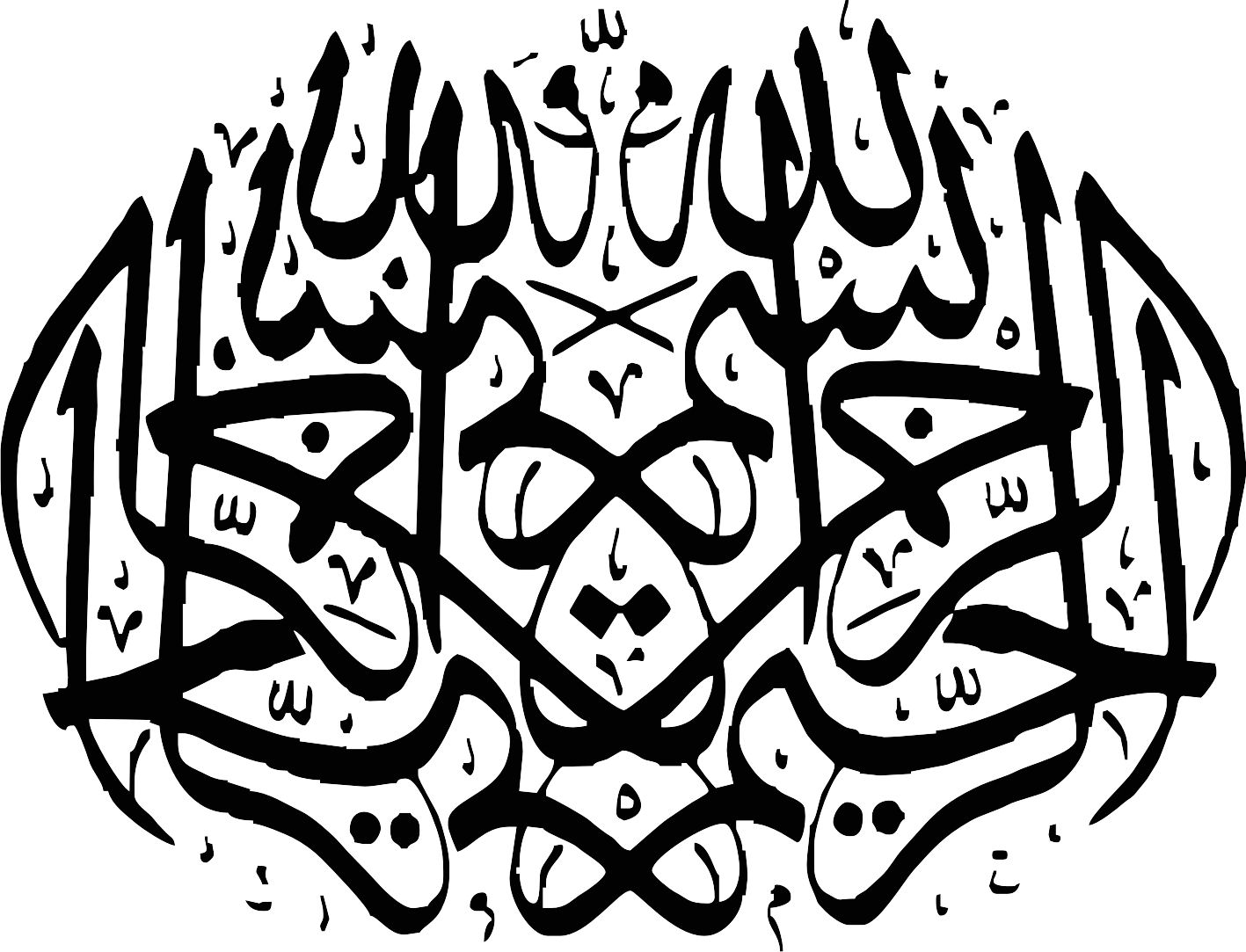


**گزارش تکلیف دوم درس یادگیری ماشین کاربردی**

**استاد درس: دکتر ناظرفرد**

تهیه کننده: سید نیما محمودیان

شماره دانشجویی: 402125005



فهرست مطالب

[1-پاسخ سوال اول 1](#_Toc163208856)

[1-1-لود کردن دیتاست، نمایش ده ردیف به صورت تصادفی، و بررسی کلی داده‌ها 1](#_Toc163208857)

[2-1-مدیریت مقادیر گمشده 5](#_Toc163208858)

[3-1-مصور سازی داده‌ها 8](#_Toc163208859)

[4-1-مدیریت داده‌های پرت 14](#_Toc163208860)

[5-1-مهندسی ویژگی‌ها 16](#_Toc163208861)

[2-پیش‌پردازش تصویر 18](#_Toc163208862)

[1-2- gray scale کردن عکس‌ها 20](#_Toc163208863)

[2-2-تنظیم روشنایی و کنتراست تصاویر 21](#_Toc163208864)

[3-2-نرمال سازی تصاویر 23](#_Toc163208865)

[3-پیش‌پردازش متن 25](#_Toc163208866)

[1-3-خواندن داده‌ها از پیکره همشهری 25](#_Toc163208867)

[2-3-پیش‌پردازش متن‌ها 29](#_Toc163208868)

[3-3-TF-IDF 32](#_Toc163208869)

[4-3-مصورسازی داده‌ها 33](#_Toc163208870)

[4-پیوست 40](#_Toc163208871)

فهرست شکل‌ها

[شکل 1: وارد کردن کتابخانه‌های مورد نیاز 1](#_Toc163166210)

[شکل 2: حذف ستون‌های اضافه 1](#_Toc163166211)

