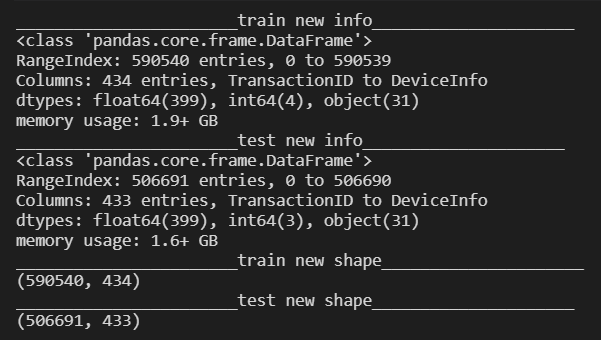
ایده اصلی و توضیحات کلی :

ایده کلی بنده برای این پروژه بر این قرار بود که یک شبکه اتو انکودر را فقط با داده های غیر جعلی آموزش داده, اینکار باعث میشود تا اتوانکودر با استفاده از فضای میانه خود بتواند داده های غیر جعلی را باز سازی کند (با توجه به صحبت های آقای شریفی در کلاس حل تمرین مربوط به اتو انکودر)

اتو انکودر این کار را با وجود یک خطا انجام میدهد در اصل هدف ما در آموزش کم کردن این خطا است

بعد از آموزش این شبکه ما قادر خواهیم بود داده های جعلی را به خورد آن دهیم و مشاهده میکنیم که خطای بالای بازسازی داده ها نشان میدهد داده ها جعلی است

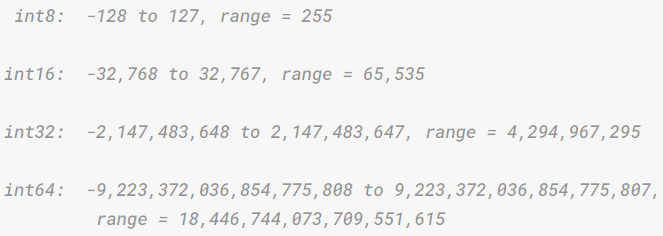
توضیحات دقیق :

اول باید دیتا ها را قابل استفاده کنیم در ابتدا اگر دقت کنیم دیتا ها حجم بسیاری از رم را اشغال میکنند که بهینه نیست

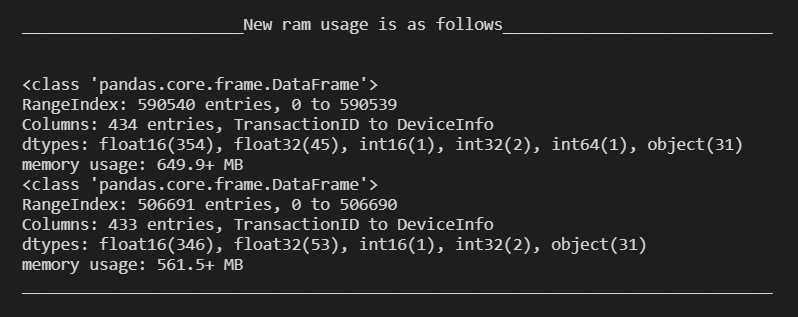
در عکس بالا مقدار اشغال شده را میبینیم

برای اینکار باید تایپ دیتا ها را عوض کرده و از float64 به تایپ های کم هزینه تری انتقال دهیم چرا که نیازی به وجودش نیست (با توجه به رنج مقادیر هر تایپ و ماکسیمم ستون مورد نظر)

مقادیر رنج هر تایپ برابر است با :

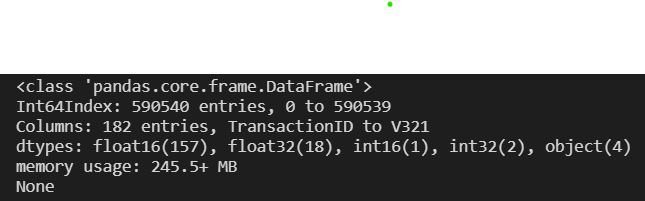


برای ستون های ما float16 وfloat 32 کفایت میکند

که بعد از تغییر تایپ ها خواهیم داشت :

