

گزارش تمرین اول درس داده‌کاوی

**«داده‌کاوری یک مسابقه در Kaggle: تایتانیک»**

گردآورنده: سعید دادخواه

استاد: دکتر ناظرفرد

بهمن ۱۳۹۵

# مقدمه

غرق شدن کشتی تایتانیک یکی از بزرگ‌ترین فاجعه‌های تاریخ بوده است. در این مسابقه اطلاعاتی در مورد مسافران این کشتی داده شده است و در مورد برخی از آن‌ها که داده‌های آموزش را تشکیل می‌دهند نجات یافتن آن‌ها یا غرق شدن آن‌ها نیز در این حادثه ذکر شده است. داده‌هایی که اطلاعات مربوط به نجان یافتن افراد را ندارند داده‌های سنجش هستند و دقت الگوریتم نیز با پیش‌بینی نجات یافتن یا غرق شدن آن‌ها سنجیده می‌شود. در این گزارش که مرحله به مرحله یا توجه به آموزش با زبان R موجود در توضیحات مسابقه آورده شده است مراحل و چگونگی پیش‌بینی این موضوع توضیح داده شده است.

نام کاربری saeiddadkhah (<https://www.kaggle.com/saeiddadkhah>) است و الگوریتم Random Forest در محیط R بهترین عملکرد را با دقت ۰.۷۹۹۰۴ داشت.

# گزارش کد محیط R

## به دست آوردن داده‌ها

داده‌ها در صفحه مسابقه و در بخش داده‌های آن (<https://www.kaggle.com/c/titanic/data>) قابل دسترسی هستند. دو فایل csv موجود در این بخش با نام‌های train.csv و test.csv باید دانلود شوند. برای این که این داده‌ها را بتوان در سایت Kaggle خواند باید آن‌ها را در پوشه قبل از پوشه کد در پوشه جدید به نام input (“../input/train.csv” و “../input/test.csv”) قرار داد.

برای پیش‌بینی باید ساختار داده‌های آموزش و سنجش یکسان باشد به همین دلیل هرکاری که روی داده‌های آموزش انجام می‌گیرد باید روی داده‌های سنجش نیز انجام گیرد. برای سادگی کار این دو دسته را با هم ترکیب می‌کنیم. برای این کار از تابع bind\_rows استفاده می‌کنیم که یکی از توابع پکیج dplyr است.

## مرحله اول مهندسی ویژگی‌ها

### عنوان مسافران

ابتدا عنوان مسافران را با استفاده از عبارت منظم استخراج می‌کنیم سپس تعداد افراد را در یک جدول نسبت به جنسیت و عنوان قرار می‌دهیم. سپس گروه‌هایی که تعداد کمی دارند را با هم ترکیب می‌کنیم.

### نجات یا غرق شدن اعضای خانوانده با هم

ابتدا یک ویژگی جدید برای مسافران با عنوان اندازه خانواده درست می‌کنیم که بر اساس ویژگی‌های قبلی ساخته می‌شود. یک ویژگی دیگر نیز بر همین پایه می‌سازیم که به صورت فاکتور خواهد بود. افرادی که اندازه خانواده‌شان در کشتی برابر یک، دو تا چهار و بیش از چهار نفر به ترتیب به صورت تنها، کوچک و بزرگ برچسب می‌خورن.

### عرشه

با استفاده از ویژگی کابین افراد عرشه آن‌ها مشخص می‌شود. حرف اول کابین عرشه مسافر را مشخص می‌کند.

## پر کردن داده‌های ناقص

### Embarked

ابتدا ویژگی Embarked را مورد بررسی قرار می‌دهیم. در این ویژگی دو رکورد فاقد مقدار هستند. برای پر کردن این مقادیر رکوردهای دیگر را در یک نمودار جعبه‌ای نمایش می‌دهیم. در این نمودار برای هر دسته از مقادیر Embarked و Pclass یک جعبه و جعبه‌ها را بر اساس Fare رسم می‌کنیم. در نمودار به‌دست آمده کاملا قابل مشاهده است که میانگین Fare داده‌هایی که Pclass آن‌ها برابر با مقدار این ویژگی در این دو داده ناقص است مقدار Embark در آن‌ها برابر ‘C’ است پس نقص این داده‌ها را اینگونه رفع می‌کنیم.

### Fare

در این ویژگی فقط یک نقص وجود دارد. برای رفع این ویژگی هیستوگرام داده‌هایی که Pclass و Embarked برابر داده ناقص دارند را رسم می‌کنیم. با توجه به نمودار این مقدار را برابر با مقدار میانه در این داده‌ها قرار می‌دهیم.

