -40.05; Heading: 355.12; Alt: 134.67 m; Temp: 21.84 C
Roll: -24.39; Pitch: -39.24; Heading: 355.27; Alt: 134.41 m; Temp: 21.85 C
Roll: -24.06; Pitch: -39.93; Heading: 354.71; Alt: 134.07 m; Temp: 21.86 C
Roll: -24.24; Pitch: -39.57; Heading: 355.14; Alt: 133.90 m; Temp: 21.88 C
```

شکل ۲-۹: نمونه خروجی در محیط توسعه آردوینو که میزان چرخش، جهت مغناطیسی، ارتفاع، و دما را گزارش می‌کند [۴]

مورد نظر در شکل ۹-۲ نشان داده شده است. کد برنامه با استفاده از زبان C++ در محیط توسعه آردوینو نوشته شده است.

باتوجه به توضیحات ارائه شده در قسمت‌های قبل، قابلیت ساخت و استفاده از میکروکنترلر و حسگر اینرسی طبق مدار الکتریکی و پروتکل‌های موجود وجود دارد و به راحتی می‌توان از آن سامانه استفاده نمود. در این پروژه بدلیل دسترسی پذیری بیشتر تلفن‌همراه و ایجاد قابلیت گسترش پروژه در مراحل آینده از حسگرهای موجود در آن استفاده شده است. اما الزامی به پیاده‌سازی این سامانه با تلفن‌های همراه نیست و از هر حسگر حرکتی که داده‌های مورد نیاز را فراهم نماید، می‌توان جهت توسعه سامانه تحلیل رفتار راننده استفاده نمود. در قسمت بعد در ارتباط با حسگرها و نرم‌افزار مورد استفاده توضیحاتی آورده شده است.

## ۷-۲-۲ نرم‌افزار SensorLog

نرم‌افزار [۱۰] مخصوص سیستم عامل iOS که قابلیت دسترسی به داده حسگرهای تلفن همراه را می‌دهد. کاربر می‌تواند داده‌های مورد نیازش را پس از ثبت در قالب csv و یا JSON ذخیره کند. داده‌ها می‌توانند تا فرکانس ۱۰۰ هرتز نمونه‌برداری شوند. نوع داده و حسگرهایی که توسط این نرم‌افزار پشتیبانی خواهند شد در زیر آورده شده است:

- اطلاعات در خصوص طول و عرض جغرافیایی و جهت دستگاه
- جهت دستگاه نسبت به قطب حقیقی و مغناطیسی شمال
- اطلاعات حسگر شتاب‌سنجد (در ۳ محور)
- اطلاعات حسگر چرخش‌نما (در ۳ محور)
- اطلاعات حسگر مغناطیس‌سنجد (در ۳ محور)
- میزان چرخش در ۳ محور (Yaw, Roll, Pitch)
- نرخ چرخش دستگاه
- اطلاعات در خصوص تعداد قدم‌ها و مسافت مسیر طی شده
- اطلاعات مربوط به سنجش ارتفاع و فشار
- اطلاعات مرتبط با توان صدای ضبط شده

پس از بررسی تغییرات هر کدام از حسگرها و با توجه به اطلاعاتی که در پژوهش‌های انجام شده به آن اشاره شده است، برای تحلیل رفتار راننده، از داده‌های:

- حسگر شتاب‌سنج در هر ۳ محور (g)

- حسگر چرخش‌نما در هر ۳ محور (رادیان بر ثانیه)

- جهت قطب شمال مغناطیسی (درجه)

- میزان چرخش در هر ۳ محور (رادیان)

استفاده شده است.

## ۳-۲ یادگیری ماشین

در این قسمت سعی شده است تا در مورد برخی از مفاهیم استفاده شده در پروژه که در ارتباط با یادگیری ماشینی هستند، توضیحاتی آورده شود. از آنجاییکه هر کدام از موارد ذکر شده، دارای ویژگی‌های پیشرفته و جزیی بسیاری هستند، صرفاً به معرفی ویژگی‌های کلی آن‌ها در راستای پروژه پرداخته خواهد شد. حوزه تحقیقاتی یادگیری ماشین زیر مجموعه هوش مصنوعی است که با استفاده از الگوریتم و مدل‌های آماری بجای دستورالعمل‌های واضح و برنامه‌ریزی آشکار، سعی در تشخیص و استنباط الگوها دارد. بخش قابل توجهی از یادگیری ماشین، بر یادگیری آماری استوار است. هدف اصلی در یادگیری آماری، پیدا کردن تابعی جهت پیش‌بینی و استنباط بر اساس داده‌ها است. تئوری یادگیری آماری، منجر به کاربردهای بسیار موفقی در زمینه‌های بینایی ماشین، پردازش زبان طبیعی، زیست‌داده‌ورزی، و پردازش داده‌های سری زمانی شده است. این موفقیت‌ها بدلیل توسعه الگوریتم‌های بهینه، حجم عظیم داده‌ها، و همچنین افزایش توان محاسباتی سامانه‌های رایانشی بوده است.

برای تعریف تابع مورد پیش‌بینی، ۲ دسته متغیر را در نظر می‌گیریم. ورودی‌های تابع که به آن‌ها عباراتی از جمله پیش‌بینی کننده، متغیرهای مستقل، یا ویژگی اطلاق می‌گردد (X) و دسته بعدی پاسخ تابع یا متغیر وابسته که بطور معمول با نماد Y مشخص می‌شود که در ارتباط بین این دو متغیر در رابطه ۱-۲ نشان داده شده است. (ε) عبارت خطأ است که مستقل از ویژگی‌ها بوده و دارای میانگین صفر می‌باشد.

$$Y = f(X) + \epsilon. \quad (1-2)$$

یکی از تقسیم‌بندی‌های متداول در یادگیری ماشینی، دسته‌بندی براساس نوع داده‌هایی است که در اختیار کارگزار هوشمند قرار می‌گیرد.

### ۱-۳-۲ یادگیری با نظارت

زمانی که برای هر مشاهده از ویژگی‌ها، پاسخ مرتبط با آن مشاهده وجود داشته باشد، مسئله یادگیری با نظارت<sup>۹</sup> است. در این مسئله می‌خواهیم مدلی مناسب با داده‌ها بدست آوریم که بتواند پاسخ‌ها را به ویژگی‌ها با هدف پیش‌بینی دقیق پاسخ برای مشاهدات آینده (مفهوم پیش‌بینی) و همچنین درک ساختار و روابط بین پاسخ و ویژگی (مفهوم استنباط)، مرتبط نماید. بسیاری از روش‌های یادگیری آماری قدیمی، مانند رگرسیون خطی<sup>۱۰</sup>، رگرسیون لجستیک<sup>۱۱</sup>، و روش‌های مدرن، مانند ماشین بردار پشتیبان<sup>۱۲</sup> در این دسته قرار می‌گیرند.

### ۲-۳-۲ یادگیری بدون نظارت

در مسئله یادگیری بدون نظارت<sup>۱۳</sup> برخلاف حالت قبل، پاسخی مرتبط با ویژگی‌های مشاهده شده وجود ندارد. عبارت بدون نظارت بدان معناست که مسئله پاسخ را نمی‌داند که قادر باشد یادگیری را نظارت کند. در واقع، هدف یافتن ارتباط و ساختارهای مشابه بین خود مشاهده‌ها است. یک ابزار یادگیری آماری در این خصوص، تحلیل خوشه می‌باشد که مشاهده‌های مشابه را در یک گروه و دسته قرار می‌دهد. از این روش در تحلیل داده اکتشافی استفاده می‌شود.

<sup>9</sup>Supervised Learning

<sup>10</sup>Linear Regression

<sup>11</sup>Logistic Regression

<sup>12</sup>Support Vector Machine

<sup>13</sup>Unsupervised Learning

### ۳-۳-۲ یادگیری تقویتی

نوع سوم از یادگیری ماشین که با دو نوع قبلی متفاوت است، یادگیری تقویتی<sup>۱۴</sup> نام دارد که هدف آن حل مسئله تصمیم‌گیری است. در این روش، عامل هوشمند با محیط پیرامون تعامل داشته و اگر طبق اهدافی که برایش معین شده است، تصمیم‌گیری کند و عمل انجام دهد پاداش دریافت می‌کند. درواقع، عامل یاد می‌گیرد که انجام چه اعمالی در چه شرایطی منجر به دریافت پاداش بیشتر خواهد شد. تفاوت این روش با دیگر روش‌های یادگیری ماشین، از آنجاییست که نظارتی در عمل یادگیری وجود نداشته و همچنین داده مشاهده شده، بستگی به عملی که عامل انجام می‌دهد دارد.

### ۴-۳-۲ مسئله دسته‌بندی

در یادگیری ماشینی و آمار، دسته‌بندی<sup>۱۵</sup> مسئله تعیین دسته یک مشاهده جدید بر اساس یک مجموعه از داده‌های مورد استفاده به منظور آموزش شامل مشاهدات است که عضویت در دسته‌هایشان معلوم است. مسئله دسته‌بندی از نوع یادگیری با نظارت است، که مجموعه‌ای از داده‌ها برای آموزش موجود هستند. الگوریتم‌های متعددی برای حل این مسئله ارائه شده است از جمله درخت تصمیم<sup>۱۶</sup>، شبکه عصبی مصنوعی<sup>۱۷</sup>، نزدیک‌ترین همسایه<sup>۱۸</sup> و ... . در مسئله تشخیص رخداد، داده حسگرهای مختلف بعنوان ویژگی‌ها و رخدادهای مدنظر نیز بعنوان پاسخ موجود هستند و می‌توانیم از روش دسته‌بندی برای پیش‌بینی داده‌هایی که در آینده وارد سامانه می‌شوند استفاده کنیم.

در ادامه به توضیح یکی از روش‌های حل مسئله دسته‌بندی آماری، شبکه‌های عصبی مصنوعی، پرداخته خواهد شد که مبنای روش اتخاذ شده برای دسته‌بندی رخدادهای رانندگی است.

<sup>14</sup>Reinforcement Learning

<sup>15</sup>Classification

<sup>16</sup>Decision Tree

<sup>17</sup>Artificial Neural Network

<sup>18</sup>Nearest Neighbor

## ۵-۳-۲ شبکه‌های عصبی مصنوعی

شبکه‌های عصبی مصنوعی سامانه‌های پردازشی هستند که از شیوه کارکرد سامانه عصبی زیستی مغز حیوانات الهام گرفته شده‌اند. این سامانه‌ها با درنظر گرفتن مشاهدات و مثال‌ها، می‌آموزند که وظایف مختلف را انجام دهنند بدون آنکه نیاز به برنامه‌ریزی قوانین از پیش مشخص شده داشته باشند. یک شبکه بر اساس مجموعه‌ای از واحدهای متصل به یکدیگر بنام نورون‌های<sup>۱۹</sup> مصنوعی ساخته می‌شود. این نورون‌ها، مدل بسیار ساده‌شده نورون‌های فعال در مغز هستند و هر ارتباط، مانند سیناپس‌های<sup>۲۰</sup> مغز زیستی، قادر به ارسال اطلاعات به سایر نورون‌ها می‌باشد. یکی از روش‌های حل مسئله دسته‌بندی داده‌ها، استفاده از شبکه‌های عصبی مصنوعی است.

هر نورون سیگنال‌های ورودی را از دندریت‌های خود دریافت کرده و سیگنال‌های خروجی را در امتداد آکسون تولید می‌کند. آکسون از طریق سیناپس به شاخه‌های دیگر سلول‌های عصبی متصل می‌گردد. در مدل محاسباتی نورون، سیگنال‌هایی که از طریق آکسون منتقل می‌شوند، بصورت حاصل‌ضرب با دندریت‌های نورون دیگر در تعامل هستند. این اتصال‌ها قابل یادگیری بوده و مقادیر انتقال یافته به سلول‌های دیگر کنترل می‌شود. در مدل پایه، دندریت‌ها سیگنال‌ها را به بدنه سلول منتقل کرده و در آنجا جمع می‌کنند. اگر میزان نهایی حاصل جمع، بیشتر از یک حد آستانه باشد، نورون فعال شده و می‌تواند اطلاعات را منتقل کند. نرخ فعال‌سازی نورون نیز توسط تابع فعال‌سازی مدل می‌شود. شما یک نورون زیستی و مدل ریاضی ساده‌شده آن در تصاویر<sup>۱۰-۲</sup> و<sup>۱۱-۲</sup> مشخص شده‌اند.

### معماری‌های شبکه عصبی

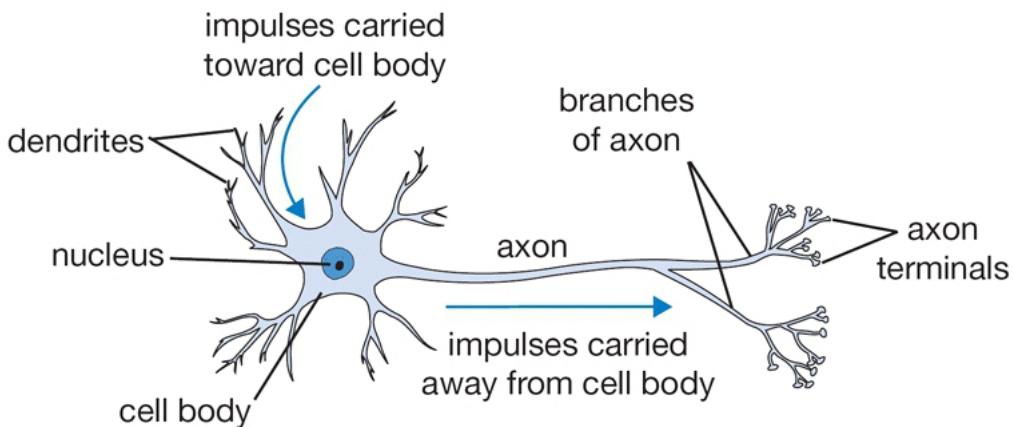
در این قسمت به ۲ نوع معماری از شبکه‌های عصبی اشاره خواهد شد.

### شبکه عصبی پیشخور

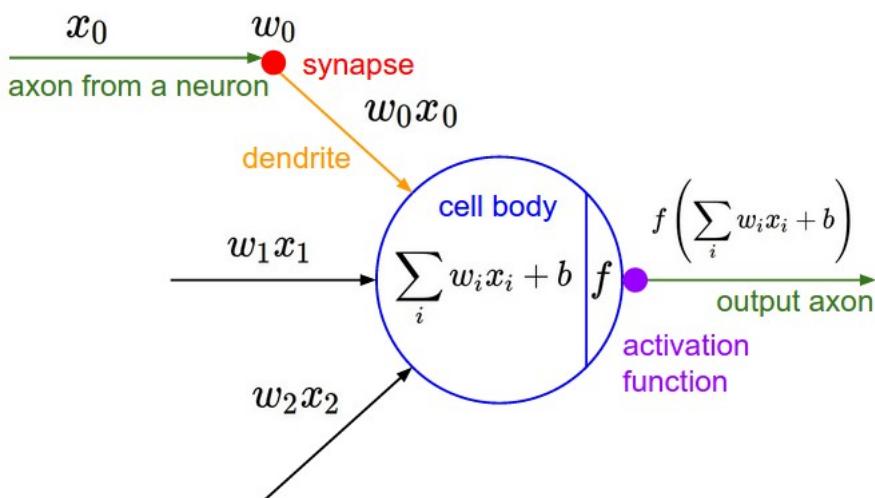
اولین و ساده‌ترین ساختار شبکه‌های عصبی را می‌توان با مجموعه‌ای از نورون‌های به هم مرتبط، تحت یک گراف بدون دور نشان داد. در این ساختار، خروجی برخی از نورون‌ها بعنوان ورودی نورون‌های دیگر استفاده می‌شود. در واقع اطلاعات با شروع از گره (نورون)‌های ورودی و گذر از لایه‌های پنهان (در صورت وجود) به سمت گره‌های خروجی می‌روند. از این معماری برای تخمین

<sup>19</sup>Neurons

<sup>20</sup>Synapse



شکل ۲: طرح یک نورون زیستی [۵]

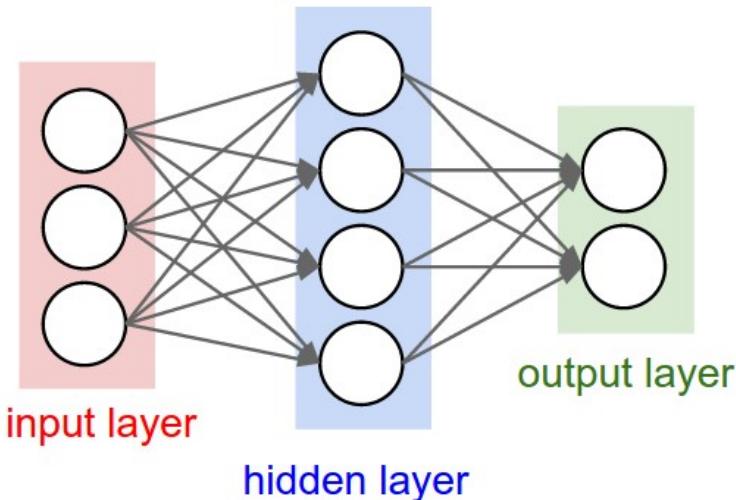


شکل ۳: مدل محاسباتی ساده شده نورون [۵]

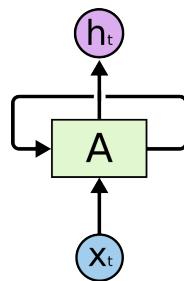
توابع خطی و غیرخطی مختلفی استفاده می‌شود، اما برای مسئله تشخیص رخداد از داده‌های حرکتی این مدل از شبکه عصبی کارا نخواهد بود. در تعریف شبکه‌های عصبی بازگشتی، ضعف این مدل را بیان نموده و راه حلی که شبکه‌های عصبی بازگشتی ارائه می‌دهند بررسی خواهد شد. در تصویر ۱۲-۲ مثالی در ارتباط با ساختار شبکه عصبی در مدل پیشخور آورده شده است.