که میبینیم حدود 2 گیگ از رم را آزاد کرده است

حالا دیتا های ما قابل استفاده تر هستند (یک مرحله دیگر نیز برای کم حجم کردن داده ها انجام شد که شامل حذف تعدادی از ستون ها بود که خیلی در روند برنامه تغییری ایجاد نکرد و با توجه به حجم رم کامپیوتر بنده که 16 گیگ بود نیازی به آن نشد و ترجیح بر تعداد ستون های بیشتر بود برای دقت بالاتر نتایج آن را در زیر مشاهده میکنیم

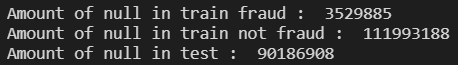


ستون هایی که حذف شد عموما ستون هایی از فال دوم بودند که مقدار زیادی از مقادیرشان در فایل اکسل خالی بود)

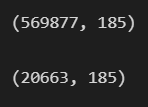
برای استفاده از داده ها در شبکه نیاز به این بود که ابتدا مقادیر خالی را در داده ها جایگزین با یک مقدار عددی کنیم

برای اینکار بنده از هر ستون مد گرفته و تمام مقادیر خالی را با آن جایگزین کردم

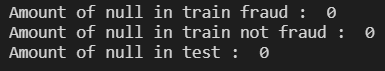
تعداد مقادیر خالی قبل از اینکار را در زیر میبینیم



برای اینکار البته باید توجه داشت که نباید مقادیر جعلی و غیر جعلی در یک فریم یکسان باشند چراکه تعداد زیاد داده های غیر جعلی مد خود را بر داده های جعلی تحمیل میکنند



برای این منظور داده های جعلی و غیر جعلی جدا سازی شده تا این مشکل رفع شود در شکل روبرو خط اول تعداد داده های غیر جعلی و خط دوم تعداد داده های جعلی است



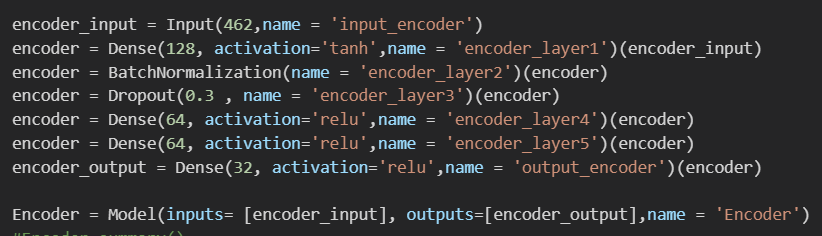
حال میبینیم که داده های خالی دیگر وجود ندارند حالا میتوانیم از داده های برای آموزش شبکه استفاده کنیم

برای نبدیل داده های کیفی به کمی نیز از labelEncoder در کتابخانه sklearn استفاده شده است (مجبور به label encoding بودیم زیرا تعداد ستون ها زیاد بود)

تنها کاری که باقیمانده است این است که داده ها را نرمال کرده

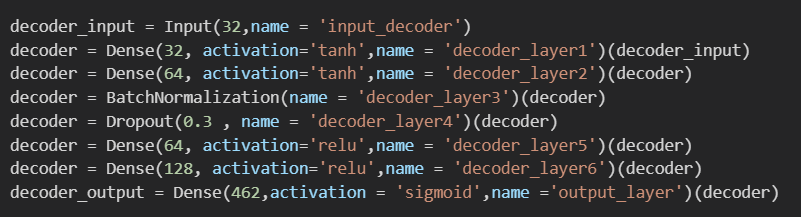
قابل به ذکر است که این کار خود به کم شدن حجم داده ها کمک میکند