[شکل 3: خواندن دیتاست و نمایش ده سطر تصادفی 2](#_Toc163166212)

[شکل 4: بررسی مجموع مقادیر ناموجود در هر ستون 2](#_Toc163166213)

[شکل 5: اطلاعات نمایش داده شده توسط متد info() 3](#_Toc163166214)

[شکل 6: تعداد مقادیر موجود در هر یک از متغیرهای دسته‌ای 3](#_Toc163166215)

[شکل 7: دسته‌بندی مجدد ستون nationality 4](#_Toc163166216)

[شکل 8: دسته‌بندی مجدد ستون edu 5](#_Toc163166217)

[شکل 9: جایگذاری ؟ با np.nan 6](#_Toc163166218)

[شکل 10: train test split 6](#_Toc163166219)

[شکل 11: پایپ‌لاین پیش پردازش داده‌های دسته‌ای 6](#_Toc163166220)

[شکل 12: نحوه تعریف کلاس ColumnTransformer() 7](#_Toc163166221)

[شکل 13: تعریف مدل و پایپ‌لاین اصلی 7](#_Toc163166222)

[شکل 14: شمای کلی پایپ‌لاین 8](#_Toc163166223)

[شکل 15:معیارسنجی مدل 8](#_Toc163166224)

[شکل 16: حذف سطرهای دارای مقادیر خالی 8](#_Toc163166225)

[شکل 17: کد رسم نمودار ستونی 9](#_Toc163166226)

[شکل 18: کد مربوط به نمودار جعبه‌ای 9](#_Toc163166227)

[شکل 19: نمودارهای ستونی رسم شده از داده‌های دسته‌ای 10](#_Toc163166228)

[شکل 20: نمودار جعبه‌ای از داده‌های عددی 11](#_Toc163166229)

[شکل 21: کد مربوط به هیستوگرام 11](#_Toc163166230)

[شکل 22: هیستوگرام‌های رسم شده 11](#_Toc163166231)

[شکل 23: نحوه انتخاب ستون‌های عددی و محاسبه ماتریس همبستگی 12](#_Toc163166232)

[شکل 24: هیت‌مپ رسم شده 12](#_Toc163166233)

[شکل 25: نحوه محاسبه همبستگی با ستون هدف 13](#_Toc163166234)

[شکل 26: نمودار میله‌ای همبستگی با ستون هدف 13](#_Toc163166235)

[شکل 27: نحوه محاسبه هیت‌مپ برای داده‌های دسته‌ای 14](#_Toc163166236)

[شکل 28: هیت‌مپ ایجاد شده از ستون‌های دسته‌ای 15](#_Toc163166237)

[شکل 29: نحوه حذف سطرهایی که داده پرت دارند 15](#_Toc163166238)

[شکل 30: one hot encoding 16](#_Toc163166239)

[شکل 31: انتقال ستون هدف به انتهای دیتافریم 17](#_Toc163166240)

[شکل 32: پایپ‌لاین ارزیابی روش PCA 17](#_Toc163166241)

[شکل 33: ایجاد کلاس و پایپ‌لاین برای mutual information 18](#_Toc163166242)

[شکل 34: شمای کلی پایپ‌لاین ایجاد شده برای mutual information 18](#_Toc163166243)

[شکل 35: وارد کردن کتابخانه‌ها و تولید سه عدد تصادفی 19](#_Toc163166244)

[شکل 36: لود کردن تصاویر و ذخیره آنها در متغیرها 19](#_Toc163166245)

[شکل 37: نمایش ابعاد عکس‌ها 19](#_Toc163166246)

[شکل 38: تبدیل عکس‌ها به rgb و نمایش آنها 20](#_Toc163166247)

[شکل 39: gary scale کردن عکس‌ها 22](#_Toc163166248)

[شکل 40: تابع تنظیم کنتراست و روشنایی 23](#_Toc163166249)

[شکل 41: تنظیم روشنایی و کنتراست شکل اول 24](#_Toc163166250)

[شکل 42: تابع نرمال کننده عکس 25](#_Toc163166251)

[شکل 43: نرمال کردن عکس با استفاده از تابع 25](#_Toc163166252)

[شکل 44: کتابخانه‌های استفاده شده در سوال سوم 26](#_Toc163166253)

[شکل 45: جداسازی رکوردها و قرار دادن آنها در چهار فیلد 26](#_Toc163166254)

[شکل 46: تابع تقسیم کننده فایل به یازده بخش 27](#_Toc163166255)

[شکل 47: تمیز کردن متن‌ها و ذخیره آنها در اکسل 28](#_Toc163166256)

[شکل 48: تبدیل فایل‌های اکسل به یک دیتافریم 28](#_Toc163166257)

[شکل 49: پیدا کردن نحوه انکودینگ متن 29](#_Toc163166258)

[شکل 50: حذف و جایگذینی کاراکترها و type casting 29](#_Toc163166259)

[شکل 51: حذف علائم نگارشی 30](#_Toc163166260)

[شکل 52: نحوه حذف اعداد 30](#_Toc163166261)

[شکل 53: توکنایز کردن متن 31](#_Toc163166262)

[شکل 54: حذف stop wordsها 31](#_Toc163166263)

[شکل 55: نمایش 5 توکنی که بیشتری استفاده را داشته‌اند 31](#_Toc163166264)

[شکل 56: تبدیل توکن‌ها به متن و نرمال سازی متن 32](#_Toc163166265)

[شکل 57: اجرای TF-IDF 32](#_Toc163166266)

[شکل 58: نمایش 5 کلمه مهم هر متن 33](#_Toc163166267)