### شبکه عصبی بازگشتی

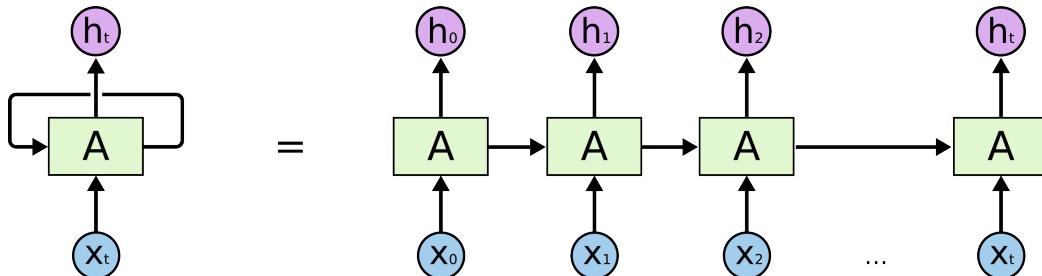
همانطور که این متن خوانده می‌شود، خواننده هر کلمه را بر اساس درکی که از کلمه قبل بوجود می‌آید متوجه خواهد شد [۱۲]. او پس از مواجهه شدن با هر کلمه، تمام افکار قبلی را دور نمی‌ریزد تا از ابتدا شروع به دریافتمن کلمه جدید کند. در واقع، افکار ما قابلیت ماندگاری دارند. شبکه‌های



شکل ۱۲-۲: شکل ساختار شبکه عصبی کاملاً متصل در مدل پیشخور [۵]



شکل ۱۳-۲: ساختار فشرده بخشی از یک شبکه عصبی بازگشتی و وجود حلقه در آن ساختار



شکل ۱۴-۲: ساختار بازشده حلقه بخشی از شبکه عصبی بازگشتی

عصبی پیشخور، قادر به مدل‌سازی این قابلیت انسان نیستند. اما شبکه‌های عصبی بازگشتی این مشکل را برطرف کرده‌اند. همانطور که در شکل ۱۳-۲ مشاهده می‌شود، بخشی از یک شبکه عصبی که با  $A$  مشخص شده است، با دیدن ورودی  $x_t$  خروجی  $h_t$  را تولید می‌کند و توسط حلقه‌ای که دارد اجازه می‌دهد اطلاعات از هر گام شبکه به گام بعدی منتقل گردد. با باز کردن این حلقه به شکل ۱۴-۲ خواهیم رسید.

چندین شبکه مشابه که هر کدام پیامی را به شبکه بعدی انتقال می‌دهند، ماهیت زنجیره‌ای

این ساختار نشان می‌دهد. این دسته از شبکه‌ها ارتباط نزدیکی با دنباله و لیست‌ها دارند. اعمال آن‌ها بر مسائل مرتبط مختلفی موققیت‌های چشم‌گیری را در چند سال گذشته به ارمغان آورده است. کاربردهایی از جمله: تشخیص گفتار، مدل‌سازی زبان، ترجمه و تولید متن، و ... .

یکی از نکات مثبت در طراحی شبکه‌های عصبی بازگشتی قابلیت ارتباط دادن اطلاعات گذشته به زمان حاضر است، مانند استفاده از فریم‌های گذشته یک فیلم جهت مطلع شدن از فریم حال. اگر این فاصله زمانی بین اطلاعات مرتبط و لحظه‌ای که اطلاعات مورد نیاز است زیاد باشد، شبکه‌های عصبی بازگشتی استاندارد در عمل عملکرد مطلوبی نخواهد داشت. به همین دلیل برای حل مشکل وابستگی طولانی مدت<sup>۲۱</sup>، معماری دیگری به نام حافظه کوتاه مدت طولانی<sup>۲۲</sup> یا به اختصار ال‌اس‌تی‌ام پدید آمد که در بسیاری از موققیت‌های شبکه‌های بازگشتی نقش ویژه‌ای را ایفا نموده است.

## ۶-۳-۲ شبکه LSTM

نوع خاصی از شبکه‌های عصبی بازگشتی است که قادر به یادگیری وابستگی‌های طولانی مدت در داده می‌باشد. [۱۶]. در این نوشه به معرفی بخشی از ویژگی‌های آن پرداخته خواهد شد. در معماری نشان‌داده شده در تصویر<sup>۲۳</sup>، متغیر وضعیت سلول<sup>۲۴</sup> در کنار<sup>۲۵</sup> ۳ تنظیم‌کننده به ازای هر واحد LSTM وجود دارد. به تنظیم‌کننده‌ها در واژه<sup>۲۶</sup> گفته می‌شود که در معماری استاندارد در واژه ورودی، خروجی، و فراموشی<sup>۲۷</sup> مشاهده خواهد شد. بطور شهودی، وضعیت سلول مسئول حفظ وابستگی بین المان‌های دنباله ورودی است. در واژه ورودی، مقادیر ورودی به سلول و در واژه خروجی، خروجی سلول را کنترل می‌کند. در واژه فراموشی در مورد باقی‌ماندن مقدار مشخص شده در سلول تصمیم‌گیری می‌کند. در واژه‌ها شامل تابع فعال‌سازی<sup>۲۸</sup> سیگموئید<sup>۲۹</sup> هستند که مقادیر را به بازه ۰ و ۱ ترجمه می‌کند. بواسطه این امر بروزرسانی و فراموشی مقادیر در وضعیت سلول تسهیل می‌گردد.

<sup>21</sup>Long-term dependency

<sup>22</sup>Long short-term memory

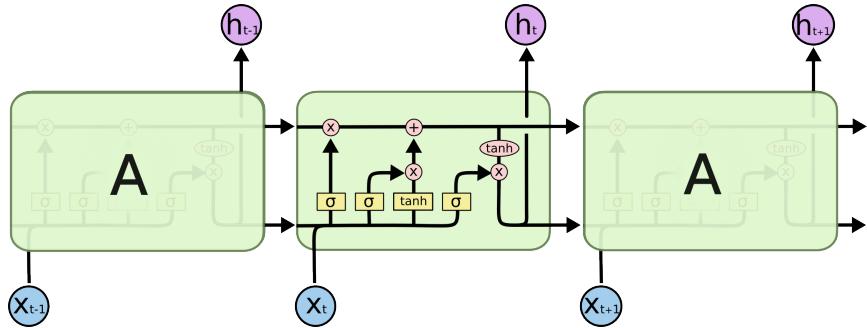
<sup>23</sup>Cell State

<sup>24</sup>Gate

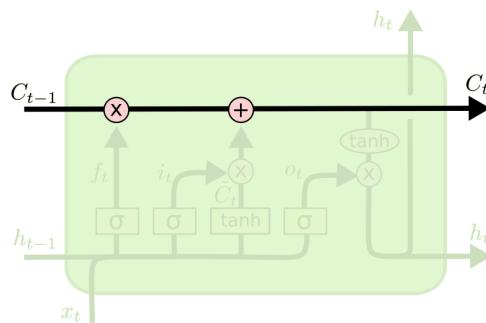
<sup>25</sup>Forget Gate

<sup>26</sup>Activation Function

<sup>27</sup>Sigmoid



شکل ۱۵-۲: معماری داخلی یک سلول LSTM



شکل ۱۶-۲: وضعیت سلول در یک سلول LSTM

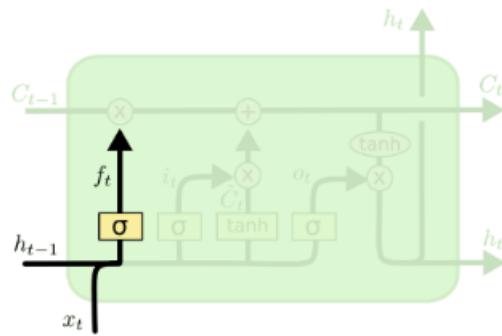
## وضعیت سلول

تصویر ۱۶-۲ نشان‌دهنده وضعیت سلول در معماری LSTM می‌باشد. با توجه به مکانیزم ارائه شده و وجود دروازه‌ها، جریان یافتن اطلاعات در طول شبکه همچنین اضافه و یا پاک نمودن آن از وضعیت سلول به راحتی مقدور می‌باشد.

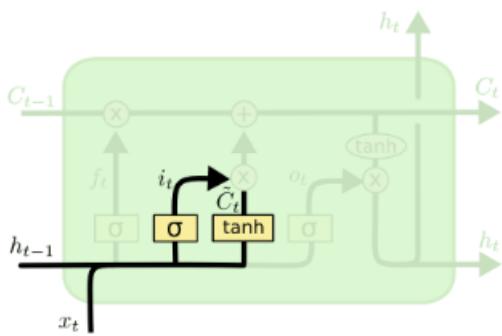
## دروازه فراموشی

تصمیم‌گیری در ارتباط با حذف اطلاعات موجود اولین قدم در معماری این نوع شبکه است. دروازه فراموشی با درنظرگرفتن  $h_{t-1}$  و  $x_t$  خروجی بین ۰ و ۱ را به ازای هر عدد در وضعیت سلول تولید می‌کند. ۱ نشان‌دهنده کامل نگهداشتن اطلاعات و ۰ به معنای فراموشی کامل آن است. (تصویر ۱۷-۲). معادله ۱۷-۲ نشان‌دهنده محاسباتی است که در این دروازه انجام می‌شود.  $f_t$  بردار مربوط به دروازه خروجی در زمان  $t$  است.  $W_f$  و  $b_f$  به ترتیب نشان‌دهنده پارامترهای وزن ورودی و وزن بایاس این دروازه می‌باشند.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (۱۷-۲)$$



شکل ۱۷-۲: دروازه فراموشی یک سلول LSTM



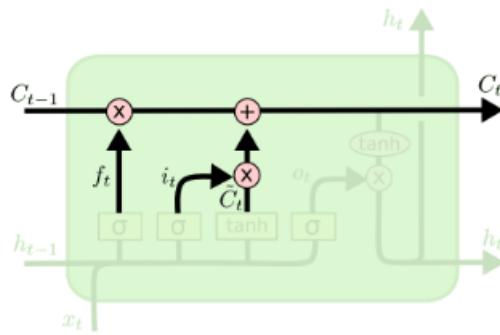
شکل ۱۸-۲: دروازه ورودی یک سلول LSTM

### دروازه ورودی

گام بعد در ارتباط با ذخیره اطلاعات جدید در وضعیت سلول است. این مرحله از دو قسمت تشکیل شده، ابتدا لایه سیگموئید که همان دروازه ورودی است داده‌ها را انتخاب می‌کند که در رابطه ۳-۲ با نماد  $i_t$  مشخص شده است. سپس، لایه تانژانت هذلولی مقادیر کандید را ( $\tilde{C}_t$ ) جهت اضافه نمودن به متغیر وضعیت سلول مشخص می‌نماید. در رابطه ۳-۲ محاسبات انجام شده در این دروازه آورده شده است. در این معادله همانند رابطه ۲-۲ از وزن‌های مشخص شده در دروازه ورودی استفاده می‌شود. تصویر ۱۸-۲ نشان‌دهنده عملکرد این دروازه در یک سلول LSTM می‌باشد.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (3-2)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$



شکل ۲-۱۹: بروزرسانی وضعیت سلول یک سلول LSTM

### بروزرسانی وضعیت سلول

در این مرحله نیاز است تا داده‌های قدیم وضعیت سلول ( $C_{t-1}$ ) بروزرسانی شوند ( $C_t$ ). وضعیت سلول قدیم در مقدار دروازه فراموشی ضرب شده و با ضریبی از مقدار وضعیت سلول فعلی جمع می‌گردد که در رابطه ۴-۲ مشخص شده است. تصویر ۱۹-۲ نیز عملکرد این بروزرسانی را نشان می‌دهد.

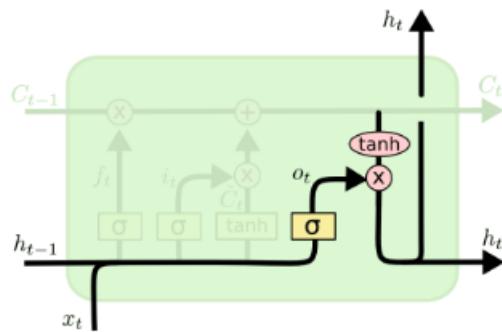
$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (4-2)$$

### دوازه خروجی

در انتهای، خروجی این بلاک بر اساس وضعیت سلول مشخص می‌گردد. با استفاده از لایه‌های سیگموئید و تانژانت هذلولی، بخش‌هایی از وضعیت سلول بعنوان خروجی درنظر گرفته خواهند شد، تصویر ۲۰-۲. نحوه محاسبه خروجی در رابطه ۵-۲ مشخص شده است. متغیرهای بکار رفته در این معادلات همانند دروازه ورودی است و  $h_t$  خروجی بلاک این سلول است.

$$\begin{aligned} o_t &= \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \\ h_t &= o_t * \tanh(C_t) \end{aligned} \quad (5-2)$$

اکثر موفقیت‌های بدست آمده در ارتباط با شبکه‌های عصبی بازگشتی بر روی داده‌های دنباله ای و سری زمانی، توسط معماری LSTM بوده است. برای دسته‌بندی داده‌های حرکتی خودرو در



شکل ۲۰-۲: دروازه خروجی یک سلول LSTM

این پروژه نیز از این معماری استفاده شده است که جزئیات پیاده‌سازی آن در بخش پیاده‌سازی توضیح داده خواهد شد.

## ۴-۲ کتابخانه‌های نرم‌افزاری

جهت توسعه و ساخت مدل‌های مبتنی بر شبکه‌های عصبی، کتابخانه‌های نرم‌افزاری متعددی توسط شرکت‌های نرم‌افزاری معتبر مانند گوگل، فیسبوک، و ... توسعه داده شده‌اند تا کاربران و برنامه‌نویسان بتوانند به مجموعه قابل قبولی از توابع و قابلیت‌های پردازشی مورد نیاز در این زمینه دسترسی داشته باشند. در این قسمت به بررسی مختصر تعدادی از شناخته شده‌ترین نرم‌افزارهای حوزه یادگیری ماشین خواهیم پرداخت که پس از مقایسه ویژگی‌ها تعدادی از آن‌ها در پروژه استفاده شده است.

### TensorFlow ۱-۴-۲

[۱۱] TensorFlow کتابخانه نرم‌افزاری متن‌بازی<sup>۲۸</sup> جهت طراحی، ساخت، و آموزش مدل‌های یادگیری ماشینی است. این کتابخانه در آغاز توسط تیم گوگل‌برین<sup>۲۹</sup> شرکت گوگل توسعه و بصورت داخلی استفاده می‌شده است که بعدتر در سال ۲۰۱۵ تحت مجوز آپاچی<sup>۳۰</sup> منتشر شد. در تعریف سطح بالا، کتابخانه‌ای به زبان پایتون است که به کاربران اجازه می‌دهد تا محاسبات‌شان را بصورت گراف جریان داده تعریف نمایند. گره‌های این گراف، مشخص‌کننده عملیات ریاضی و

<sup>28</sup>open source

<sup>29</sup>Google Brain

<sup>30</sup>Apache License

یال‌ها نشان‌دهنده داده‌هایی هستند که از یک گره به گره دیگر منتقل می‌شود. داده در تنسورفلو بصورت تنسور<sup>۳۱</sup> بیان می‌شود، آرایه یا ماتریس<sup>n</sup> بعدی که می‌تواند نماینده هر نوع داده‌ای باشند.

### Keras ۲-۴-۲

کتابخانه نرم‌افزاری متن‌باز شبکه‌های عصبی است که توسط زبان پایتون توسعه یافته است و می‌تواند بر بستر تنسورفلو اجرا شود<sup>[۷]</sup>. کراس رابط برنامه‌نویسی کاربردی<sup>۳۲</sup> به همراه مجموعه‌ای از دستورات سطح بالا و انتزاعی را در اختیار کاربران قرار می‌دهد تا بتوانند صرفاً بر توسعه مدل‌های یادگیری عمیق<sup>۳۳</sup> و شبکه‌های عصبی متمرکز بشوند.

### PyTorch ۳-۴-۲

کتابخانه متن‌بازی که توسط شرکت Facebook تحت زبان پایتون توسعه یافته است<sup>[۸]</sup>. از قابلیت‌های سطح بالای آن می‌توان به محاسبات بر مبنای تنسور و ابزارهای توسعه شبکه‌های عصبی عمیق اشاره نمود. پشتیبانی از واحدهای پردازشی گرافیکی در زمان اجرای محاسبات قابلیتی است که سرعت و کارایی مناسبی را برای پیاده‌سازی مدل‌های مختلف یادگیری ماشین فراهم می‌نماید.