حال تنها کاری که لازم است این است که یک آتو انکودر بسازیم و داده های غیر جعلی را به آن فیت کنیم تا یاد بگیرد داده های غیر جعلی را از نو بسازد



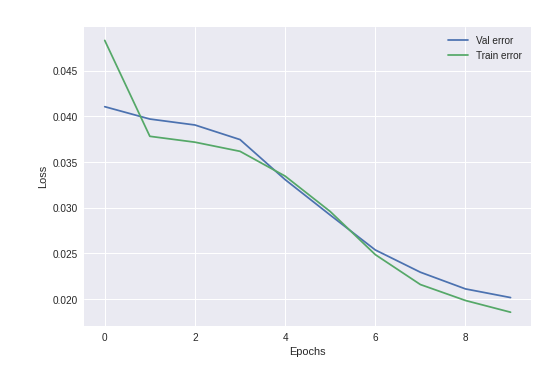
تصویر بالا ساختار قسمت اول اتو انکودر را نشان میدهد که در شکل واضح است

لایه ورودی با 462 ستون که تعداد فیچر های ماست 3 لایه fully connected با عمق های 128 و 64 و 64 و لایه اخر که با عمق 32 همان فضای میانه است



شکل بالا نیز قسمت دوم آتوانکودر که دیکودر آن نام دارد ساختار استفاده شده دقیقا برابر ساختار انکودر است

حال به آموزش نتورک میپردازیم



در شکل بالا خطای آتوانمودر با mae است

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

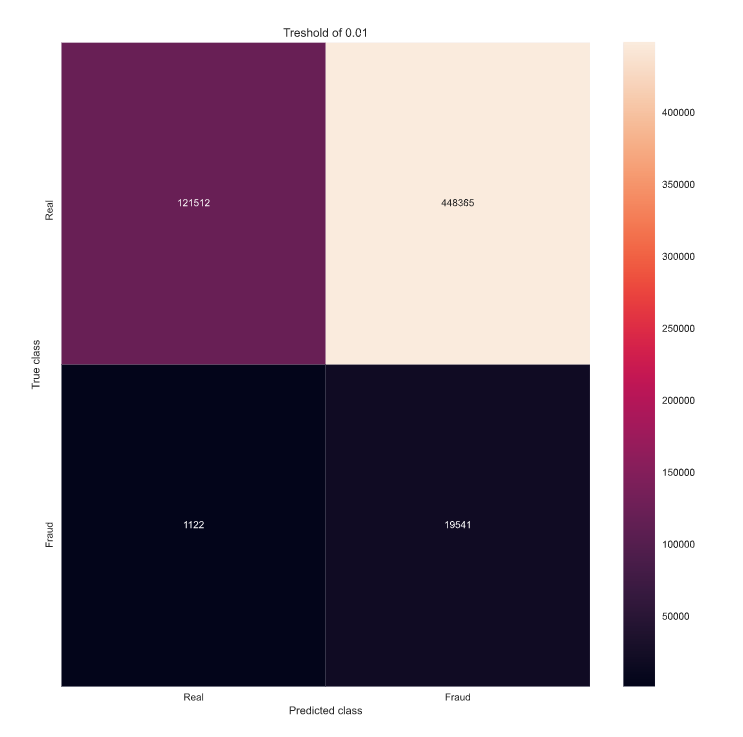
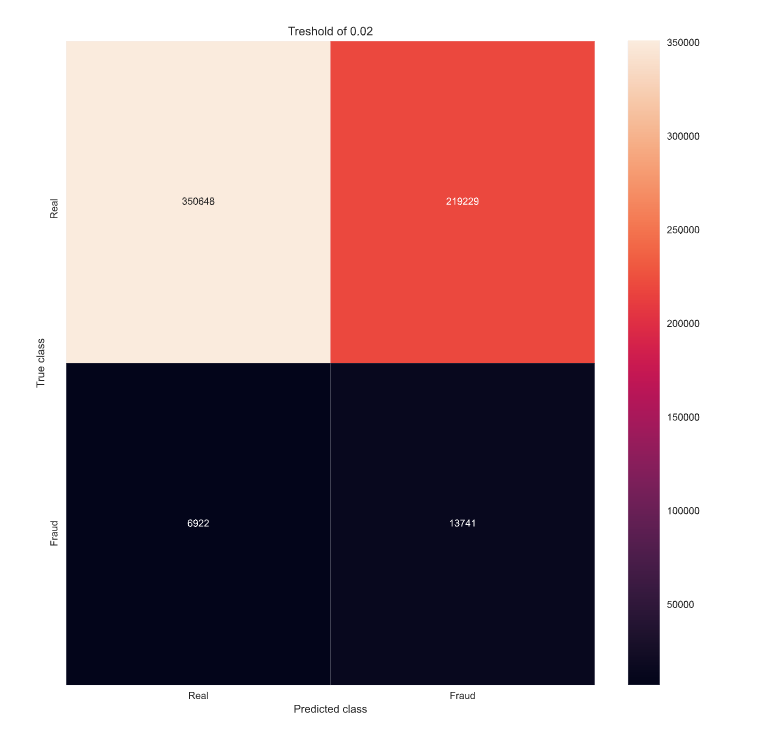
اتونکودر در حال حاضر میتواند با خطای نسبتا قابل قبولی داده های غیر جعلی را باز سازی کند