### Age

در این ویژگی ۲۶۳ داده ناقص وجود دارد. برای رفع این نقص از پکیج MICE (Multivariate Imputation by Chained Equations) استفاده می‌کنیم. برای این کار دو ویژگی دیگر که یکی نام خانوادگی و دیگر نام خانواده است را نیز به داده‌ها اضافه می‌کنیم و روش rf برای این کار استفاده می‌شود. در نهایت ستون Age از خروجی mice جایگزین ستون اصلی Age می‌شود.

## مرحله دوم مهندسی ویژگی‌ها

### کودکان و مادرها

در این مرحله مسافرانی که سنی کمتر از ۱۸ دارند با عنوان کودک و بقیه با عنوان بزرگسال برچسب می‌خورند و زنانی که فردی با رابطه والد-فرزندی داشته باشند، سنشان بیشتر از ۱۸ باشد و عنوانشان بانو نباشد با عنوان مادر برچسب می‌خورند.

## پیش‌بینی

در این مرحله ابتدا داده‌های آموزشی از داده‌های سنجش جدا می‌شوند سپس یک randomForest آموزش داده می‌شود. برای نمایش اهمیت هرکدام از ویژگی‌ها یک نمودار رسم می‌شود که اهمیت هرکدام از ویژگی‌ها را نمایش می‌دهد. در نهایت نیز داده‌های سنجش نیز به عنوان ورودی به الگوریتم داده می‌شود و خروجی آن‌ها در یک data frame ذخیره و در یک فایل csv به عنوان خروجی قرار داده می‌شوند تا در سایت Kaggle مورد ارزیابی قرار گیرند. در نهایت دقت ۰.۷۹۹۰۴ برای خروجی این کد ثبت شد.

# گزارش کد محیط پایتون

در این کد ابتدا پکیج‌های مورد نیاز اضافه می‌شوند سپس مقادیری برای کنترل روند کلی کد تعیین می‌شوند. برنامه در سه فاز انجام می‌پذیرد. ابتدا داده‌ها خوانده می‌شوند سپس ویژگی‌های مناسب از آن استخراج می‌شوند و در نهایت مدل‌هایی آموزش داده می‌شوند و مقایسه می‌شوند.

## مهندسی ویژگی‌ها

برای انجام این کار در هر قسمت تعدادی نمودار هم رسم می‌شود که در گزارش آورده نمی‌شود و برای مشاهده باید کد اجرا شود.

### Embarked

هدف در این قسمت این است که هر گروه را به یک ستون تبدیل کنیم یعنی در مجموع سه ویژگی اضافه خواهد شد که در هر داده فقط یکی از این‌ها برابر یک خواهد بود. یکی از گروه‌ها با فرض به عنوان گروه پایه حذف می‌شود و داده‌هایی که این ویژگی از آن‌ها دارای مقداری نیست را نیز به گروه پیشفرض اضافه می‌کنیم.

### Fare

تعدادی از داده‌ها وجود دارند که مقدارشان برای این ویژگی نامعلوم است ابتدا برای پر کردن آن‌ها از میانه باقی داده‌ها استفاده می‌کنیم.

### Age

برای پر کردن مقادیر گم شده در این ویژگی میانگین و انحراف معیار باقی داده‌ها را می‌یابیم و مقادیر آن‌ها را برابر مقادیری تصادفی در فاصله یک انحراف معیار از میانگین قرار می‌دهیم.

### Cabin

به کلی از این ویژگی صرف نظر می‌کنیم.

### Family

یک ویژگی برای نمایش این که آیا یک فرد همراه با خانواده در کشتی است یا خیر می‌سازیم. ویژگی‌های Parch و SibSp که به ترتیب نشان دهنده تعداد والد-فرزند و برادر-خواهر در کشتی هستند برای ساختن ویژگی جدید استفاده می‌شوند. ابتدا حاصل جمع این دو مقدار در ستون جدید نگه می‌داریم، سپس در نهایت در صورتی که مقدار غیر صفر داشتند آن‌ها را به مقدار یک تغییر می‌دهیم.

### Sex

در این قسمت سه ویژگی کودک، مونث و مذکر را در نظر می‌گیریم. سه ویژگی که هرکدام نشان دهنده هرکدام از این ویژگی‌ها هستند را می‌سازیم، برای هر داده فقط مقدار یکی از این ویژگی‌ها یک و بقیه صفر خواهند بود. برای این کار ابتدا یک ویژگی جدید می‌سازیم که سه مقدار کودک، مونث و مذکر می‌تواند داشته باشد. برای مقداردهی به این ویژگی ابتدا سن افراد بررسی می‌شود که اگر کمتر از شانزده بود مقدار کودک برای آن‌ها ثبت شود در غیر این صورت جنیست آن‌ها برای آن‌ها ثبت می‌شود. سپس این ویژگی به سه ویژگی جداگانه تبدیل می‌شود. در نهایت ستون مذکر به عنوان حالت پیشفرض در نظر گرفته شده و حذف می‌شود.