### Caffe ۴-۴-۲

کتابخانه متن‌باز برای توسعه مدل‌های یادگیری عمیق است که پروژه‌های متعددی در حوزه‌های بینایی ماشین و تشخیص گفتار توسط آن توسعه داده شده‌اند<sup>[۹]</sup>. سرعت پردازشی و کارایی بالا جزو مزیت‌های این کتابخانه هستند. توسعه این کتابخانه از دانشگاه برکلی آمریکا آغاز شد.

### scikit-learn ۵-۴-۲

این کتابخانه نرم‌افزاری زبان برنامه‌نویسی پایتون از دیگر نرم‌افزارهای یادگیری ماشین ذکر شده قدیمی‌تر است<sup>[۹]</sup>. این کتابخانه ابزارها و الگوریتم‌های مختلفی را برای انجام عملیات مورد نیاز

<sup>31</sup>Tensor

<sup>32</sup>Application Programming Interface

<sup>33</sup>Deep Learning

در حوزه یادگیری ماشین و آمار مانند دسته‌بندی، رگرسیون، خوشه‌بندی و همچنین عملیات پایه آماری را ارائه می‌دهد. scikit-learn مستندات بسیار کاملی را در اختیار توسعه دهنده‌گان قرار می‌دهد و از این نظر جزو بهترین کتابخانه‌های بشمار می‌آید.

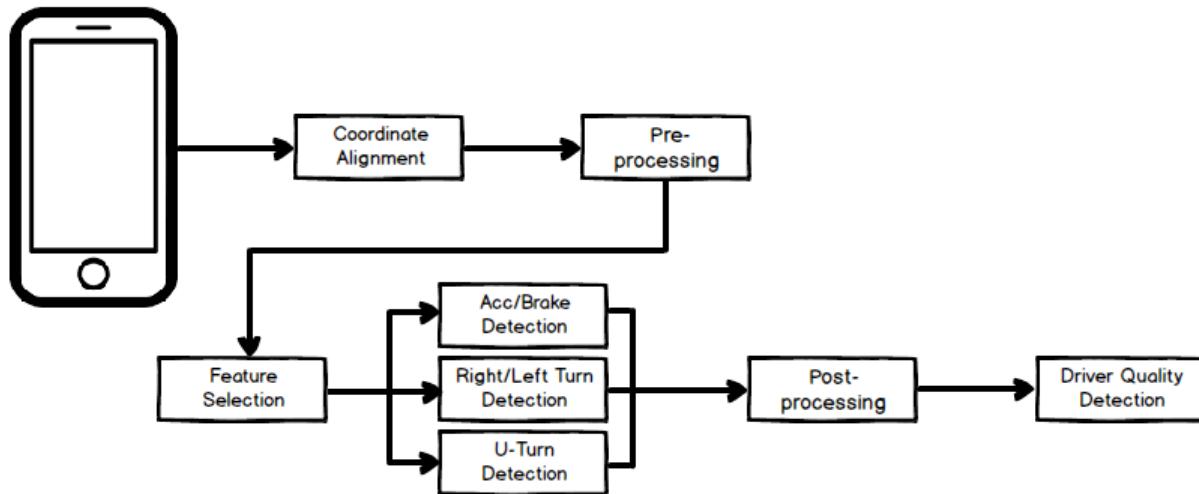
جهت توسعه مدل‌های مبتنی بر شبکه‌های عصبی در این پژوهه، پس از بررسی کتابخانه‌های اشاره شده و چارچوب‌های نرم‌افزاری مختلف در ارتباط با توسعه مدل‌های یادگیری ماشین، کتابخانه نرم‌افزاری Keras و زبان برنامه نویسی پایتون انتخاب شده‌اند. فراهم نمودن دستورات سطح بالا، قابلیت تغییر و اصلاح آسان بلاک‌های شبکه عصبی بواسطه معماری پیمانه‌ای، مستندات برنامه‌نویسی جامع و کامل، و همچنین حضور انجمان‌های مختلفی از کاربران و برنامه‌نویسان از دلایل اصلی انتخاب تکنولوژی‌ها بوده است. برای مراحل پیش پردازش، پس پردازش داده‌ها و اجرای عملیات‌های مختلف بر روی داده‌های جمع‌آوری شده نیز از کتابخانه scikit-learn استفاده شده است.

## ۵-۲ جمع‌بندی

همانطور که در قسمت‌های مختلف این بخش به تفصیل توضیح داده شد، این پژوهه از تکنولوژی و نرم‌افزارهای مختلفی تشکیل شده است. برای استفاده از حسگر IMU قطعات و روش ساخت مدار الکتریکی جهت دریافت و جمع‌آوری داده‌های حرکتی شرح داده شد. نرم‌افزار SensorLog به منظور استفاده از حسگرهای اینرسی تعییه شده درون تلفن‌های همراه معرفی شد. همچنین، برای توسعه مدل‌های تشخیص رخداد رانندگی، مقدماتی از روش‌های یادگیری ماشین، شبکه‌های عصبی، و کتابخانه‌های نرم‌افزاری مرتبط با آن شرح داده شد.

# فصل سوم

## طراحی و پیاده‌سازی



شکل ۱-۳: معماری کلی پروژه

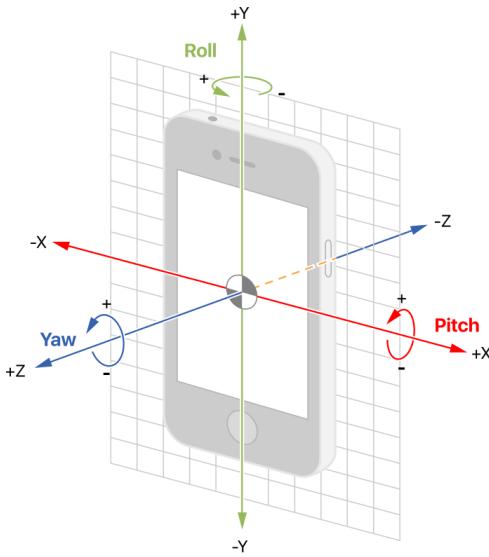
در این بخش، طراحی و پیاده‌سازی بخش‌های مختلف مدل پیشنهادی شرح داده خواهد شد.

### ۱-۳ مقدمه

شمای کلی پروژه در تصویر ۱-۳ آمده است. پس از ثبت داده‌های حرکتی در رانندگی، یک مرحله هموارسازی به علت کاهش تاثیر نویز بر روی داده‌ها انجام می‌گیرد. سپس داده‌ها از دستگاه مختصات تلفن همراه به دستگاه مختصات خودرو منتقل شده و پس از آماده سازی داده‌ها، ویژگی‌های مدنظر جهت پردازش انتخاب می‌گردند. بر اساس صوت ضبط شده هنگام ثبت داده، برچسب‌گذاری رخدادهای صورت گرفته انجام می‌شود و داده‌ها به همراه برچسب‌هایشان بعنوان ورودی به مدل‌های الگوریتم‌های دسته‌بندی جهت یادگیری داده می‌شوند. پس از یادگیری و انتخاب مدل بر اساس روش اعتبار سنجی متقابل<sup>۱</sup>، آن‌ها قادر به تشخیص رخدادها و همچنین نسبت دادن مشخصه‌هایی به آن‌ها می‌باشند. پس از بدست آوردن لیستی از رخدادهای صورت گرفته در طی یک دوره رانندگی، با توجه به توزیع شدت رخدادها در واحد زمان، نمره‌ای را بعنوان کیفیت رانندگی اختصاص داده می‌شود.

در قسمت‌های بعد به تفصیل، طراحی و پیاده‌سازی هر کدام از این عناصر توضیح داده خواهد شد.

<sup>۱</sup>Cross-validation method



شکل ۲-۳: دستگاه مختصات تلفن همراه

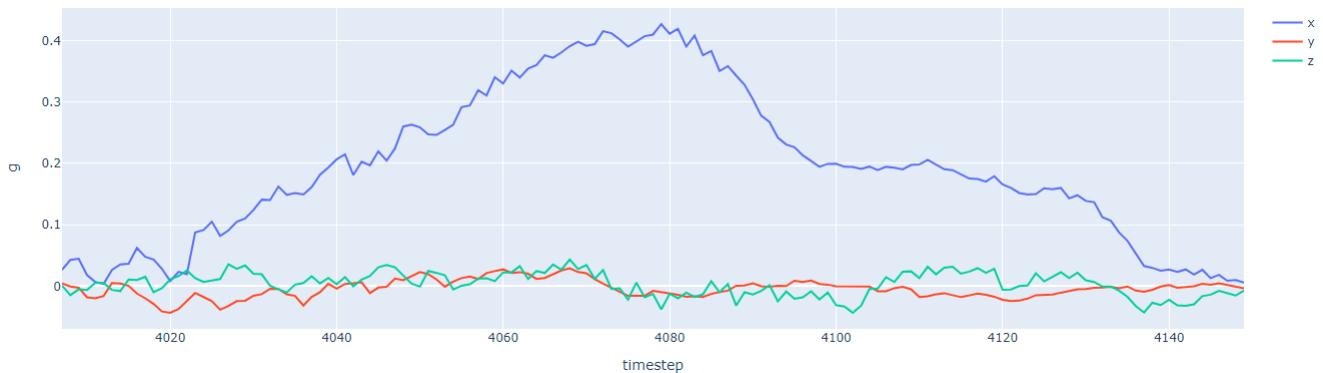
## ۲-۳ داده حسگرهای حرکتی

رفتارهای مختلف رانندگی به تغییرات داده‌های حرکتی در محورهای مختلف منجر خواهد شد. برای تشخیص رخداد توسط داده‌های اینرسی، نیاز به درک نحوه تغییرات هر کدام از داده‌ها می‌باشد. در این بخش تغییرات داده حسگرهای مختلف بررسی خواهد شد.

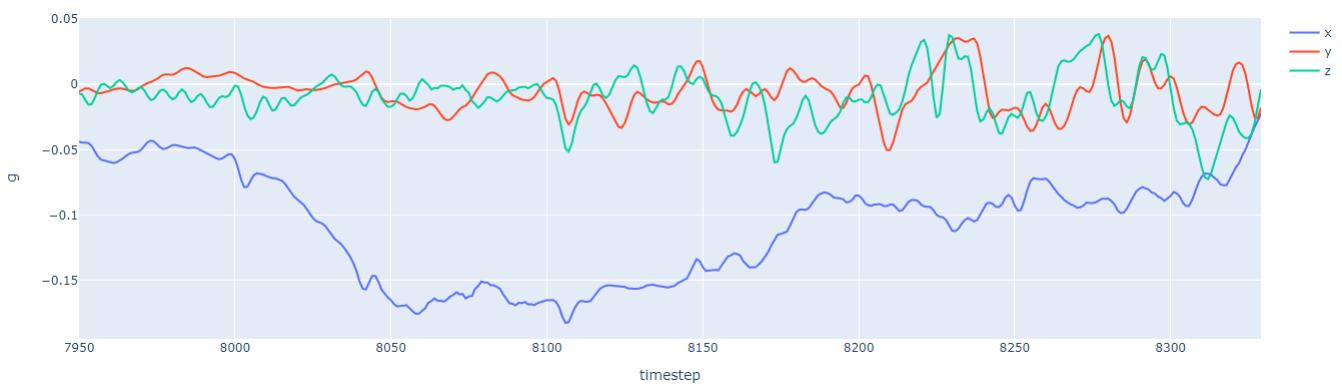
از آنجاییکه داده‌ها توسط حسگرهای تعییه شده درون تلفن همراه ثبت شده‌اند، نیاز است تا جهت محورهای مختلف حسگرهای تلفن همراه در نظر گرفته شود. در تصویر ۲-۳ دستگاه مختصات تلفن همراه مدنظر نشان داده شده است.

## ۱-۲-۳ رخدادهای ترمز و شتاب

هنگامی که خودرو به طور مستقیم در حال حرکت است، حرکات ترمز و شتاب گرفتن، ارتباط مستقیم با داده حسگر شتاب‌سنج دارند. درحالیکه در سرعت زاویه‌ای و چرخش تغییری مشاهده نخواهد شد. از آنجاییکه محور منفی  $x$  تلفن همراه در جهت حرکت مستقیم خودرو قرار دارد. با انجام عمل ترمز، شتابی خلاف جهت حرکت ایجاد می‌شود که مقداری مثبت خواهد داشت و در تصویر نشان ۳-۳ داده شده است. در قسمت اول نمودار که سیر صعودی شتاب در جهت  $x$  را نشان می‌دهد، سرعت در حال کاهش است و نیرویی خلاف جهت حرکت به خودرو وارد خواهد شد. این نیرو به مرور کاهش یافته تا به صفر برسد.



شکل ۳-۳: رخداد ترمز

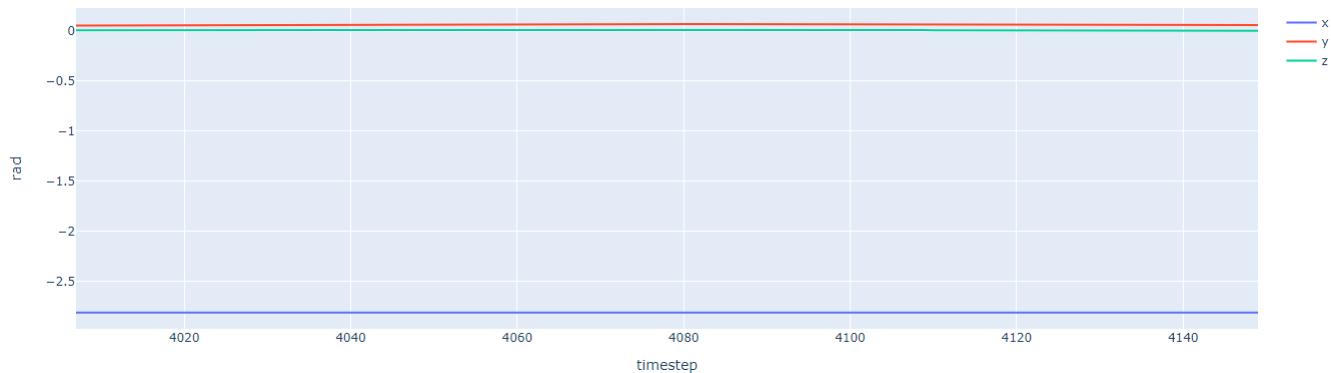


شکل ۳-۴: رخداد شتاب

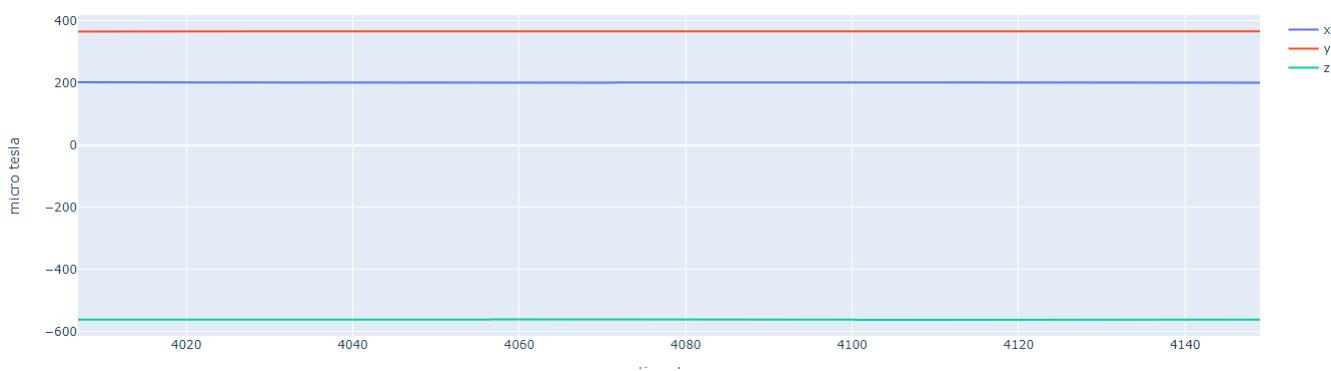
برخلاف ترمز، عمل شتاب گرفتن بدلیل ایجاد نیرو در راستای محور منفی  $x$ ، مقدار شتاب را از صفر به منفی تغییر داده و سپس مجدد به صفر باز می‌گردد. در این نمودار، ابتدا نیرویی در جهت موافق حرکت که باعث بالا رفتن سرعت می‌گردد به خودرو وارد شده و پس از مدتی، شتاب مجدداً به مقدار صفر باز می‌گردد.

در این دو رخداد انجام شده، شتاب دیگری به خودرو وارد نشده و از ثابت بودن داده‌های مربوط به چرخش، می‌توان دریافت که خودرو در مسیر مستقیم درحال حرکت بوده است. نمودار ۵-۳ نشان‌دهنده داده‌های حسگر چرخش‌نما در زمان انجام عمل ترمز می‌باشد که نشان‌دهنده مستقیم بودن حرکت است. نمودار مغناطیس‌سنج نیز که تغییر میدان مغناطیسی را ثبت می‌کند تغییری را در خصوص انحراف خودرو ثبت نکرده است، تصویر ۶-۳.

از آنجاییکه هم در مدل‌سازی فیزیکی تئوری و هم در عمل [۱۹] تنها داده‌های موثر در رخداد ترمز و شتاب گرفتن حسگر شتاب‌سنج است، برای آموزش مدل دسته‌بندی در خصوص این رخدادها، صرفاً از داده‌های حسگر شتاب‌سنج استفاده شده است.