[شکل 59: نمایش 5 کلمه مهم در همه متون 33](#_Toc163166268)

[شکل 60: فرایند ایجاد wordcloud 34](#_Toc163166269)

[شکل 61: ابر کلمه تولید شده 35](#_Toc163166270)

[شکل 62: انجام تحلیل عواطف با استفاده از کتابخانه polyglot 36](#_Toc163166271)

[شکل 63: نمودار سری زمانی 37](#_Toc163166272)

[شکل 64: نمودار میله‌ای تعداد متن‌های مثبت، خنثی و منفی 37](#_Toc163166273)

[شکل 65: نمودار سهم هر دسته از قطبیت‌ها از کل متون 38](#_Toc163166274)

[شکل 66: هسیتوگرام میزان قطبیت 38](#_Toc163166275)

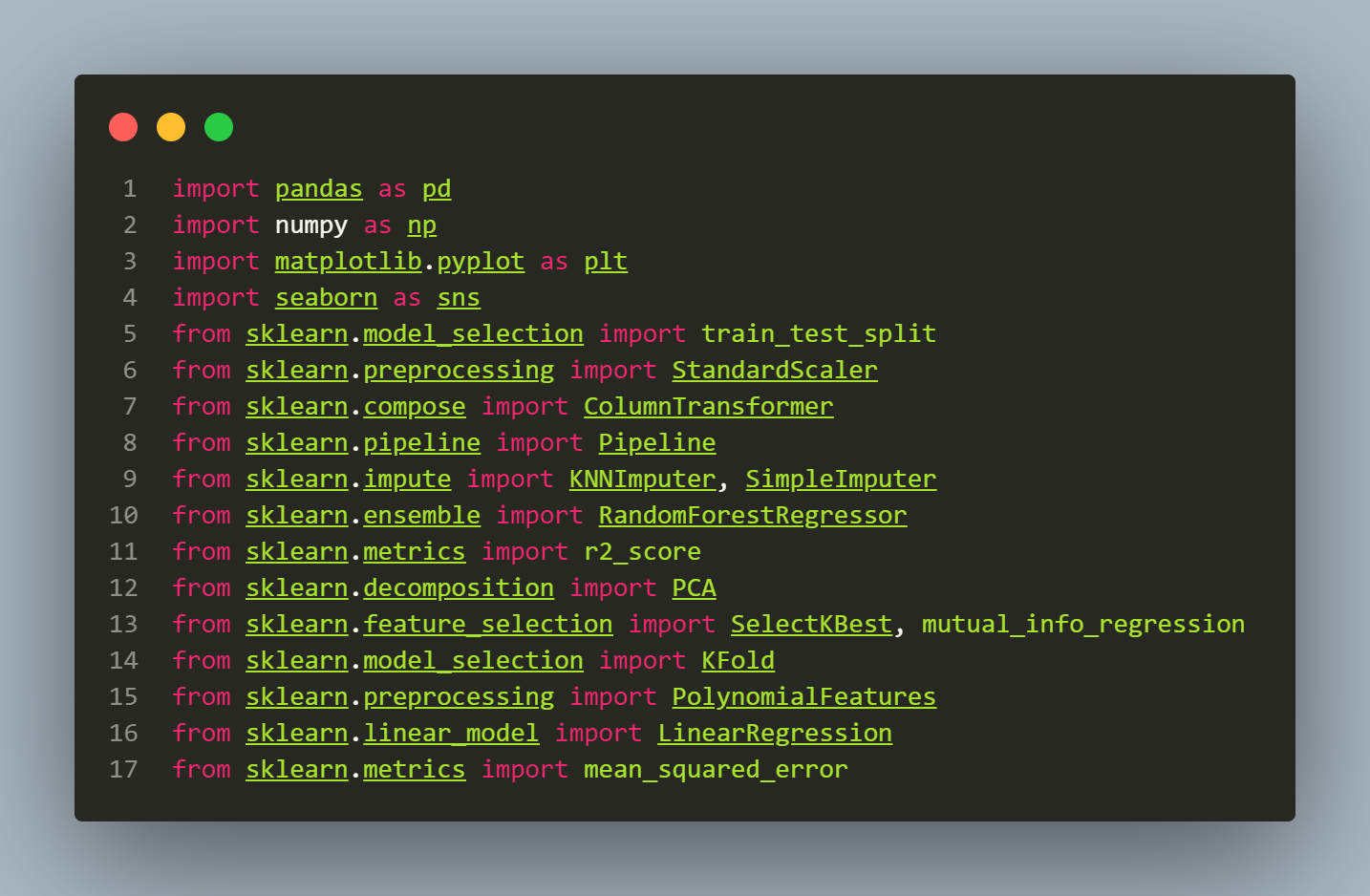
[شکل 67: نمودار تعداد اخبار مثبت، خنثی و منفی در طول زمان 39](#_Toc163166276)

[شکل 68: نمودار میله‌ای تعداد متون در بیست دسته اول موضوعات 40](#_Toc163166277)