### Pclass

این ویژگی دارای سه مقدار متفاوت است پس آن را تبدیل به سه ویژگی می‌کنیم. پس این ویژگی را به سه ویژگی تبدیل می‌کنیم که اگر یک داده مقدار یک، دو یا سه را داشت فقط ویژگی مربوط به آن مقدار یک و دو ستون بعدی مقدار صفر خواهند گرفت.

### آماده‌سازی برای یادگیری

فرایندهای فوق روی داده‌های خوانده شده از فایل داده‌های آموزشی و آزمایش انجام شدند. برای آماده کردن ویژگی‌های داده‌های آموزشی و برچسب ‌آن‌ها همه ستون‌های داده‌های آموزشی به جز ستون Survived را به عنوان ویژگی‌ها و ستون Survived را به عنوان برچسب آن‌ها در نظر می‌گیریم. چون داده‌های آزمایش برچسب نداشتند می‌توانند مستقیما به عنوان داده‌های آزمایشی مورد استفاده قرار بگیرند.

## آموزش الگوریتم‌ها و سنجش

الگوریتم‌های Logistic Regression، Support Vector Classifier، Linear SVC، Random Forest، K-Nearest Neighbor و Gaussian Naïve Bayes با داده‌ها آموزش داده شدند و با استفاده از تابع score عملکرد آن‌ها سنجیده شد که نتایج به شکل زیر هستند.

|  |  |
| --- | --- |
| الگوریتم | دقت |
| Logistic Regression | ۰.۸۱۹۳۰ |
| Support Vector Classifier | ۰.۸۵۷۴۶ |
| Linear SVC | ۰.۷۸۶۷۵ |
| Random Forest | ۰.۹۵۳۹۸ |
| K-Nearest Neighbor | ۰.۷۹۱۲۴ |
| Gaussian Naïve Bayes | ۰.۸۰۱۳۴ |

در نهایت خروجی الگوریتم Random Forest به عنوان خروجی نهایی در سایت Kaggle ثبت شد که دقت آن ۰.۷۲۷۲۷ اعلام شد.

# سوال‌ها

۱. DataFrame چیست؟

DataFrameها ساختمان داده‌هایی برای نگه‌داری داده‌هایی با طول برابر به صورت جدول هستند. در این ساختمان داده هر ستون می‌تواند header و هر سطر یک نام داشته باشد.

۲. مقصود از Imputation چیست؟

داده‌هایی که برای پیش‌بینی استفاده می‌شوند کامل نیستند و تعدادی از ویژگی‌هایشان مقدار ندارند. برای پیش‌بینی بهتر ابتدا باید این نقص‌ها را برطرف کرد. این کار را Imputation می‌نامند.

۳. Normalize کردن داده‌ها به چه منظوری انجام می‌گیرد؟

در صورتی که داده‌ها نرمال نشده باشند می‌توانند روی عملکرد الگوریتم تاثیر بگذارند. برای مثال ممکن است الگوریتم نیاز به زمان بیشتری برای همگرایی داشته باشد. حتی ممکن است عملکرد الگوریتم نیز تضعیف شود.

۴. آیا در درخت تصمیم، تمام ویژگی‌ها دارای اهمیت یکسانی هستند؟

خیر، اهمیت ویژگی‌ها در الگوریتم‌ها متفاوت است و در گزارش فوق نیز یک نمودار برای نمایش اهمیت ویژگی‌ها رسم شد.

۵. در الگوریتم جنگل تصادفی دو پارامتر max\_depth و min\_sample\_split بیانگر چه چیزی هستند؟

پارامتر max\_depth عمق درخت (در واقع جنگل) را محدود می‌کند و زمانی که عمق به حد مشخص شده رسید دیگر الگوریتم از تصمیم‌گیری در آن نقطه صرف نظر می‌کند و از رای نمونه‌های موجود در آن برگ برای تعیین کلاس نمونه استفاده می‌کند. پارامتر min\_sample\_split تعیین می‌کند که حداقل تعداد نمونه‌ها در یک برگ باید چند باشد تا تصمیم‌گیری دیگری انجام نگیرد و از رای نمونه‌های موجود در آن برگ برای تعیین کلاس نمونه استفاده شود.