شکل ۳-۵: داده حسگر چرخنما هنگام انجام رخداد ترمز



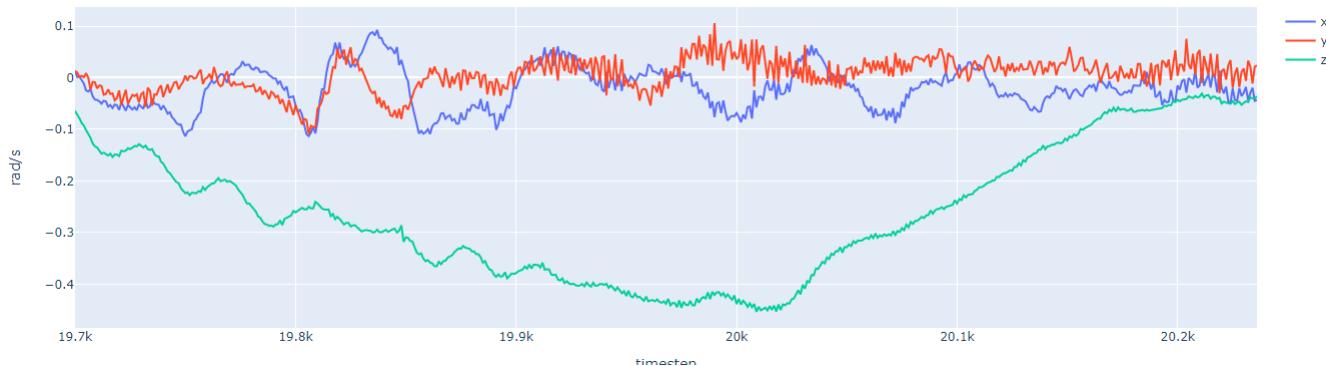
شکل ۳-۶: داده حسگر مغناطیس سنج هنگام رخداد ترمز

## ۲-۲-۳ رخدادهای چرخش

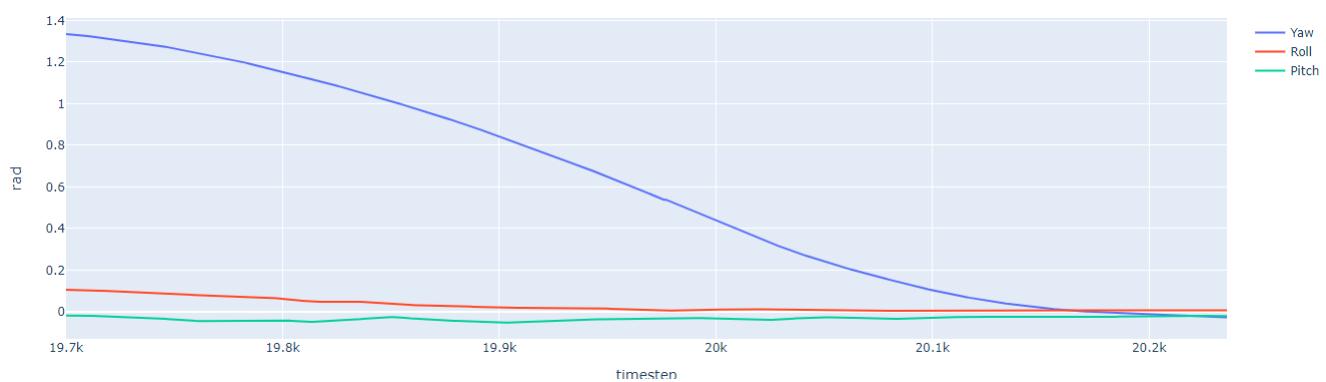
رخدادهای چرخش را می‌توان بطور کلی به ۳ دسته اصلی تقسیم نمود. چرخش به راست، چپ، و عمل دور زدن. هر کدام از این رخدادها با تغییر داده‌های چرخشی و مغناطیسی همراه هستند.

### چرخش به راست

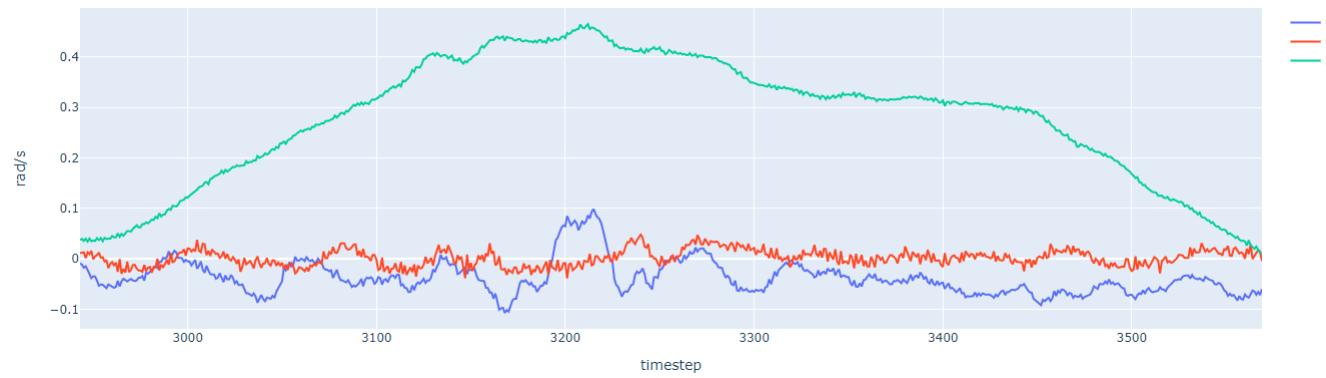
بدلیل قرارگیری تلفن همراه بصورت افقی در درون خودرو، چرخش به راست و چپ در واقع گردش به دور محور z تلفن همراه خواهد بود. در تصویر ۷-۳ داده حسگر چرخنما در ۳ محور آمده است که علامت آن جهت گردش و مقدار آن قدر مطلق سرعت زاویه‌ای را مشخص می‌نماید. همانطور که در تصویر ۸-۳ مشخص است، در حین چرخش به راست حدود ۱.۳ رادیان (74.47 درجه) چرخش ثبت شده است. از این داده می‌توان برای بررسی اعتبار رخداد چرخش استفاده نمود.



شکل ۳-۳: داده حسگر چرخش‌نما هنگام رخداد چرخش به راست



شکل ۳-۸: میزان چرخش هنگام رخداد چرخش به راست

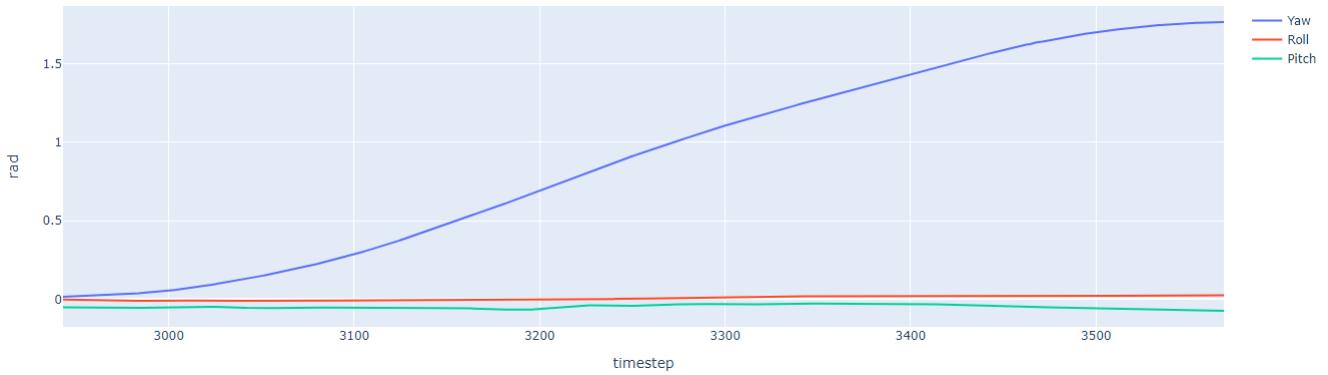


شکل ۳-۹: داده حسگر چرخش‌نما هنگام رخداد چرخش به چپ

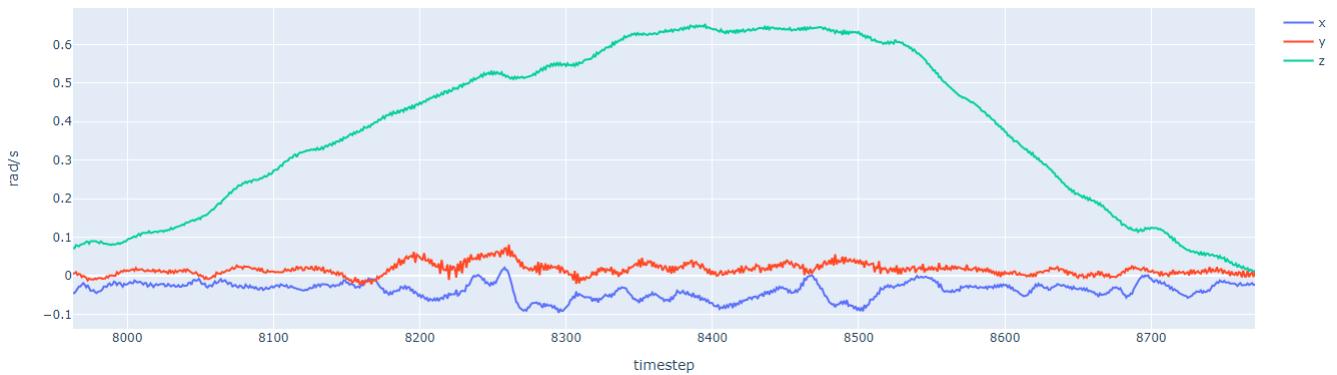
### چرخش به چپ

همانطور که در قسمت چرخش به راست مشاهده شد، داده چرخش‌نما محور z دچار تغییر شده است و بدلیل جهت چرخش مقدار آن صعودی و مثبت شده است. اندازه سرعت زاویه‌ای در این مثال تقریباً برابر با اندازه سرعت زاویه‌ای در مثال چرخش به راست می‌باشد.

با توجه به تصویر ۱۰-۳ که میزان چرخش بر حسب رادیان را نشان می‌دهد، چرخش به اندازه



شکل ۳-۱۰: میزان چرخش هنگام رخداد چرخش به چپ

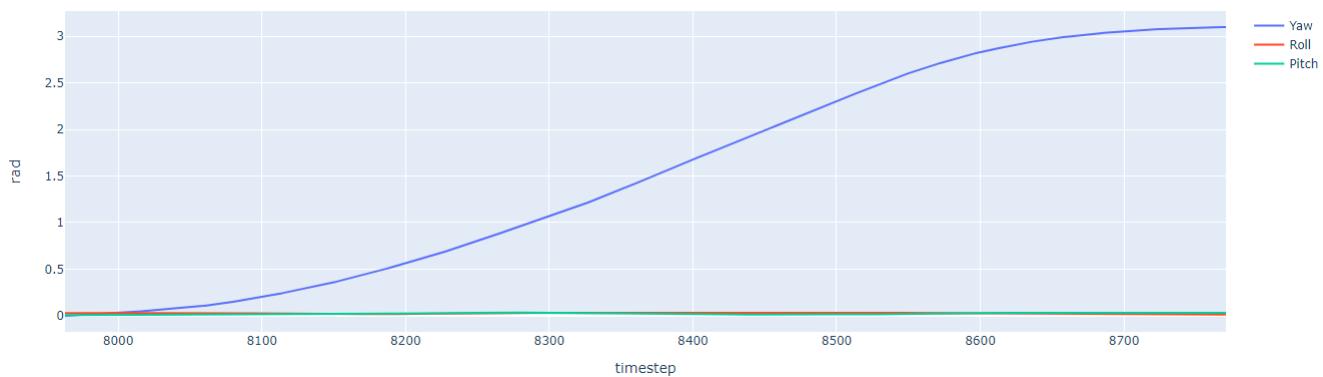


شکل ۳-۱۱: داده حسگر چرخش‌نما هنگام رخداد دور زدن

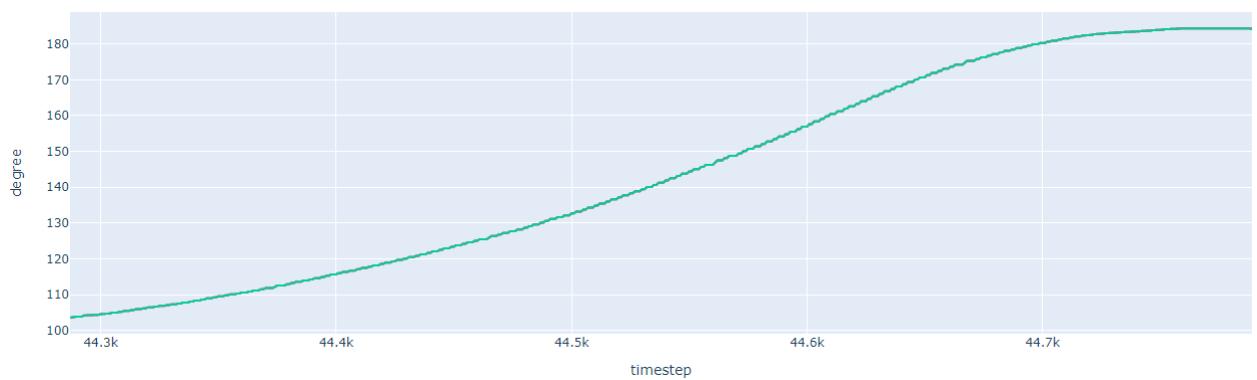
تقریبی ۱.۷ رادیان (۹۷.۴ درجه) می‌باشد.

## دور زدن

بخشی از عمل دور زدن با چرخش به سمت راست و چپ همپوشانی دارد، اما داده‌های میزان زاویه چرخش و جهت خودرو نسبت به قطب مغناطیسی شمال می‌توانند در تمایز دور زدن با سایر چرخش‌ها مفید باشند. در این قسمت به بررسی تغییرات این داده‌ها در عمل دور زدن پرداخته خواهد شد. همانطور که در تصویر ۳-۱۱ مشخص شده است، تغییر سرعت زاویه‌ای همانند تغییر این داده در هنگام عمل چرخش به چپ می‌باشد. بدلیل اینکه دور زدن در این مثال از سمت چپ انجام شده است. این داده اطلاعات بیشتری در اختیار ما نمی‌گذارد. میزان چرخش خودرو در عمل دور زدن تقریباً برابر با ۳.۱ رادیان (۱۷۷.۶ درجه) است که نشان‌دهنده تغییر مسیر به طرف خلاف جهت قبلی حرکت می‌باشد درحالیکه در چرخش‌ها این میزان به اندازه تقریبی ۹۰ درجه بوده است، تصویر ۳-۱۲.



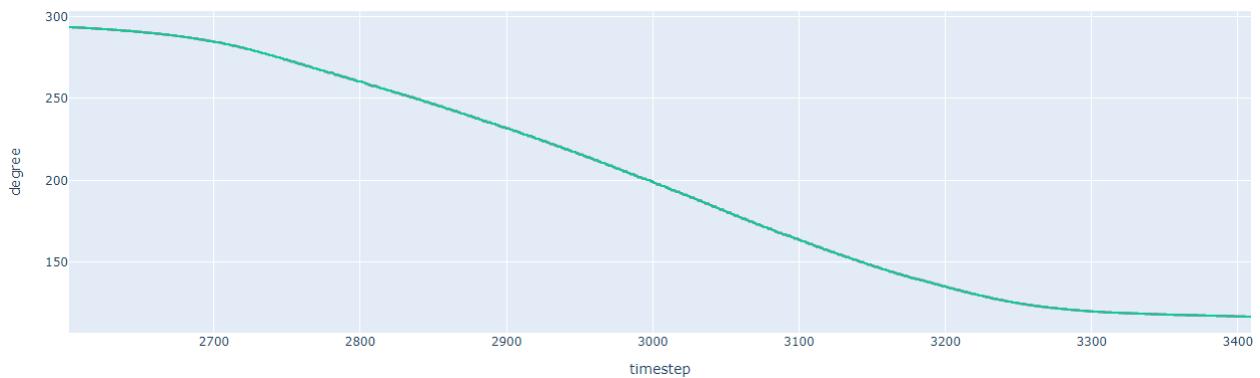
شکل ۱۲-۳: میزان چرخش هنگام رخداد دور زدن



شکل ۱۳-۳: جهت حرکت نسبت به قطب شمال مغناطیسی، چرخش به راست

جهت حرکت نسبت به قطب شمال مغناطیسی در دو حرکت دور زدن و چرخش به راست جهت مقایسه آورده شده است. در تصویر ۱۳-۳ که عمل چرخش به راست انجام شده است، اختلاف ۸۰ درجه‌ای بین زمان شروع و انتهای عمل نشان‌دهنده تغییر جهت نسبی که در چرخش صورت می‌گیرد است. اما در تصویر ۱۴-۳ که دور زدن انجام شده، این عدد به حدود ۱۸۰ درجه می‌رسد که نمایانگر دور زدن است.