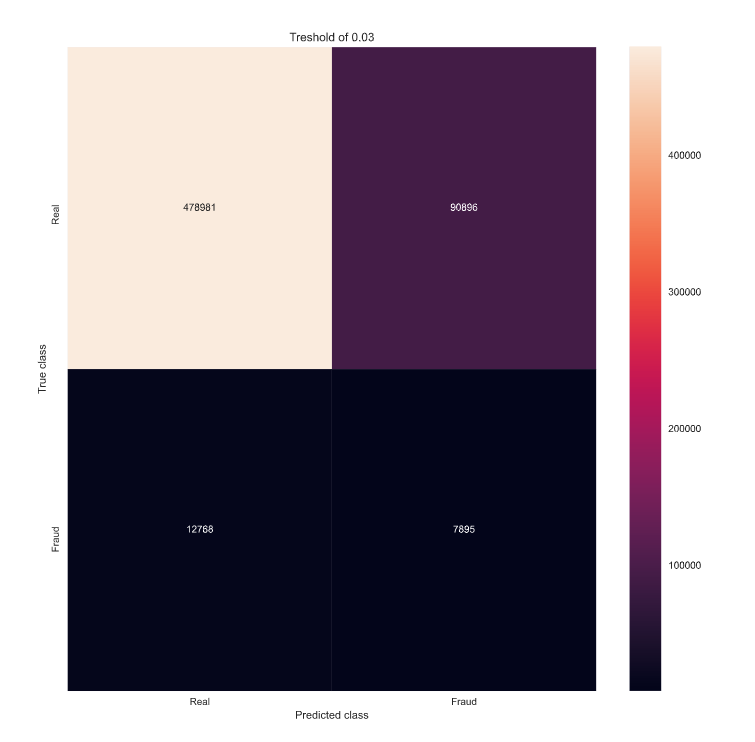
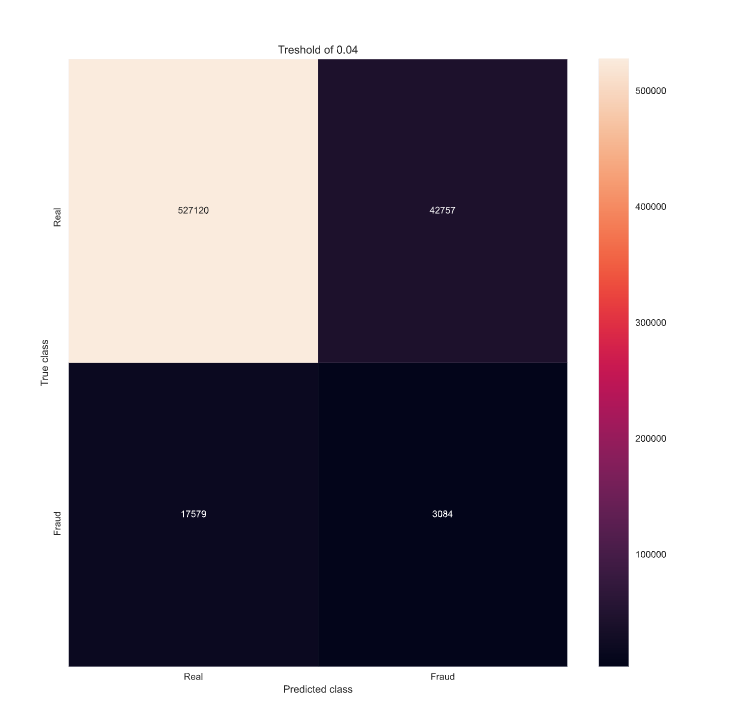
حال کاری که میکنیم این است که داده های غیر جعلی را با این اتوانکودر predict میکنیم