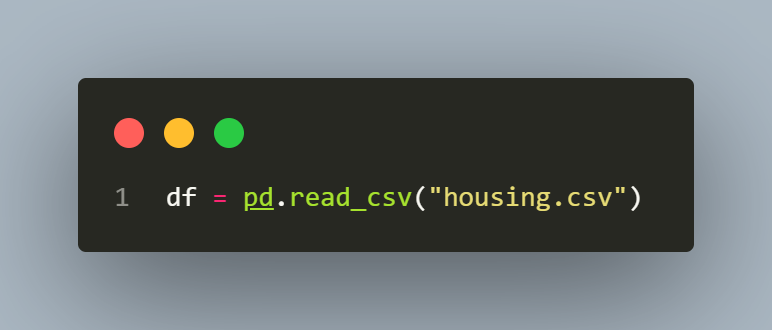
# 1-پاسخ سوال اول

## 1-1-لود کردن دیتاست، نمایش ده ردیف به صورت تصادفی، و بررسی کلی داده‌ها

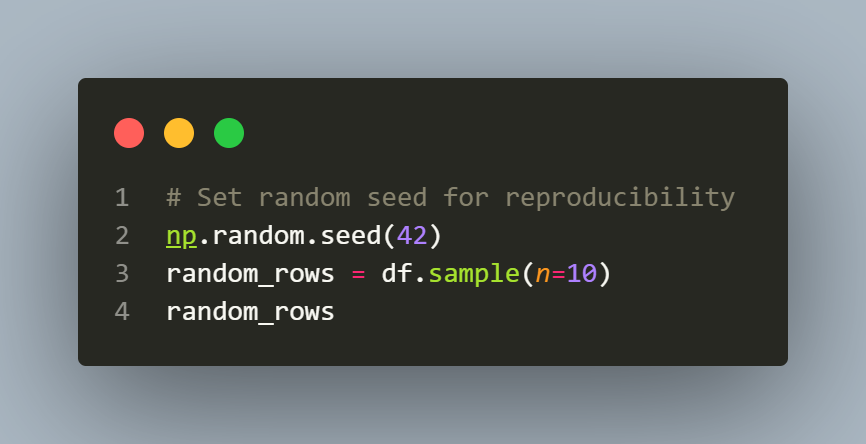
توجه شود که هرجا احتیاجی به ایپمورت کردن یک کتابخانه بود، برای حفظ نظم، ایمپورت در اولین بلوک کد انجام شده است شکل یک بلوک مربوط به وارد کردن کتابخانه‌ها را نمایش می‌دهد. برای لود کردن دیتاست، و نمایش ده ردیف رندوم از کتابخانه pandas استفاده می‌کنیم. با استفاده از کتابخانه numpy یک random seed ایجاد می‌کنیم. Random seed به مظور بازتولید نتایج مشابه تعریف می‌شود. شکل (2)، خواندن دیتاست و شکل (3) نشان دادن ده ردیف رندوم را نمایش می‌دهد.



شکل 1: وارد کردن کتابخانه‌های مورد نیاز

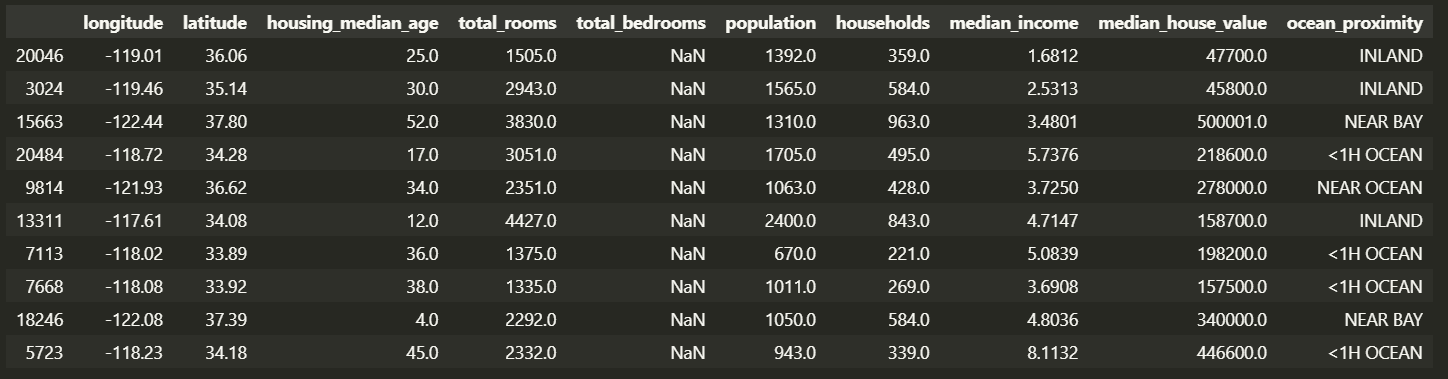


شکل 2: خواندن دیتاست



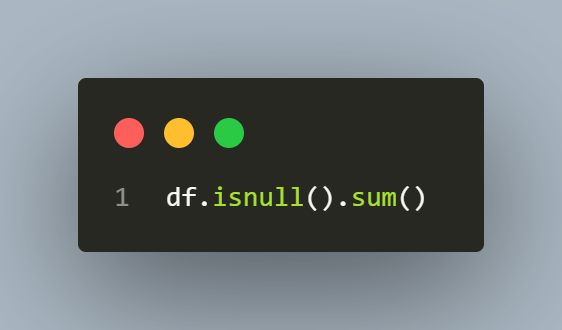
شکل 3: کد مربوط به نمایش ده سطر تصادفی

همچنین شکل (4) ده سطر تصادفی نمایش داده شده را نشان می‌دهد.



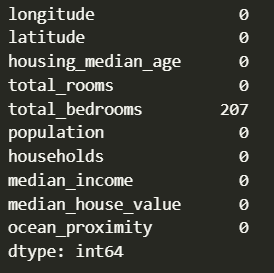
شکل 4: ده سطر تصادفی نمایش داده شده

در ادامه با استفاده از متد info() نگاهی کلی به ستون‌های باقی‌مانده می‌اندازیم. سپس با استفاده از method chaining نمایش داده شده در شکل (5) بررسی می‌کنیم که در هر ستون در مجموع چه تعداد مقادیر تعریف نشده وجود دارد.