با مشاهده رفتار داده‌ها در حین انجام حرکات مختلف خودرو درک مدل فیزیکی حرکت آن مشخص‌تر گردید. در نتیجه برای ساختن مدل دسته‌بندی هر کدام از رخدادها می‌توان داده حسگرهایی را انتخاب نمود که تاثیر مستقیم با حرکت مربوطه دارند. در این پژوهه، برای شناسایی ترمز و شتاب گرفتن، از حسگر شتاب‌سنج، و برای حرکات چرخش از حسگرهای شتاب‌سنج، چرخش‌نما، و مغناطیس‌سنج استفاده شده است.



شکل ۳-۱۴: جهت حرکت نسبت به قطب شمال مغناطیسی، دور زدن

### ۳-۳ جمع‌آوری داده

برای جمع‌آوری داده‌های حرکتی در رانندگی، مسیرها و جاده‌های مشخصی (در شهر ساری، مرکز استان مازندران) تعیین شده اند. سعی شده است تا در زمان‌های مختلف روز و در شرایط ترافیکی مختلفی داده‌ها ثبت گردند. همچنین تردد خودرو هم در خیابان و هم در بزرگراه انجام شده است. خودروی استفاده شده برای انجام آزمایش‌ها، پژو ۲۰۶ اسدی (سدان، صندوقدار) V8 بوده است. تلفن همراه استفاده شده، iPhone 6s و سیستم عامل آن iOS 12.4.5 می‌باشد. در این مدل از تلفن همراه شرکت اپل از ۲ حسگر شتاب‌سنج، به منظور کاهش توان مصرفی و بهبود تجربه کاربری، استفاده شده است. مدل ۳ محوره و سریع‌تر و مدل دیگر که ۶ محوره است. زمانی که نیاز به مدل ۶ محوره وجود نداشته باشد، از مدل ۳ محوره استفاده خواهد شد. موقعیت تلفن همراه استفاده شده در داخل خودرو بصورت افقی بر روی محفظه جلوی خودرو قرار گرفته است. سعی شده است تا تلفن موقعیت ثابتی داشته باشد تا از ارتعاشات ناخواسته جلوگیری شود. مشخصات مجموعه داده بدست آمده در جدول ۱-۳ آمده است. برای ثبت و ضبط این مجموعه داده حدود ۳۵.۵ کیلومتر رانندگی انجام شده است. بخشی از مسیرهای انتخابی در تصاویر ۱۵-۳، ۱۶-۳، ۱۷-۳، و ۱۸-۳ آورده شده است.

### ۱-۳-۳ هماهنگ‌سازی دستگاه مختصات

تا زمانی که دستگاه مختصات تلفن همراه با دستگاه مختصات ماشین همتراز نباشد، نمی‌توان اطلاعات معناداری از پردازش اطلاعات حسگرها بدست آورد [۱۸]. به این منظور نیاز داریم تا دادگان حسگرها را به دستگاه مختصات یکتاپی تبدیل کنیم. برای سهولت محاسبات، این دستگاه

نوع رخداد	تعداد نمونه جمع‌آوری شده
شتاب	۲۶
ترمز	۵۵
چرخش به راست	۵۰
چرخش به چپ	۲۷
دور زدن	۱۸
جمع رخدادها	۱۷۶

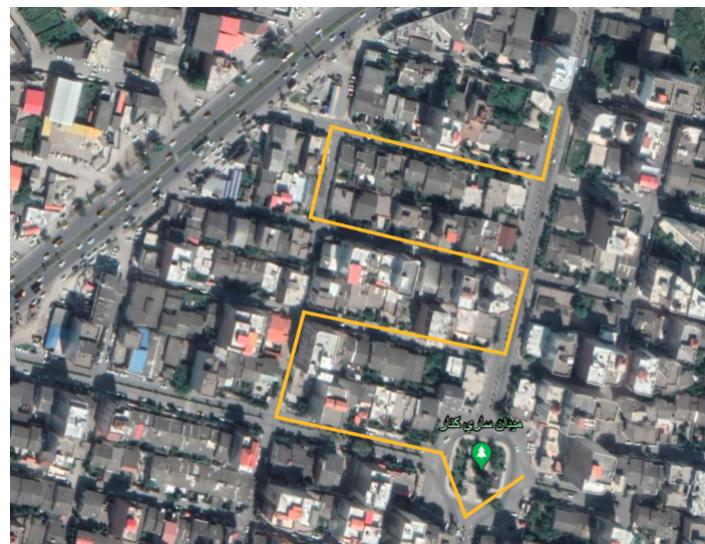
جدول ۳-۱: جدول مشخصات مجموعه داده بدست آمده

مختصات جدید در جهت حرکت مستقیم خودرو انتخاب شده است. تصویر ۱۸-۳ نشان‌دهنده دستگاه مختصات خودرو و تلفن همراه است.

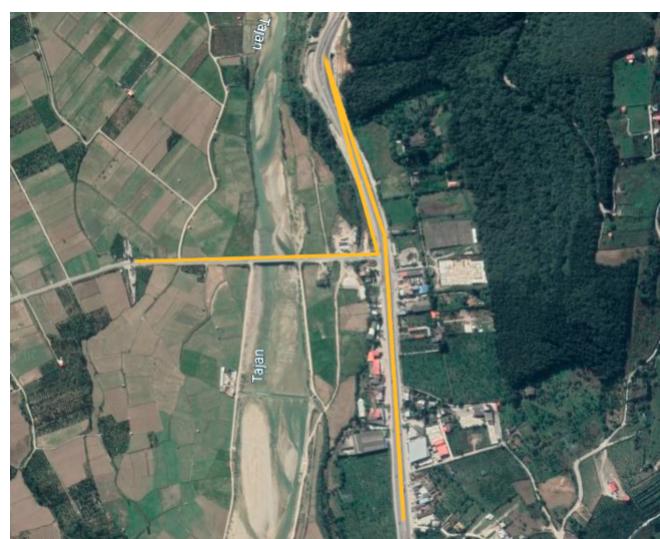


شکل ۳-۱۵: مسیر انتخابی ۱

هماهنگ سازی دستگاه مختصات توسط داده حسگرهای شتاب‌سنج و چرخ‌نمای صورت می‌پذیرد. همانطور که در تصویر ۱۸-۳ مشخص است، دستگاه مختصات تلفن توسط نحوه قرارگیری آن در خودرو تعیین می‌شود  $X_p, Y_p, Z_p$ . بدنبال ماتریس دوران  $R$  ای هستیم که با دوران دستگاه مختصات تلفن به دستگاه مختصات تعیین شده برای خودرو  $X_c, Y_c, Z_c$  دست یابد. در این خصوص ۳ بردار یکه  $\hat{i}, \hat{j}, \hat{k}$  در دستگاه مختصات خودرو درنظر گرفته و هر کدام را در دستگاه مختصات تلفن مطابق با رابطه ۱-۴ بازنویسی می‌کنیم.



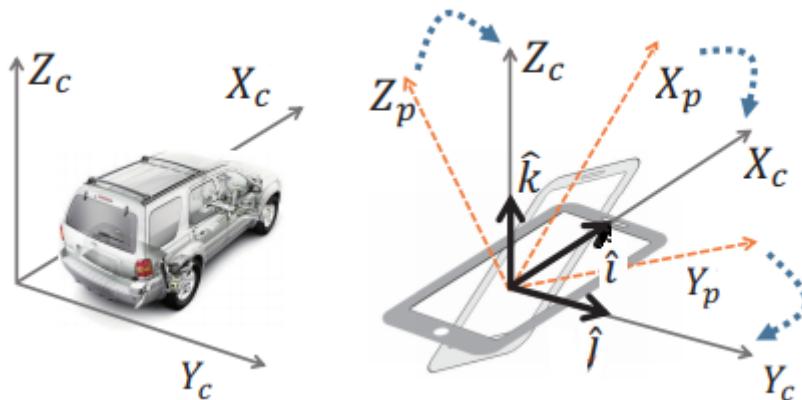
شکل ۳-۱۶: مسیر انتخابی ۲



شکل ۳-۱۷: مسیر انتخابی ۳

$$\hat{q} = [x_q, y_q, z_q]^T, q \in i, j, k \quad (1-3)$$

$$R = \begin{pmatrix} x_i & x_j & x_k \\ y_i & y_j & y_k \\ z_i & z_j & z_k \end{pmatrix} \quad (2-3)$$



شکل ۳-۱۸: دستگاه مختصات خودرو و تلفن همراه

برای بدست آوردن هر یک از المان‌های ماتریس دوران کافیست ۳ مرحله زیر را انجام دهیم.

**بدست آوردن  $\hat{k}$**

با اعمال فیلتر پایین‌گذر بر بازه‌ای از حسگر شتاب‌سنج می‌توان مولفه‌های ثابت شتاب را جهت محاسبه شتاب جاذبه بدست آورد. سپس با نرمال‌سازی آن به بردار یکه  $\hat{k}$  دست یافت.

**بدست آوردن  $\hat{i}$**

همانطور که اشاره شد، عمل شتاب گرفتن هنگام حرکت مستقیم، تاثیر مستقیم بر داده حسگر شتاب‌سنج در جهت  $x$  دارد. بنابراین با استفاده از داده این حسگر در زمان شتاب می‌توان المان  $\hat{i}$  را بدست آورد. حسگر چرخ‌نمای جهت تعیین اینکه آیا خودرو در جهت مستقیم حرکت می‌کند

یا خیر استفاده می‌شود. بدلیل شتاب اولیه که خودرو از حالت سکون به حالت شروع به حرکت ایجاد می‌نماید، در این پروژه از داده‌های مربوط به ثانیه‌های ابتدایی حرکت برای بدست آوردن المان  $\hat{z}$  استفاده شده است. در حالت کلی نیز می‌توان از داده‌های حسگر موقعیت یاب برای بدست آوردن زمان شروع حرکت استفاده نمود.

### بدست آوردن $\hat{z}$

از آنجاییکه دستگاه مختصات از قانون دست راست پیروی می‌کند، می‌توان بردار یکه  $\hat{z}$  را توسط ضرب خارجی بردارهای  $\hat{i}$  و  $\hat{k}$  تعیین نمود.  $\hat{j} = \hat{i} \times \hat{k}$

بدین ترتیب ماتریس دوران تعیین می‌گردد. اگر بردارهای ثبت شده از طریق تلفن همراه را  $s$  بنامیم، طبق رابطه زیر می‌توان به بردار دوران یافته  $s'$  که با دستگاه مختصات خودرو هماهنگ شده است دست یافت.

$$s' = s \times R$$

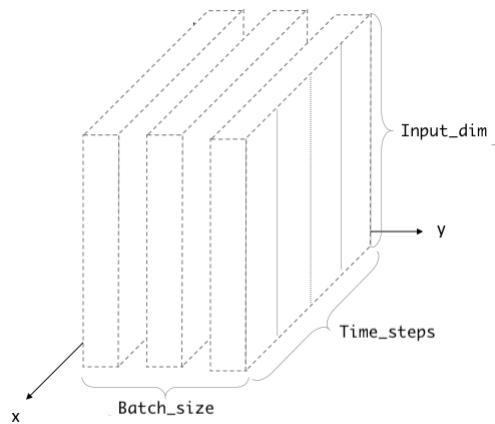
برای دریافت داده‌ها در این پروژه بدلیل یکسان نمودن دستگاه مختصات تلفن همراه و خودرو، نحوه قراردهی تلفن در داخل خودرو بصورت افقی انتخاب شده است. قسمت هماهنگ سازی دستگاه مختصات برای اجرا در محیط‌های غیر آزمایشی توسعه داده شده است.

## ۴-۳ مدل تشخیص رخداد

پس از دریافت داده‌ها، برچسب‌گذاری، و اعمال یک مرحله هموارسازی، مقدمات آموزش الگوریتم‌های یادگیری دسته‌بندی فراهم می‌گردد. این برنامه متتشکل از ۳ قسمت مجزا برای تشخیص رخدادها می‌باشد. دسته‌بند ترمز و شتاب، دسته‌بند چرخش راست و چپ، و دسته‌بند دور زدن. پیاده‌سازی دسته‌بندها توسط معماری شبکه عصبی LSTM براساس توضیحات فصل قبل انجام شده است. معماری استفاده شده برای هر ۳ دسته‌بند یکسان درنظر گرفته شده است.

### ۱-۴-۳ طراحی مدل LSTM

برای تعیین معماری شبکه عصبی جهت حل یک مسئله متغیرهای مختلفی را باید مشخص نمود. متغیرهای مدنظر در این پروژه با استفاده از تجربیات تحقیقات مرتبط، قالب داده‌های موجود، و



شکل ۳-۱۹: ساختار ورودی شبکه LSTM

آزمایش متغیرهای مختلف توسط نویسنده انتخاب شده اند.

## ورودی

ورودی LSTM آرایه ۳ بعدی است که با پارامترهای اندازه دسته<sup>۲</sup>، گام زمانی<sup>۳</sup>، و واحد خروجی<sup>۴</sup> مشخص می‌گردد و داده‌های ورودی بر اساس این ساختار به مدل داده خواهند شد. با توجه به تعداد داده‌های موجود، تعداد کلاس‌ها جهت پیش‌بینی، و اندازه دسته این ساختار تشکیل خواهد شد.

## خروجی

در لایه خروجی از تابع بیشینه هموار<sup>۵</sup> استفاده شده است. این تابع بردار ورودی را به مقادیر بین ۰ و ۱ نگاشت می‌کند، بطوریکه جمع تمامی مولفه‌های بردار خروجی ۱ شود. در حقیقت، خروجی این مدل احتمال تعلق هر نقطه زمانی به کلاس یا دسته مدنظر را اعلام می‌دارد.

<sup>2</sup>Batch size

<sup>3</sup>Time-step

<sup>4</sup>Output unit

<sup>5</sup>Softmax

Layer (type)	Output Shape	Param #
lstm_1 (LSTM)	(None, 200, 20)	1920
dense_1 (Dense)	(None, 200, 3)	63
Total params:	1,983	
Trainable params:	1,983	
Non-trainable params:	0	

شکل ۲۰-۳: معماری مدل دسته‌بند ترمز و شتاب

In [15]:	1 X_train[0:5]
Out[15]:	
<b>x      y      z</b>	
0	0.027076 -0.008920 -0.002108
1	0.005088 -0.001176 0.014310
2	0.005863 -0.022363 -0.011907
3	0.009860 -0.028556 -0.009833
4	0.001507 -0.014297 0.014919

شکل ۲۱-۳: نمونه ورودی به مدل دسته‌بند ترمز و شتاب

## تابع ضرر

یکی از بخش‌های مهم در حین فرآیند یادگیری ماشینی انتخاب روشی جهت ارزیابی عملکرد الگوریتم یادگیری با توجه به داده‌های موجود می‌باشد. تابع ضرر<sup>۶</sup> که انجام این امر را بر عهده دارد، ارتباط بین پیش‌بینی مدل و نتایج حقیقی را مشخص می‌کند. در صورت افزایش مقدار تابع ضرر، می‌توان مدل را با استفاده از روش‌های بهینه‌سازی تصحیح نمود و خطای پیش‌بینی را کاهش داد. عوامل متعددی از جمله نوع مسئله و الگوریتم یادگیری بر انتخاب تابع ضرر تاثیرگذار هستند. در این پژوهه با توجه به اینکه با مسئله دسته‌بندی چند کلاسه<sup>۷</sup> و دوتایی<sup>۸</sup> مواجه هستیم، از توابع آنتروپی متقاطع<sup>۹</sup> چند کلاسه و دوتایی استفاده شده است.

## ۲-۴-۳ دسته‌بند ترمز و شتاب

ورودی این مدل، داده حسگر شتاب‌سنج در ۳ محور x, y, z است. مدل ساخته شده بر اساس رابط برنامه‌نویسی کاربردی کراس بصورت شکل ۲۰-۳ می‌باشد. داده‌ها بصورت پنجره‌های ۲۰۰ تایی به مدل داده می‌شوند و ترتیب پنجره‌ها بصورت تصادفی انتخاب می‌گردد.

تصویر ۲۱-۳ نشان‌دهنده نمونه پاسخ ورودی به مدل دسته‌بندی ترمز و شتاب که با استفاده

<sup>6</sup>Loss function

<sup>7</sup>Multi-class classification

<sup>8</sup>Binary classification

<sup>9</sup>Cross-entropy

```
In [21]: 1 y_train_onehot[0:5]
Out[21]: [array([1, 0, 0]),
           array([1, 0, 0]),
           array([1, 0, 0]),
           array([1, 0, 0]),
           array([1, 0, 0])]
```

شکل ۲۲-۳: نمونه پاسخ ورودی به مدل دسته بندی ترمز و شتاب

```
In [5]: 1 X_train[0:5]
Out[5]:
   magneticHeading    gyroX    gyroY    gyroZ      Yaw      Roll     Pitch     accx     accy     accz
0    215.865997 -0.026412  0.007243 -0.008219  2.531805  0.023190 -0.000239 -0.004740  0.000799  0.009695
1    215.865997 -0.029835  0.012782 -0.005231  2.531806  0.023225 -0.000246  0.006699 -0.014742  0.013219
2    215.865997 -0.032145  0.016516 -0.005960  2.531803  0.023273 -0.000265  0.004531 -0.010519 -0.014965
3    215.865997 -0.030438  0.010106 -0.007348  2.531787  0.023259 -0.000276 -0.009997  0.004973 -0.010418
4    215.865997 -0.027040  0.010524 -0.005051  2.531785  0.023250 -0.000267 -0.011743  0.003425  0.013737
```

شکل ۲۳-۳: نمونه ورودی به مدل دسته بندی چرخش

```
In [21]: 1 y_train_onehot[0:5]
Out[21]: [array([1, 0, 0]),
           array([1, 0, 0]),
           array([1, 0, 0]),
           array([1, 0, 0]),
           array([1, 0, 0])]
```

شکل ۲۴-۳: نمونه پاسخ ورودی به مدل دسته بندی چرخش

از دستگاه one-hot ساخته شده است می‌باشد. در این دسته‌بند ۳ حالت موجود است. • بدون رخداد، ۱ رخداد ترمز، ۲ رخداد شتاب. از این مقادیر آرایه‌های مانند شکل ۲۲-۳ ساخته خواهد شد. در این مدل، پیش‌بینی به ازای هر گام زمانی صورت می‌گیرد و احتمال تعلق هر نقطه در یکی از ۳ دسته مدنظر مشخص می‌شود.