و یک لیست از مقادیر پیش بینی شده میسازیم

سپس مقادیر را از مقادیر اصلی کم میکنیم و از آن قدر مطلق میگیریم

از اینجا به بعد به یک متغیر نیاز خواهیم داشت treshold تا اگر خطای ما از این مقدار فیکس بیشتر شد آن را داده جعلی بنامیم

با 4 مقدار مختلف از threshold جواب ها را میبینیم



بنظر میاید trshold با مقدار 0.03 متعادل ترین جواب را میسازد

یک راه دیگر نیز برای حل این پروژه امتحان شد که کلیت آن به شرح زیر است

طبق صحبت های آقای شریفی در کلاس حل تمرین فضای میانه ای که در آتو انکودر ساخته میشود حامل چکیده هر ورودی است

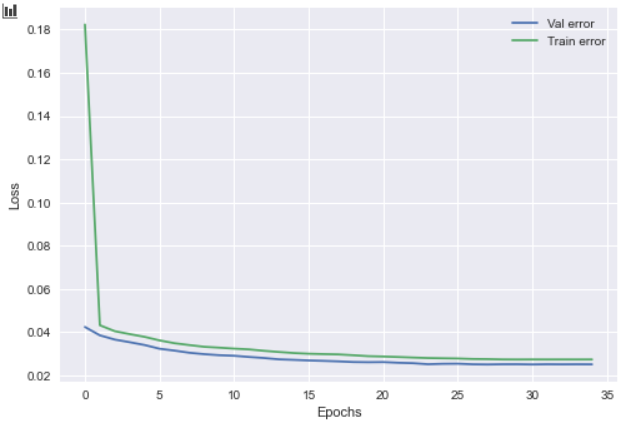
میتوانیم آتوانحکودر را آموزش داده و بعد از آموزش آن یک شبکه عادی بسازیم

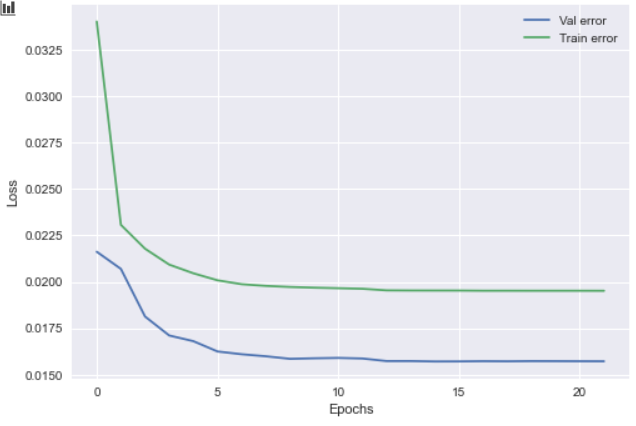
سپس بجای ورودی دادن هر 436 ستون فضای میانه مربوط به آن ورودی را به شبکه عادی بدیم و بگذاریم شبکه عادی ما ککه خروجی با عمق 1 دارد مقدار isFraud را پیشبینی کند