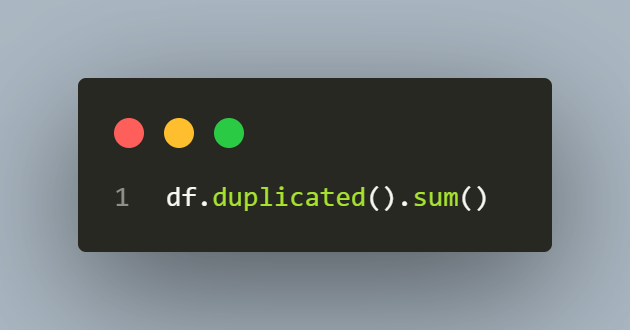
شکل 5: کد مربوط به نشان دادن تعداد مقادیر گمشده

با توجه به شکل (6) مشاهده می‌شود که ستون total\_bedrooms دارای مقادیر خالی است که در ادامه به مدیریت آنها خواهیم پرداخت.

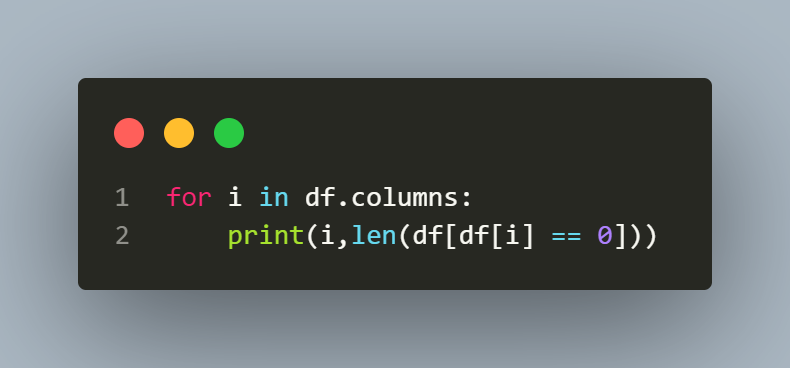


شکل 6: تعداد مقادیر خالی در هر سطر

در ادامه سه مورد را بررسی می‌کنیم: اول اینکه چه تعداد سطر تکراری داریم. شکل (7) نحوه انجام این کار را نمایش می‌دهد. سپس بررسی می‌کنیم که آیا در ستون‌هایی که مقدار صفر معنایی ندارد، رکوردی با مقدار صفر داریم یا خیر. شکل (8) نحوه انجام این کار را نمایش می‌دهد.



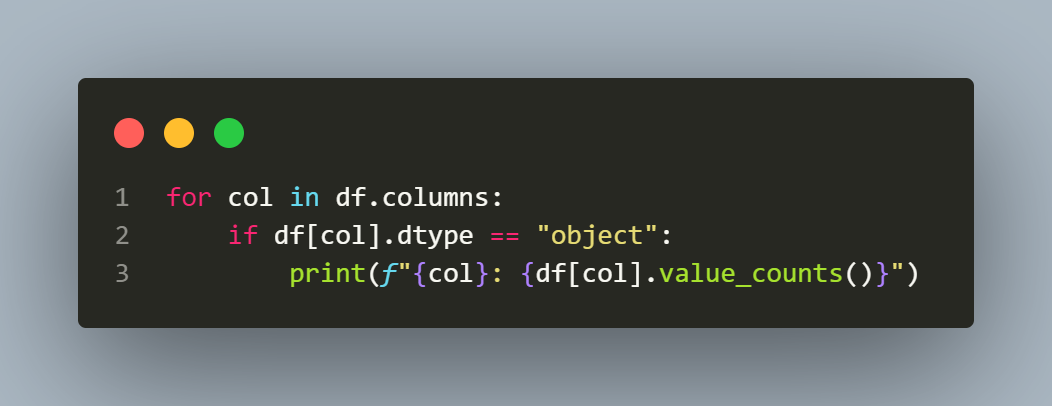
شکل 7: کد بررسی تعداد سطرهای تکراری



شکل 8: کد بررسی تعداد صفرها در هر ستون

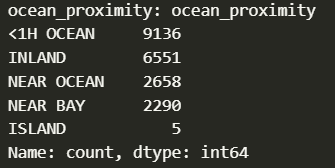
با بررسی خروجی این دو کد، متوجه می‌شویم که هیچ سطر تکراری در دیتافریم وجود ندارد. همچنین در هیچ یک از ستون‌ها مقدار صفر نداریم.

در قدم بعدی بررسی می‌کنیم که در سطرهایی که نوع داده‌ی دسته‌ای دارد، چه کلاس‌هایی موجود داریم و جمعیت هر دسته چقدر است.