### ۳-۴-۳ دسته‌بند چرخش به راست و چپ

ساختار و معماری مدل همانند دسته‌بندی ترمز و شتاب می‌باشد. طبق نمونه داده‌های آموزش این مدل در تصویر ۲۳-۳، از داده حسگرهای شتاب‌سنج، چرخش‌نما، و مغناطیس‌سنج استفاده شده است. برای پاسخ ورودی نیز همانند قسمت قبل از دستگاه one-hot استفاده شده و ۳ حالت موجود است. • برای حالت بدون رخداد، ۱ برای رخداد چرخش به راست، و ۲ برای رخداد چرخش به چپ.

Layer (type)	Output Shape	Param #
lstm_1 (LSTM)	(None, 200, 20)	2480
dense_1 (Dense)	(None, 200, 2)	42
Total params:	2,522	
Trainable params:	2,522	
Non-trainable params:	0	

شکل ۳-۲۵: معماری مدل دسته‌بند دور زدن

In [5]:	Out[5]:
	1 X_train[0:5]
	magneticHeading gyroX gyroY gyroZ Yaw Roll Pitch accx accy accz
0	215.865997 -0.026412 0.007243 -0.008219 2.531805 0.023190 -0.000239 -0.004740 0.000799 0.009695
1	215.865997 -0.029835 0.012782 -0.005231 2.531806 0.023225 -0.000246 0.006699 -0.014742 0.013219
2	215.865997 -0.032145 0.016516 -0.005960 2.531803 0.023273 -0.000265 0.004531 -0.010519 -0.014965
3	215.865997 -0.030438 0.010106 -0.007348 2.531787 0.023259 -0.000276 -0.009997 0.004973 -0.010418
4	215.865997 -0.027040 0.010524 -0.005051 2.531785 0.023250 -0.000267 -0.011743 0.003425 0.013737

شکل ۳-۲۶: نمونه ورودی به مدل دسته بند دور زدن

In [7]:	Out[7]:
1 y_train_onehot[0:5]	[array([1, 0]), array([1, 0]), array([1, 0]), array([1, 0]), array([1, 0])]

شکل ۳-۲۷: نمونه پاسخ ورودی به مدل دسته‌بند دور زدن

### ۴-۴-۳ دسته‌بند دور زدن

در این دسته‌بند خروجی شبکه عصبی بدلیل وجود ۲ کلاس در مسئله، ۲ بعد دارد. از ویژگی‌های حسگرهای قسمت قبل استفاده می‌شود و همچنین ۰ نمایانگر حالت بدون رخداد و ۱ نشان‌دهنده رخداد دور زدن می‌باشد.

در الگوریتم‌های یادگیری ماشینی، معمولاً داده‌ها را به ۲ دسته آموزش و آزمایش تقسیم می‌کنند، سپس مدل بر اساس داده آموزش شروع به آموختن نموده و مطابق با داده آزمایش، امتحان می‌شود. مدل بر اساس عملکردی که در ارتباط با مجموعه داده آزمایش ارائه می‌دهد، سنجیده خواهد شد. در این پژوهه نسبت داده آموزش و آزمایش بر اساس روش اعتبار سنجی متقابل که جلوتر توضیح داده خواهد شد انتخاب شده است.

### ۵-۳ پس پردازش داده‌ها

راننده در حین رانندگی به بسیاری از اتفاقات پیرامون واکنش نشان می‌دهد که در نتیجه رخدادهای بسیار زیادی رقم خواهد خورد. در حالیکه، برای درک کیفیت رانندگی نیازی به دریافت و ثبت تمام رخدادها نمی‌باشد و صرفاً رخدادهای اصلی هستند که تاثیری زیادی بر روی

خودرو و سایر عوامل دارند. بنابراین پس از تشخیص رخداد توسط دسته‌بندهای توسعه داده شده، نیاز داریم تا با اعمال بررسی‌هایی رخدادهای معتبر را شناسایی و آن‌هایی که غیر معتبر هستند را حذف نماییم. از ویژگی‌های مختلفی برای شناسایی رخداد معتبر می‌توان استفاده نمود. در این پژوهه از عوامل مدت زمان رخداد و انرژی سیگنال رخداد استفاده می‌شود.

برای آنکه حد آستانه مشخصی جهت معتبر بودن یک رخداد درنظر گرفته شود، علاوه بر در نظر گرفتن مقادیر پژوهش‌های انجام شده [۱۵]، به بررسی عوامل بیان شده در رخدادهای کم خطر و پر خطر می‌پردازیم تا درک بهتری از نقش این عوامل در رخدادها بدست آید. داده‌هایی که برای این بررسی استفاده شده‌اند، از مخزن [۶] بدست آمده است. این مقایسه به درک تفاوت مشخصه‌های رخدادهای پر خطر و عادی نیز کمک می‌کند.

در تصاویر ۲۸-۳، ۲۹-۳، ۳۰-۳، و ۳۱-۳ نمودار جعبه‌ای انرژی و مدت زمان رخدادهای پر خطر و عادی آورده شده است. با کمک این نمودار می‌توان نحوه پراکندگی داده‌های مربوط به این عوامل را در رخدادهای مشابه بررسی کرده و براساس آن حد آستانه‌ای جهت انتخاب رخداد معتبر مشخص نمود.

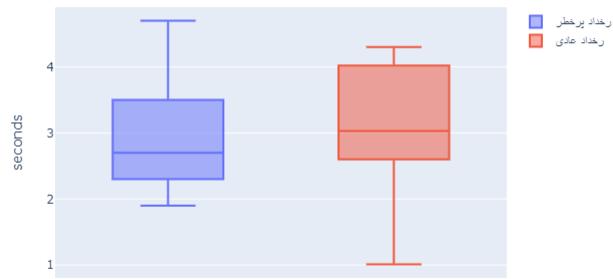


شکل ۳-۲۸: مقایسه انرژی سیگنال‌های رخداد شتاب و ترمز

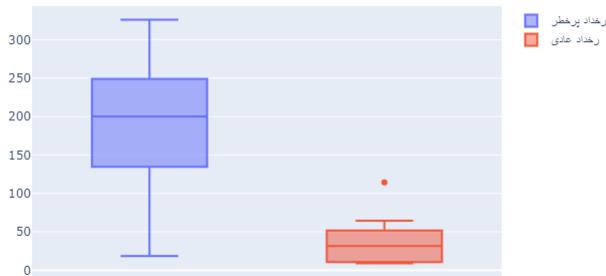
باتوجه به مقایسه‌های انجام شده در ارتباط با انرژی سیگنال و همچنین طول مدت زمان انجام رخداد، آستانه‌های زیر تعریف شده است که البته کمی سخت‌گیرانه تر از داده‌های بدست آمده در مقایسه قسمت قبل نیز هستند:

### رخدادهای ترمز و شتاب

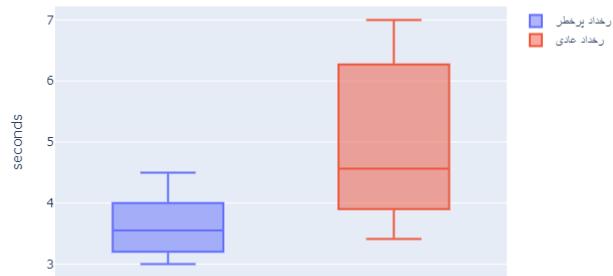
رخدادهای پیش‌بینی شده با انرژی بالاتر از ۳۰ و طول زمان بیشتر از ۵.۰ ثانیه معتبر می‌باشند.



شکل ۳-۲۹: مقایسه طول زمانی رخداد شتاب و ترمز



شکل ۳-۳۰: مقایسه انرژی سیگنال‌های رخداد چرخش



شکل ۳-۳۱: مقایسه طول زمانی سیگنال‌های رخداد چرخش

### رخدادهای چرخش

رخدادهای پیش‌بینی شده با انرژی بالاتر از ۱۰ و طول زمان بیشتر از ۱ ثانیه معتبر می‌باشند.

آستانه شدت			نوع رخداد
زیاد	متوسط	کم	
$0.4g <  a_x $	$0.2g <  a_x  < 0.4g$	$0.1g <  a_x  < 0.2g$	شتاب، ترمز
$0.4g <  a_y $	$0.2g <  a_y  < 0.4g$	$0.1g <  a_y  < 0.2g$	چرخش

شکل ۳-۲-۳: تعریف حدود آستانه ناگهانی بودن رخدادها

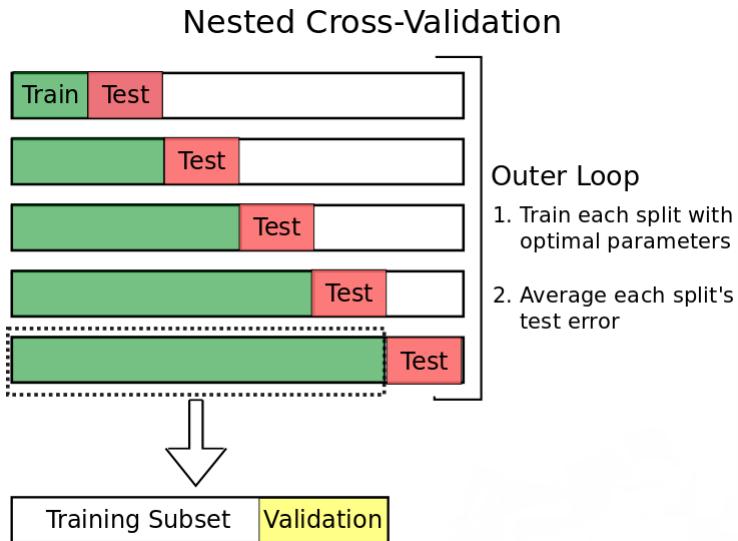
### ۱-۵-۳ میزان ناگهانی (شدید) بودن عمل

پس از مشخص شدن رخدادهای صورت گرفته در یک دوره از رانندگی، نیاز است تا میزان ناگهانی رخ دادن آن عمل را نیز بسنجیم. بنابراین، حد آستانه‌ای برای رخدادها در ۳ بازه درنظر گرفته می‌شود تا به ناگهانی انجام شدن رخدادها مقادیر زیاد، متوسط، و کم نسبت داده شود. تعریف این آستانه طبق جدول ۳-۲-۳ صورت گرفته که از مقاله [۱۴] بدست آمده است. با توجه به میانگین ۳ نقطه از بیشترین مقادیر سیگنال رخداد، نسبت به شدت آن تصمیم‌گیری می‌شود. بنابراین لیستی از رخدادهای صورت گرفته در یک دوره از رانندگی بدست خواهد آمد که نشان‌دهنده رخدادها به همراه شدت رخ دادنشان می‌باشد. با توجه به این لیست می‌توان کیفیت رانندگی را تخمین زد که نحوه سنجش آن در ادامه توضیح داده خواهد شد.

### ۳-۶ ارزیابی و انتخاب مدل دسته‌بند

برای انتخاب و صحتسنجی مدل دسته‌بند نیاز به یک روش ارزیابی داریم. یکی از روش‌های ارزیابی که در پژوهش‌های مربوط به الگوریتم‌های یادگیری آماری استفاده می‌شود، روش ارزیابی اعتبار سنجی متقابل است. این روش یک روش ارزیابی مدل است که تعیین می‌نماید نتایج یک تحلیل آماری بر روی یک مجموعه داده تا چه اندازه قابل تعمیم و مستقل از داده‌های آموزشی است. این روش به‌طور ویژه در کاربردهای پیش‌بینی مورد استفاده قرار می‌گیرد تا مشخص شود مدل موردنظر تا چه اندازه در عمل مفید خواهد بود.

یکی از الگوریتم‌های این روش، اعتبار سنجی متقابل k-fold می‌باشد. در این نوع اعتبار سنجی متقابل، نمونه اصلی به‌طور تصادفی به زیرنمونه‌های فرعی با اندازه k تقسیم می‌شود. از زیرنمونه‌های فرعی، یک زیرنمونه به عنوان داده‌های اعتبار سنجی برای آزمایش مدل حفظ شده و k-1 زیرنمونه



شکل ۳-۳: مدل پیشنهادی اعتبارسنجی متقابل

دیگر به عنوان داده‌های آموزشی استفاده می‌شوند. فرایند اعتبارسنجی متقابل،  $k$  بار تکرار می‌شود که همه‌ی نمونه‌ها به‌طور دقیق یک بار به عنوان داده‌ی اعتبارسنجی مورد استفاده قرار گیرند. پس از آن می‌توان جهت تولید یک برآورد، از میانگین نتایج بدست آمده استفاده نمود. مزیت این روش این است که همه مشاهدات برای هر دو عمل آموزش و اعتبار مورد استفاده قرار می‌گیرند. از آنجاییکه داده مورد نظر در این پروژه، سری زمانی است و ترتیب داده‌ها اهمیت دارد، نمی‌توان بصورت تصادفی داده‌ها را به بخش‌های مختلف تقسیم نمود. بنابراین از روش دیگری، مشابه روش توضیح داده شده، استفاده شده است.

الگوریتم استفاده شده در پروژه، مطابق تصویر ۳-۳، داده‌ها را در ۵ مرحله به مجموعه آموزش و آزمایش تقسیم می‌نماید. و عملکرد نهایی بر اساس میانگین عملکرد هر کدام از ۵ مدل ارائه شده است.

برای بررسی دقت مدل نیاز به معیاری جهت ارائه عملکرد مدل انتخابی خواهیم داشت. در این پروژه، ۲ معیار جهت بررسی این مسئله ارائه می‌گردند.

### ۱-۶-۱ معيار امتياز اف-۱

در مدل‌های آماری، گزارش تعداد موارد درست پیش‌بینی شده و نسبت آن به تعداد کل موارد، الزاماً نشان‌دهنده چگونگی عملکرد و دقت مدل نمی‌باشد و نیاز است تا از معیارهای دیگری به غیر از نسبت موارد صحیح پیش‌بینی شده استفاده گردد. از آنجاییکه نسبت حضور کلاس‌ها

نسبت به یکدیگر در داده‌های رخداد رانندگی متعادل نیست و تعداد حالات بدون رخداد بسیار بیشتر از تعداد حالاتی است که رخدادی صورت گرفته، از معیار امتیاز اف-۱<sup>۱۰</sup> برای بررسی دقت این مدل استفاده می‌شود.

معیار امتیاز اف-۱، که در رابطه <sup>۳-۳</sup> مشخص شده است، به نوعی میانگین بین ۲ عامل دقت (Precision) و پوشش (Recall) در سامانه‌های آماری می‌باشد. Precision، دقت مدل در میان داده‌های پیش‌بینی شده است (مثال، نسبت تعداد ترمزهای واقعی تشخیص داده شده به کل تعداد ترمزهای تشخیص داده شده) و Recall نسبت داده‌های پیش‌بینی شده به تعداد کل داده‌های قابل پیش‌بینی است. (مثال، نسبت تعداد ترمزهای واقعی تشخیص داده شده به تعداد کل ترمزهای واقعی موجود)

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \quad (3-3)$$

برای محاسبه این معیار در این پروژه، عوامل دقت و پوشش بر اساس رخدادهای پیش‌بینی شده و تعداد کل رخدادهای موجود در داده در نظر گرفته شده‌اند.

## ۲-۶-۳ معیار نسبت اشتراک به اجتماع

امتیاز اف-۱ عملکرد مدل در تشخیص رخدادها را بررسی می‌کند اما برای آنکه دید درستی از دقت آن در تشخیص یک رخداد مشخص بدست آید از نسبت اشتراک به اجتماع<sup>۱۱</sup> استفاده شده است. این روش بعنوان یکی از معیارهای اصلی ارزیابی برنامه‌های بازنگاری شیء<sup>۱۲</sup> در زمینه بینایی رایانه‌ای<sup>۱۳</sup> می‌باشد. در حالت کلی، می‌توان از آن در برنامه‌هایی که خروجی‌شان پیش‌بینی بازه‌های محدود و مشخصی است، استفاده نمود.