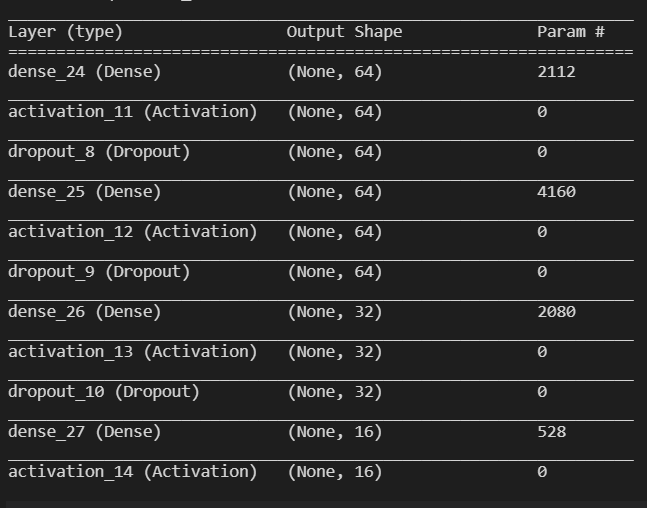
برای اینکار دو بار آتوانکودر را میسازیم و آموزش میدهیم یک بار با داده های جعلی یک بار هم با داده های غیر جعلی (زیرا میخواهیم اتوانکودر ما بتواند یک بار جعلی ها را چکیده کند و یک بار غیر جعلی ها را)

نتیجه دو بار اموزش دادن به شکل زیر است

غیر جعلی :



غیر جعلی ( در این دفعه که خطا از راه اولکمتر است دلیلش تغیر کوچکی در ساختار شبکه بود (پیچیده تر شد))

و البته ساختار شبکه عادی ما که به شکل رو برو میباشد

همه لایه های فعال ساز Relu هستند به جز اهرین لایه فعال ساز که linear است

قابل ذکر نیز است که بگویم برای این روش باید تعداد داده هایمان از هر کلاس تقریبا برابر  
باشد و ازآنجایی که داده های ما Sparseبوده و توضیع بسیاز نا متوازنی داشتند میبایستی  
ابتدا تعداد ورودی هایمان را یکی کنیم

برای اینکار میتوان از این دو روش استفاده کرد :

:upSampling – 1به این معنی که تعداد داده های جعلیمان را تا مقدار داده های غیر  
جعلی باز سازی کنیم  
برای این کار میتوان از داده های موجود داده های دیگر را الگو سازی کرد و ساخت  
ایده ای که داشتم این بود که بعد از آموزش یک اتو انکودر که داده های جعلی را بازسازی  
میکرد, داده های جعلی را چکیده کنیم ( latent vectorانها را بدست اوریم) و با تغییرات  
بسیار کوچکی در latent vectorدوباره آن ها را به Decoderداده و بازسازی کنیم

:downSampling – 2در این روش بجای اینکه داده های جعلی را به اندازه داده های  
غیر جعلی برسانیم  
از داده های غیر جعلی یک نمونه تصادفی با اندازه داده های جعلی میگیریم (حدود 24  
هزار تا) و از آن نمونه برای ادامه کار استفاده میکنیم

اما مشکلی که downSamplingدارد این است که به دلیل حجم کم داده های جعلی ما  
ممکن است شبکه قابلیت generalبودن خود را از دست بدهد  
البته متاسفانه برای پیاده سازی ایده ای که برای upSamplingگفتم وقت تنگی داشتم و  
موفق به پیاده سازی آن نشدم و از downSamplingاستفاده کردم

بعضی از قسمت های کد (قسمت به کم حجم کردن داده ها و label encoding) را با کمک از تعدادی از notebook هایی که در کگل مشاهده کردم نوشتم