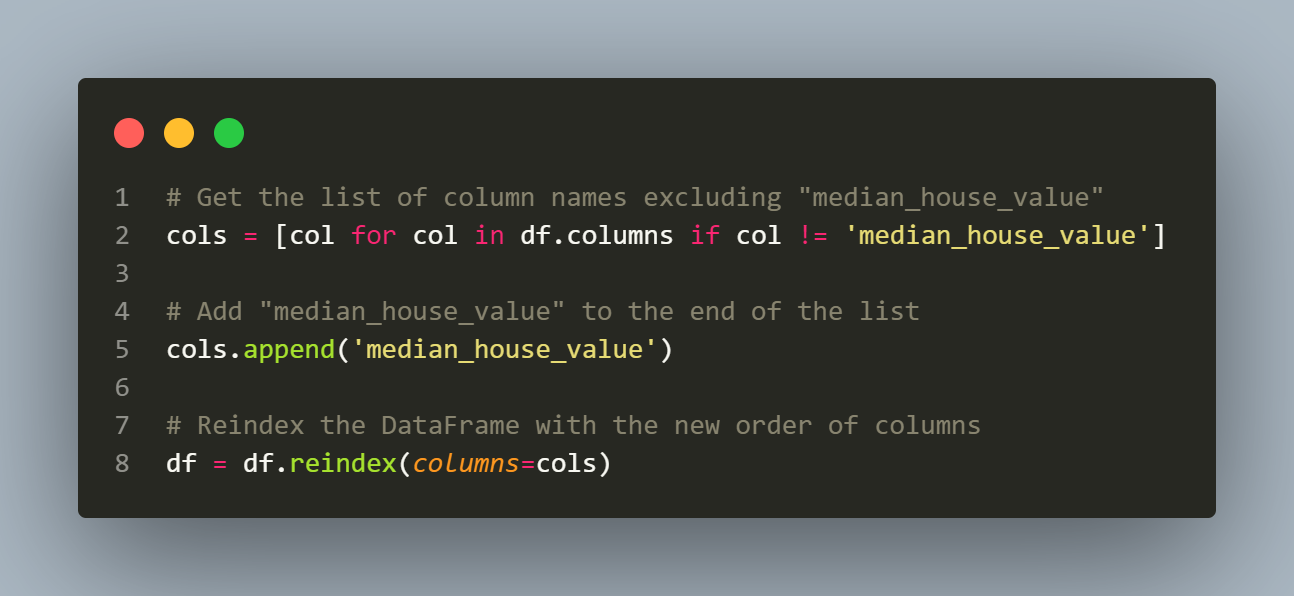
شکل 9: بررسی تعداد و جمعیت موجود در سطرهای گسسته

شکل (10) خروجی این کد را نمایش می‌دهد. مشاهده می‌شود که دسته‌ای با نام غیر معمول در ستون ocean\_proximity وجود ندارد.



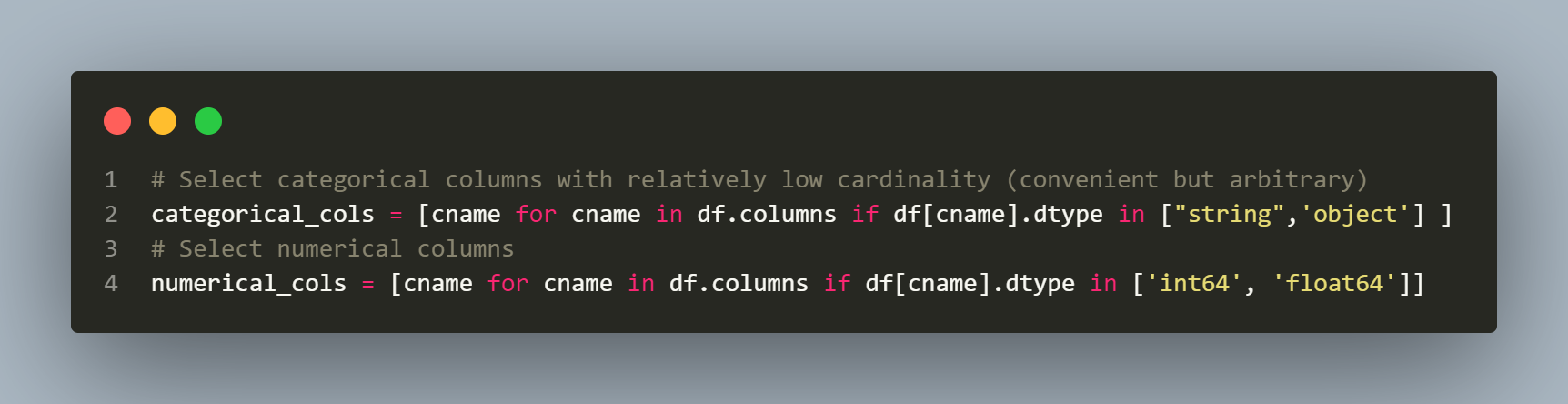
شکل 10: خروجی کد شکل 9

حال برای حفظ قرارداد، سطر هدف را به سطر آخر دیتافریم انتقال می‌دهیم. شکل (11) کدی که این کار را بر عهده دارد نمایش می‌دهد.



شکل 11: انتقال سطر هدف به سطر آخر

حال یک تقسیم‌بندی انجام می‌دهیم که در ادامه‌ی کار مفید واقع می‌شود. با استفاده از List comprehension نام ستون‌هایی که داده‌های عددی دارند را در یک لیست و نام ستون‌هایی که دارای مقادیر دسته‌ای هستند را در یک لیست دیگر قرار می‌دهیم. شکل (12) نحوه انجام این کار را نمایش می‌دهد.



شکل 12: جداسازی داده‌های عددی و دسته‌ای در دو لیست جدا

## 2-1-مصور سازی داده‌ها

### 1-2-1-رسم و تفسیر هیستوگرام

شکل 13 نحوه رسم هیستوگرام را نمایش می‌دهد. num\_rows و num\_cols ابعاد شبکه فرعی را مشخص می کنند. در این مورد، یک شبکه سه در سه ایجاد می کند، بنابراین می توان تا 9 نمودار فرعی را در خود جای داد. تابع plt.subplots برای تولید یک شکل و مجموعه ای از نمودارهای فرعی (محور) استفاده می شود. (20,10) figsize اندازه کل شکل (20 اینچ عرض و 10 اینچ ارتفاع) را مشخص می کند.