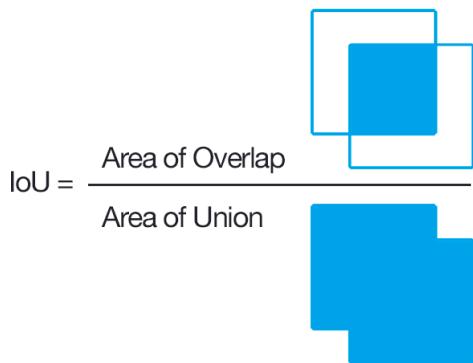
در تصویر <sup>۲-۳</sup> نمونه‌ای از پیش‌بینی به همراه پاسخ درست آن آورده شده است. همانطور که مشاهده می‌شود مدل پیشنهادی قادر به پیش‌بینی این رخداد بوده، اما خطأ در تشخیص ابتدا و انتهای بازه نیز قابل مشاهده است. پس از محاسبه نسبت اشتراک و اجتماع این دو ناحیه و تقسیم آن‌ها بر یکدیگر می‌توان به نسبت همپوشانی که عددی بین ۰ و ۱ است دست یافت. در

<sup>10</sup>F1-score

<sup>11</sup>Intersection over Union

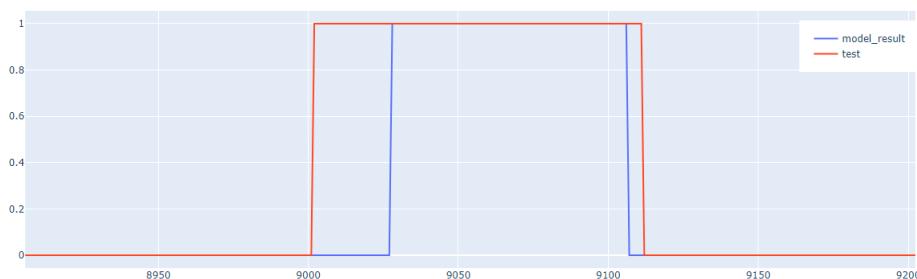
<sup>12</sup>Object Recognition

<sup>13</sup>Computer Vision



شکل ۳-۳: رابطه نسبت اشتراک به اجتماع

نهایت با در نظر گرفتن معیارهای اف-۱ و نسبت اشتراک به اجتماع، عملکرد کلی مدل تشخیص رخداد مورد بررسی قرار می‌گیرد. در جدول ۲-۳ امتیازات بدست آمده توسط این برنامه آورده شده است.



شکل ۳-۴: نمونه‌ای از پیش‌بینی رخداد

### ۳-۶-۳ بررسی خطای دسته‌بندی

شکل ۳-۴ نشان‌دهنده نمونه‌ای از خطای مدل است که به اشتباه سیگنال حرکتی را در دسته رخداد ترمز قرار می‌دهد، در حالیکه در داده جمع‌آوری شده رخدادی ثبت نشده است. یکی از موارد تاثیرگذار در تصمیم اشتباه مدل، واکنش‌های اعمال شده از طرف راننده است. شرایط محیط، نوع جاده‌ای که خودرو در آن حرکت می‌کند، و دینامیک خودرو در سیگنال‌های حرکتی که توسط حسگرها جمع‌آوری می‌شود تاثیرگذار هستند و واکنش‌های مختلف راننده در شرایطی می‌تواند منجر به تغییرات حرکتی مشابه با الگوی ترمز و شتاب شود. بعنوان مثال، می‌توان به عمل کاهش سرعت ناگهانی راننده با برداشتن پا از روی پدال گاز اشاره نمود که الگوی مشابه با عمل ترمز دارد اما در واقع رخداد ترمز با فشردن پدال ترمز صورت نگرفته است. مورد

دیگری که احتمال خطای مدل را افزایش می‌دهد، وجود خطا در عمل ضبط صدای راننده جهت برچسب‌گذاری داده هنگام رانندگی می‌باشد. اگرچه سعی بر انجام شدن با کمترین میزان خطا شده، اما همچنان ممکن است راننده به ناخودآگاه اعمالی را انجام دهد که در زمان صحیح انجام رخداد را بیان نکرده باشد.

بنابراین، استفاده از پارامترهای بیشتر جهت تشخیص رخداد واقعی، برچسب‌گذاری داده‌ها با درصد خطای کمتر و همچنین تشخیص و کاهش نویز داده‌ها می‌تواند دقیق سامانه را افزایش دهد.

False Sample Data



شکل ۳-۳۶: نمودار سیگنال حسگر شتاب‌سنج که بصورت اشتباه بعنوان رخداد ترمز دسته‌بندی شده است

```
In [186]: 1 events_acc
Out[186]: [('acc', 1793, 1886, 93, 0),
 ('acc', 1920, 1969, 49, 0),
 ('acc', 2234, 2344, 110, 0),
 ('acc', 5297, 5398, 101, 0),
 ('acc', 5436, 5514, 78, 0),
 ('acc', 6231, 6322, 91, 0),
 ('acc', 20978, 21198, 220, 1),
 ('acc', 21676, 21746, 70, 1),
```

شکل ۳-۳۷: رخدادهای شتاب/ترمز انجام شده

```
In [190]: 1 events_turn
```

```
Out[190]: [('rt', 1358, 1957, 599, 0),
 ('lt', 30481, 30739, 258, 1),
 ('lt', 37459, 37798, 339, 1),
 ('lt', 44758, 45198, 440, 1),
 ('rt', 46906, 47107, 201, 0),
 ('lt', 49807, 49993, 186, 1),
 ('rt', 51023, 51538, 515, 1)]
```

شکل ۳۸-۳: رخدادهای چرخش انجام شده

```
In [191]: 1 events_uturn
```

```
Out[191]: [('ut', 30439, 30707, 268, 1),
 ('ut', 37425, 37622, 197, 1),
 ('ut', 44688, 45151, 463, 1)]
```

شکل ۳۹-۳: رخدادهای دور زدن انجام شده

دسته‌بند	دور زدن	چرخش به راست و چپ	شتاب/ترمز	نسبت اشتراک به اجتماع	امتیاز اف-۱	پوشش
				%۴۹	%۷۸	%۸۳
				%۷۳	%۷۹	%۱۰۰
				%۵۱	%۷۵	%۷۹

جدول ۲-۳: جدول نتایج مدل تشخیص رخداد

## ۷-۳ برنامه بررسی نوع رانندگی

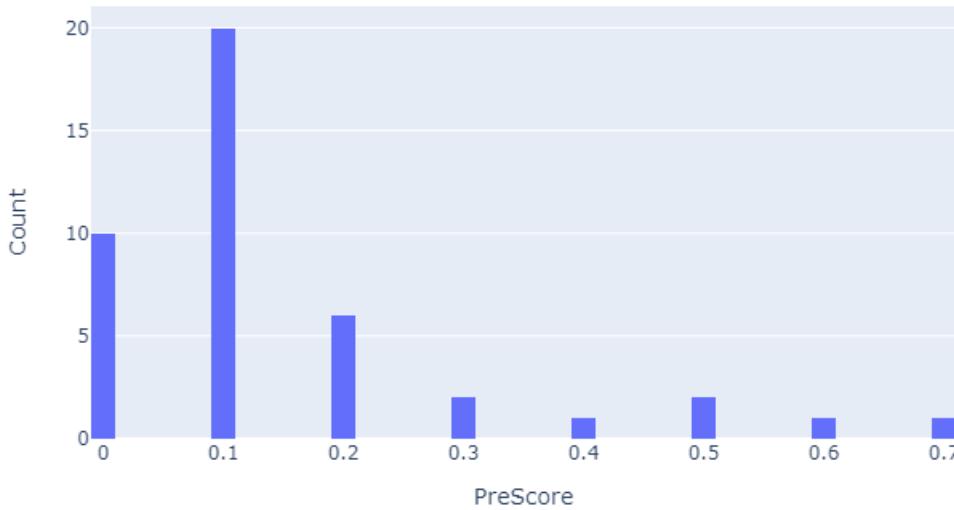
هدف این برنامه سنجش کیفیت رانندگی توسط لیست رخدادهایی است که در مراحل قبل بدست آمده است. تا این مرحله، اطلاعات کاملی از وقوع رخدادهای رانندگی به همراه نوع انجام‌شان در اختیار داریم. در این برنامه پس از مقایسه مشخصات رانندگی مدنظر با رانندگی مطلوب که بطور تجربی بدست آمده است، نوع رانندگی انجام شده مورد بررسی قرار خواهد گرفت.

در این پروژه، رانندگی مطلوب، رانندگی با کمترین تعداد رخداد (نگهانی) در نظر گرفته می‌شود. تعداد کم رخدادها در هر بازه زمانی علاوه بر مسئله امنیت بر الگوی مصرف سوخت خودرو نیز اثر قابل توجهی خواهد گذاشت و نشان‌دهنده وضعیت پایدار در طی رانندگی است. طبق تعریف ارائه شده از رانندگی خوب و مطلوب، رابطه ۵-۳ پیش-امتیازی را به هر واحد از دوره رانندگی اختصاص می‌دهد. پیش-امتیاز برای هر ۵ رخداد متوالی در لیست رخدادها محاسبه شده و بر مدت زمان شروع رخداد اول تا انتهای رخداد پنجم تقسیم می‌گردد. بنابراین  $k_3, k_2, k_1$  این امتیاز می‌تواند مستقل از طول زمانی رخداد، مورد بررسی قرار گیرد. مقادیر ۱, ۲, ۴ به ترتیب ثابت‌هایی هستند که بصورت تجربی بدست می‌آیند و در این برنامه از مقادیر ۱, ۲, ۴ به ترتیب برای آن‌ها استفاده شده است. Low, Medium, High تعداد رخدادهای انجام شده با میزان نگهانی بودن کم، متوسط، و زیاد در بازه مشخص شده هستند. هرچه مقدار پیش-امتیاز بیشتر باشد، استفاده از رخدادهای بیشتر و نگهانی‌تر در دوره مشخص شده از رانندگی بیشتر بوده است و مقدار بالای پیش-امتیاز در طولانی مدت نشان‌دهنده کیفیت نامطلوب رانندگی فرد می‌باشد. نحوه دیگر بدست آوردن این متغیر محاسبه آن به ازای هر کیلومتر از مسیر است. در روش ارائه شده نیز از عدد ۵ رخداد بدلیل تخمین مسافت ۱ کیلومتر استفاده شده است.

$$\forall e \in EventList, e = \{e_i, e_{i+1}, e_{i+2}, e_{i+3}, e_{i+4}\} \quad (4-3)$$

$$PreScore(e) = [(k_1 \cdot Low_e + k_2 \cdot Medium_e + k_3 \cdot High_e) / duration] \quad (5-3)$$

یکی از راههای بررسی کیفیت رانندگی، مقایسه آن با رانندگی مطلوب ارائه شده است. در این پروژه سعی شده است تا داده رانندگی خوب توسط ۲ فرد با سابقه ۲۰ سال رانندگی جمع‌آوری



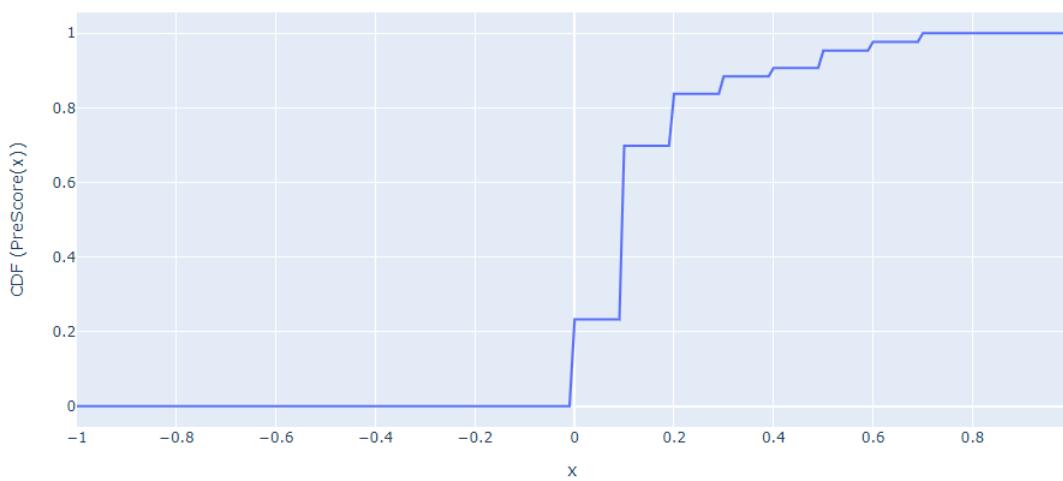
شکل ۳-۴: نمودار Histogram پیش‌امتیاز رانندگی خوب

شود. تمام رخدادهای مورد بررسی در این پروژه در داده موجود هستند و عمل رانندگی نیز بدون پیش‌فرض ذهنی ثبت داده برای راننده انجام شده است. داده‌ها بعنوان ورودی به سامانه وارد شدند و پس از آن عمل شناسایی و محاسبه پیش‌امتیاز انجام شد. پیش‌امتیازهای بدست آمده توسط نمودار ۴۰-۲ تصویر شده است که توزیع مقادیر را برای یک رانندگی خوب مشخص می‌کند. ترکیب داده‌های حرکتی ۲ راننده توزیع کامل‌تری ارائه می‌نماید. برای بدست آوردن امتیاز نهایی جهت مقایسه و سنجش کیفیت رانندگی طبق رابطه ۶-۳ ازتابع توزیع تجمعی<sup>۱۴</sup> استفاده می‌شود. این رابطه یک مقدار بین ۰ و ۱ را به رانندگی فرد اختصاص می‌دهد. هرچه این نمره به ۱ نزدیک‌تر باشد، رانندگی فرد پایدارتر و شامل رخدادهای (ناگهانی) کمتری است. نمودار تابع توزیع تجمعی پیش‌امتیاز رانندگی خوب در شکل ۴۱-۳ نشان داده شده است.

$$Score_e = 1 - CDF_{standard}(PreScore(e)) \quad (6-3)$$

جهت بررسی نحوه کار سامانه و مدل آماری ارائه شده، از پردازش نمونه داده رانندگی پرخطر موجود در [۶] و داده ضبط شده استفاده می‌شود. داده رانندگی [۶] از ۶ رخداد عادی و ۱۶ رخداد

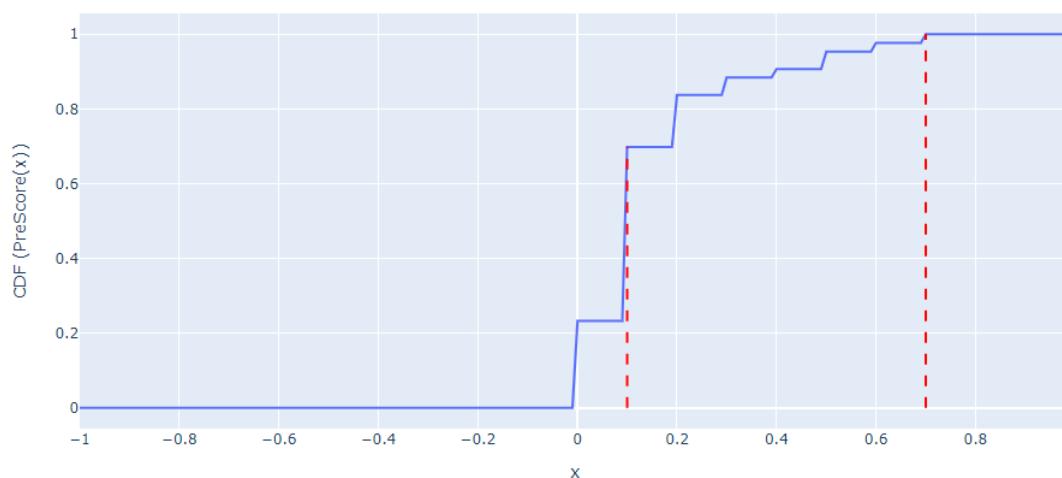
<sup>14</sup>Cumulative Distribution Function



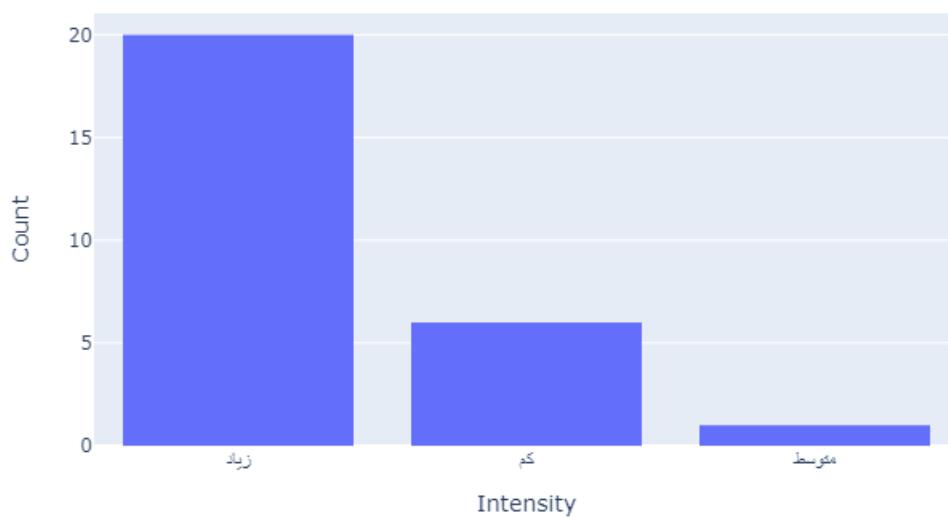
شکل ۳-۴۱: نمودار تابع توزیع تجمعی پیش‌امتیاز رانندگی خوب

پرخطر تشکیل شده است که طی ۱۳ دقیقه رانندگی بدست آمده است. انجام رخدادهای پرخطر توسط راننده با سابقه ۱۵ سال انجام شده و دستورالعمل مشخصی جهت اجرای آن رخدادها به راننده داده نشده است. داده دوم به منظور آزمایش برنامه سنجش کیفیت رانندگی و مقایسه با داده رانندگی پرخطر [۶] تهیه شده است. این داده شامل ۲۰ دقیقه رانندگی است که مانند داده [۶] دستورالعمل مشخصی جهت اجرای رخدادها به راننده داده نشده است. بدلیل استفاده سامانه توسعه داده شده از حسگرهای متفاوت با داده‌های موجود در [۶]، امکان بررسی نوع رفتار در خصوص رخدادهای چرخش و دور زدن وجود نداشته است و این بررسی مربوط به رخداد ترمز و شتاب می‌باشد.