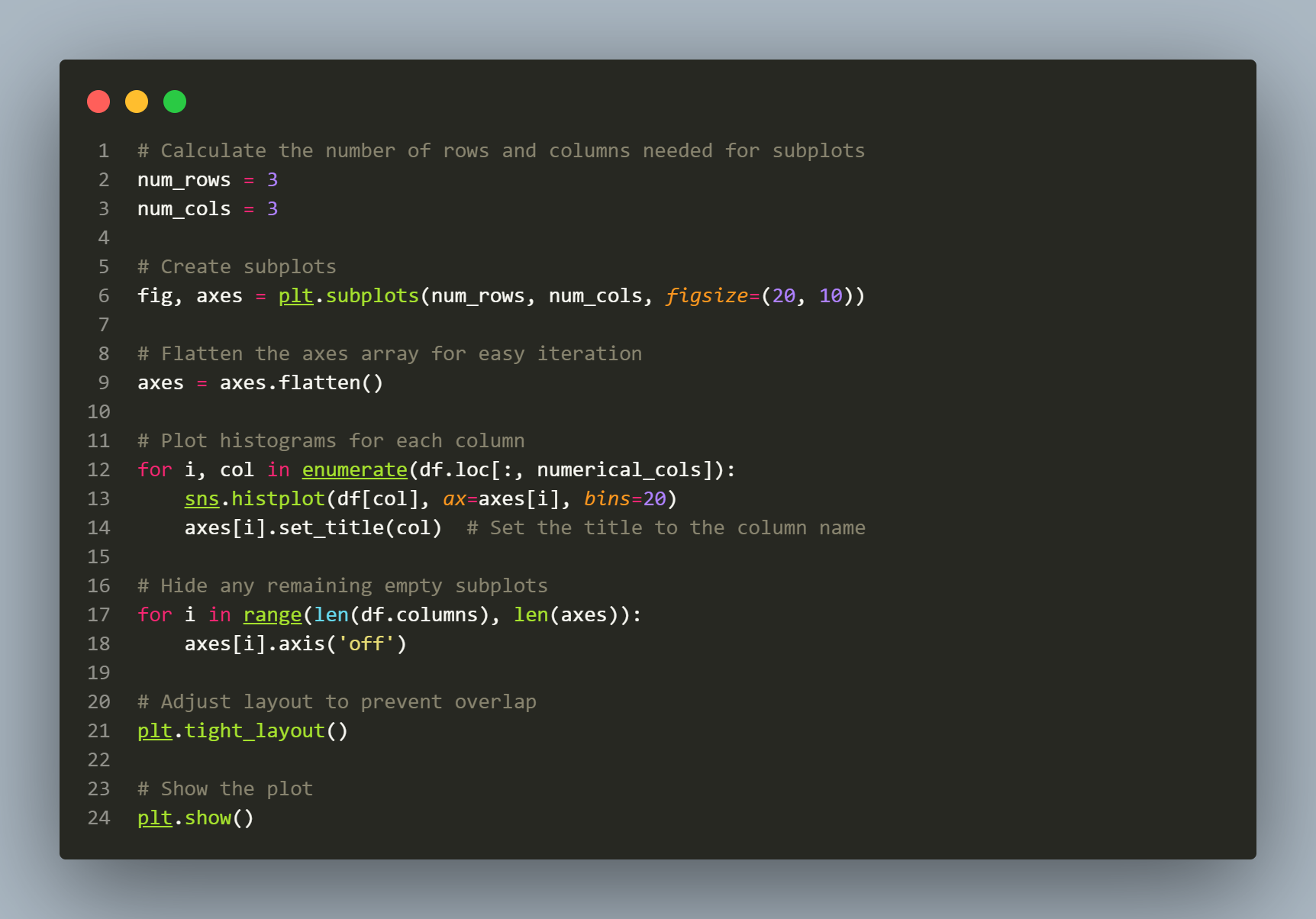
سپس حلقه روی هر ستون مشخص شده در لیست numerical\_cols از df تکرار می شود. برای هر ستون:

sns.histplot یک هیستوگرام از ستون ایجاد می کند و آن را بر روی نمودار فرعی مربوطه ترسیم می کند bins=20 مشخص می کند که هر هیستوگرام باید 20 میله داشته باشد. سپس عنوان هر نمودار بالای آن مشخص می‌شود.

سپس لوپ بعدی هر فضای خالی که در آن نمودار وجود ندارد را حذف می‌کند.