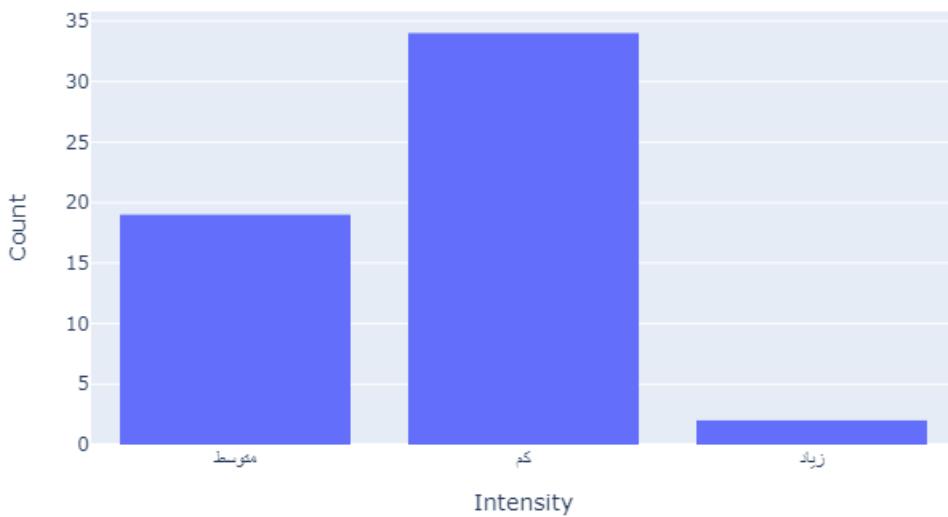
داده‌های بدست آمده را بعنوان ورودی به سامانه داده تا رخدادهای موردنظر شناسایی و براساس میزان ناگهانی بودن برچسب‌گذاری شوند. در خروجی سامانه ۲۷ رخداد شناسایی شده است که ۲۰ مورد از آن‌ها با شدت زیاد، ۶ مورد با شدت کم، و ۱ مورد با شدت متوسط صورت گرفته است. در داده تهیه شده ۵۵ رخداد شناسایی شده است که ۲ مورد آن‌ها با شدت زیاد، ۳۴ مورد با شدت کم، و ۱۹ مورد با شدت متوسط انجام شده است. پس از شناسایی رخدادها، امتیازها را محاسبه و توزیع امتیازها را با رانندگی موردنظر مقایسه می‌نماییم. طبق رابطه  $E-3$  نیاز به استفاده از تابع توزیع تجمعی پس از محاسبه پیش‌امتیاز می‌باشیم. بعنوان مثال اگر مقادیر ۱.۰ و ۰.۷ را بعنوان پیش‌امتیاز بدست آورده باشیم، می‌توانیم طبق شکل ۴۲-۳ مقدار تابع توزیع تجمعی را بیابیم که در این مثال به ترتیب برابر با ۰.۶۹ و ۱ می‌باشد.



شکل ۳-۴۲: مشخص نمودن مقدار تابع توزیع تجمعی دو نمونه داده



شکل ۳-۴۳: توزیع رخدادها در نمونه داده [۶]



شکل ۳-۴۴: توزیع رخدادها در نمونه داده تهیه شده

```
In [430]: 1 pre_score_test
Out[430]: array([0.7, 0.3, 0.8, 0.7, 1. , 0.8])
```

شکل ۳-۴۵: پیش-امتیاز به ازای هر ۵ رخداد داده [۶]

```
In [7]: 1 scores = []
2 for i in pre_score:
3     scores.append(1 - CDF(data, i))
4 print("Score per time unit:", scores)
5 print("Mean Score:", np.mean(scores))

Score per time unit: [0.0, 0.11627906976744184, 0.0, 0.0, 0.0, 0.0]
Mean Score: 0.019379844961240306
```

شکل ۳-۴۶: امتیاز نسبت داده شده به رانندگی داده [۶]

```
In [500]: 1 pre_score_test
Out[500]: array([0. , 0. , 0.1, 0. , 0.1, 0. , 0.3, 0. , 0.1, 0. , 0.1, 0.1])
```

شکل ۳-۴۷: پیش-امتیاز به ازای هر ۵ رخداد داده ضبط شده

```
In [13]: 1 scores = []
2 for i in pre_score:
3     scores.append(1 - CDF(data, i))
4 print("Score per time unit:", scores)
5 print("Mean Score:", np.mean(scores))

Score per time unit: [0.7674418604651163, 0.7674418604651163, 0.7674418604651163, 0.7674418604651163, 0.7674418604651163, 0.7674418604651163, 0.3023255813953488, 0.3023255813953488, 0.3023255813953488, 0.3023255813953488, 0.3023255813953488, 0.11627906976744184]
Mean Score: 0.5193798449612402
```

شکل ۳-۴۸: امتیاز نسبت داده شده به رانندگی داده ضبط شده

در تصاویر ۴۵-۳ و ۴۶-۳ پیش-امتیاز و امتیازهای نسبت داده شده به داده [۶] مشخص شده است. پیش-امتیاز طبق رابطه مشخص شده هر ۵ رخداد پشت سر هم را درنظر می‌گیرد. امتیازات مربوط به رانندگی ضبط شده در تصاویر ۴۷-۳ و ۴۸-۳ نشان داده شده است. میانگین امتیاز داده شده به داده [۶] برابر با ۰.۰۱ و میانگین امتیاز داده شده به داده ضبط شده برابر با ۰.۵۱ بوده است. بنابراین سامانه توانسته است به داده رانندگی پرخطر مشخص شده و همچنین رانندگی عادی نمره مدنظر را اختصاص دهد که از آن می‌توان بعنوان مقایسه کیفیت دو رانندگی در طی یک مسیر استفاده نمود.

### ۸-۳ جمع‌بندی

در این پژوهه برنامه‌های هماهنگ‌سازی دستگاه مختصات، شناسایی رخداد، و همچنین بررسی نوع رانندگی پیاده‌سازی شده است. پیاده‌سازی‌ها توسط زبان برنامه‌نویسی پایتون، کتابخانه‌های نرم‌افزاری تنسورفلو و کراس انجام گرفته است. داده‌ها توسط نرم‌افزار مربوطه جمع‌آوری و برای تحلیل به سامانه فرستاده می‌شوند. پس از اجرای عملیات پیش‌پردازش رخدادهای مورد بررسی شناسایی می‌گردند. در نهایت نوع و رفتار راننده توسط روابط ارائه شده ارزیابی می‌شود.

## فصل چهارم

### جمع‌بندی، نتیجه‌گیری و پیشنهادات

## ۱-۴ جمع‌بندی و نتیجه‌گیری

خودروها به بخش قابل توجهی از زندگی روزمره انسان‌ها تبدیل شده‌اند بطوریکه شاهد استفاده، تولید، و پیشرفت چشمگیر تکنولوژی‌های مرتبط هستیم. صنایع مختلفی مانند تولید و طراحی خودرو، تاکسی‌های برخط، سامانه‌های اجاره خودرو، خودروهای هوشمند، بیمه، و ... بطور مستقیم یا غیرمستقیم با خودرو در ارتباط هستند. بنابراین بدست آوردن اطلاعات و داده‌های حرکتی-دینامیکی خودرو و تحلیل رفتار رانندگان منجر به بهبود فرآیندها و تصمیم‌گیری‌ها در بخش‌های مختلفی خواهد شد.

از سویی دیگر، استفاده همه روزه و فراغیر شدن تلفن‌های همراه، ساعت‌های هوشمند، و دستبندهای سلامتی آن‌ها را به منبع مناسبی جهت جمع‌آوری داده از محیط اطراف و کاربران تبدیل نموده است. علاوه بر آن، بهبود مدل‌های یادگیری ماشینی و آماری و توسعه سامانه‌های پردازش کلان‌داده‌ها بستر پردازشی مناسبی را برای این دسته از کاربردها فراهم ساخته است. کاربردهایی که با پایش و شناخت حرکات انسان، خودرو، و سایر دستگاه‌های مکانیکی و الکترونیکی همراه هستند. هدف این پروژه نیز شناخت، بررسی، و امکان‌سنجی ترکیب این فناوری‌ها با یکدیگر بوده است تا به کمک سخت‌افزار و نرم‌افزارهای موجود بتوان سامانه‌های دستیار راننده را توسعه داد.

همانطور که در نوشته ارائه شد، با استفاده از حسگرهای اینرسی تلفن همراه می‌توان به دقت مطلوبی در شناسایی رخدادهای رانندگی دست یافت و از آن برای بررسی نوع و رفتار راننده استفاده نمود. تشخیص و اصلاح رفتار نادرست و پرخطر چه برای رانندگان و چه برای افرادی که در حال یادگیری رانندگی هستند، می‌تواند در کاهش آمار تصادفات جاده‌ای، میزان استهلاک خودروها، و آلودگی محیط زیست اثرگذار باشد. همچنین این اطلاعات می‌تواند بعنوان پایه مدل‌سازی ریاضی رفتار رانندگان در شرایط و محیط‌های مختلف جهت توسعه و حل مسئله پیچیده خودروهای خودران درنظر گرفته شوند.

برای پیاده‌سازی پروژه، تلفن‌های همراه در خودروها قبل از شروع به حرکت قرارداده می‌شوند. حسگرهای شتاب‌سنج، چرخش‌نما، و مغناطیس‌سنج تلفن‌های همراه داده‌های مرتبط را در حین رانندگی ثبت نموده و در تلفن‌همراه ذخیره می‌کنند. پس از اتمام دوره رانندگی، از این داده‌ها برای شناسایی و بررسی رفتار راننده بصورت برونو خط استفاده می‌شود. از آنجاییکه بررسی و پردازش کامل و صحیح داده‌ها هدف اصلی پروژه بوده است، از انتقال جریان داده‌های ضبط شده

به مرکز پردازشی جهت پردازش برخط استفاده نشده است تا با صرف نظر نمودن از سربار انتقال داده بر بستر اینترنت، بر مسئله پردازش داده‌ها تمرکز شود. سپس، لیست رخدادها از ورودی‌های سامانه (داده‌های ثبت شده) بدست خواهد آمد و در نهایت نوع و رفتار راننده مورد بررسی قرار می‌گیرد.

## ۲-۴ کارهای آینده

یکی از عوامل تاثیرگذار در عملکرد الگوریتم‌های مبتنی بر یادگیری ماشینی، داده‌ها هستند. بدلیل محدودیت در قسمت ثبت داده در این پروژه، جمع‌آوری طولانی مدت داده‌های رانندگی در مکان‌های متفاوت و با رانندگان مختلف تاثیر بسزایی بر دقت آن خواهد گذاشت. ترکیب حسگرهای مختلف دیگر مانند دوربین، گذرگاه CAN، و داده‌های مربوط به شخص راننده احتمالاً می‌تواند در تشخیص رخدادهای بیشتر و کلیتبخشی سامانه تاثیر گذار باشد.

استفاده از لیست رخدادهای بدست آمده جهت ساخت نمایه‌ای برای هر راننده و شخصی‌سازی بررسی رفتار راننده نیز از مواردی است که می‌توان انجام داد. بررسی‌های انجام شده نشان می‌دهد رانندگان بر اساس نوع انجام رخدادهای رانندگی قابل تمیز دادن با یکدیگر هستند. ویژگی‌های مرتبط با رخداد شتاب تاحدی بین رانندگان مشابه است، اما ویژگی‌های مربوط به چرخش و ترمز نقش قابل توجهی در تشخیص رانندگان ایفا می‌کنند. طبق مقاله [۱۷]، رخداد ترمز بعنوان مهم‌ترین عامل متمایز‌کننده رانندگان اعلام شده است. یکی از دلایل این امر، جابه‌جایی پا برای فشردن پدال ترمز می‌باشد که حرکات متفاوتی را بین رانندگان ایجاد می‌نماید. ایجاد نمایه به ازای هر راننده، می‌تواند امکانات پیشرفته‌تری را برای آن‌ها بوجود آورد. بعنوان مثال، در زمان ناوبری، خودرو می‌تواند نوع جاده مورد ترجیح راننده را پیشنهاد دهد. تحلیل رفتار راننده در چنین سطحی می‌تواند به بررسی رانندگی در قسمت‌های مختلف یک جاده نیز منجر شود. بعنوان مثال، اگر در یک تقاطعی ترمز ناگهانی رخ دهد، خودرو هنگام نزدیک شدن به تقاطع در زمان بعد نسبت به آن وضعیت هشدار می‌دهد.

در این پروژه، برای ارزیابی رفتار راننده، ارتباط بین رخداد انجام شده و هندسه جاده و موقعیت خودرو در جاده درنظر گرفته نمی‌شود. در حالیکه درنظر گرفتن موقعیت خودرو نسبت به جاده‌ای که در آن حرکت می‌کند احتمالاً بتواند دقت تشخیص رخدادها و همچنین تحلیل رفتار راننده را افزایش دهد.

با پیشرفت علوم در حوزه تحلیل داده و الگوریتم‌های یادگیری ماشینی، مسلم‌اً می‌توان از روش‌های نوین مانند استفاده از یادگیری تقویتی در دسته‌بندی، معماری‌های بهینه‌تر در یادگیری عمیق، و ... برای توسعه چنین سامانه‌هایی استفاده نمود. با استفاده درازمدت راننده، داده‌های ارزشمندی از نحوه رانندگی در شرایط مختلف بدست می‌آید که به شناسایی و دسته‌بندی آن‌ها کمک شایانی خواهد کرد.

# منابع و مراجع

- [1] Adafruit's 9-dof (9 degrees of freedom) breakout board. <https://learn.adafruit.com/adafruit-9-dof-imu-breakout/>. Accessed: 2020-07-10.
- [2] Arduino website. <https://www.arduino.cc/>. Accessed: 2020-07-10.
- [3] Caffe. <https://caffe.berkeleyvision.org/>. Accessed: 2020-07-10.
- [4] Connecting adafruit's 9-dof to the arduino uno board. <https://learn.adafruit.com/adafruit-9-dof-imu-breakout/connecting-it-up>. Accessed: 2020-07-10.
- [5] Cs231n: Convolutional neural networks for visual recognition course, stanford university. <https://cs231n.github.io/neural-networks-1/>. Accessed: 2020-02-10.
- [6] Driver behavior dataset. <https://github.com/jair-jr>. Accessed: 2020-02-19.
- [7] Keras. <https://keras.io/>. Accessed: 2020-07-10.
- [8] Pytorch. <https://pytorch.org/>. Accessed: 2020-07-10.
- [9] scikit-learn. <https://scikit-learn.org/>. Accessed: 2020-07-10.
- [10] Sensorlog application official website. <http://sensorlog.berndthomas.net/>. Accessed: 2020-02-19.
- [11] Tensorflow. <https://www.tensorflow.org/>. Accessed: 2020-07-10.

- 
- [12] Understanding lstm networks. <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>. Accessed: 2020-02-10.
- [13] Alluhaibi, Sarah Kadhim, Al-Din, Munaf S Najim, and Moyaid, Aiman. Driver behavior detection techniques: a survey. *Int. J. Appl. Eng. Res*, 13(11):8856–8861, 2018.
- [14] Bergasa, Luis M, Almería, Daniel, Almazán, Javier, Yebes, J Javier, and Arroyo, Roberto. Drivesafe: An app for alerting inattentive drivers and scoring driving behaviors. in *2014 IEEE Intelligent Vehicles symposium proceedings*, pp. 240–245. IEEE, 2014.
- [15] Ferreira, Jair, Carvalho, Eduardo, Ferreira, Bruno V, de Souza, Cleidson, Suhara, Yoshihiko, Pentland, Alex, and Pessin, Gustavo. Driver behavior profiling: An investigation with different smartphone sensors and machine learning. *PLoS one*, 12(4):e0174959, 2017.
- [16] Hochreiter, Sepp and Schmidhuber, Jürgen. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.
- [17] Van Ly, Minh, Martin, Sujitha, and Trivedi, Mohan M. Driver classification and driving style recognition using inertial sensors. in *2013 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1040–1045. IEEE, 2013.
- [18] Wang, Yan, Chen, Yingying Jennifer, Yang, Jie, Gruteser, Marco, Martin, Richard P, Liu, Hongbo, Liu, Luyang, and Karatas, Cagdas. Determining driver phone use by exploiting smartphone integrated sensors. *IEEE Transactions on Mobile Computing*, 15(8):1965–1981, 2015.
- [19] Wu, Minglin, Zhang, Sheng, and Dong, Yuhan. A novel model-based driving behavior recognition system using motion sensors. *Sensors*, 16(10):1746, 2016.