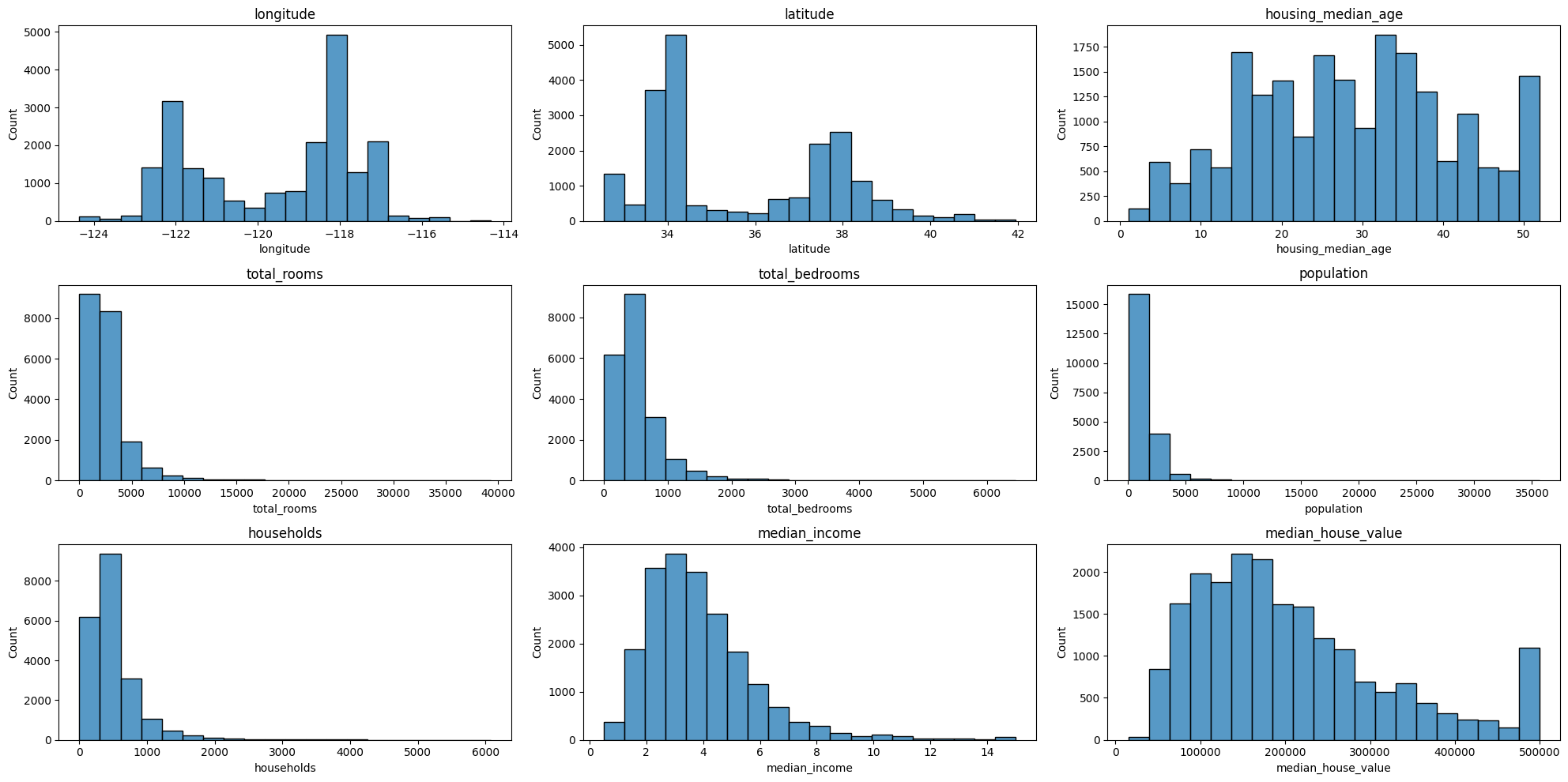
()plt.tight\_layout، نمودارهای فرعی و برچسب ها را طوری تنظیم می کند که به خوبی در ناحیه شکل قرار گیرند و روی هم قرار نگیرند.

plt.show () شکل کامل را با تمام نمودارهای فرعی نمایش می دهد.



شکل 13: کد مربوط به رسم هیستوگرام

همچنین شکل (14) هیستوگرام‌های رسم شده را نمایش می‌دهد.



شکل 14: هیستوگرام‌های رسم شده

**تفسیر:**

**Longitude:**

توزیع چند وجهی است، که نشان دهنده چندین قله است که در آن داده های مسکن به صورت طولی متمرکز شده اند.

این نشان می دهد که خوشه های متعددی از خانه ها در طول های جغرافیایی مختلف، احتمالاً مربوط به شهرها یا مناطق مختلف است.

**Latitude:**

مشابه طول جغرافیایی، توزیع چند وجهی با تمرکز داده های مسکن در عرض های جغرافیایی خاص است. این خوشه ها را در مناطق جغرافیایی خاصی نشان می دهد، که وقتی با طول جغرافیایی ترکیب می شوند، می توانند مناطق جغرافیایی خاصی را مشخص کنند.

**Housing Median Age:**

این هیستوگرام توزیع نسبتاً یکنواختی با قله های متعدد نشان می دهد که نشان می دهد خانه هایی در طیف وسیعی از سنین وجود دارد.

برای خانه های بسیار جدید و بسیار قدیمی، با تعداد خانه های بیشتر در محدوده سنی متوسط، افت جزئی وجود دارد.

**Total Rooms:**

چولگی به سمت راست است، به این معنی که بیشتر نقاط داده در سمت چپ قرار دارند، با تعداد خانه های کمتری که تعداد کل اتاق ها بسیار زیاد است.

تعداد کمی از خانه ها با تعداد بسیار زیاد اتاق وجود دارد که به صورت پرت به نظر می رسند.

**Total Bedrooms:**

مشابه کل اتاق ها، این هیستوگرام نیز دارای انحراف راست است.

اکثر خانه ها اتاق خواب های کمتری دارند و با افزایش تعداد اتاق خواب ها، تعداد آنها کاهش می یابد.

**Population:**

هیستوگرام جمعیت نیز دارای انحراف راست است، که مناطق زیادی با جمعیت کوچکتر و مناطق کمتر با جمعیت بسیار زیاد را نشان می دهد.

**Households:**

این یکی دیگر از توزیع‌های منحرف به راست است که با افزایش تعداد خانوارها، کاهش شدیدی دارد. بیشتر نقاط داده در ناحیه ای متمرکز شده اند که خانوارهای کمتری را نشان می دهد.

**Median Income:**

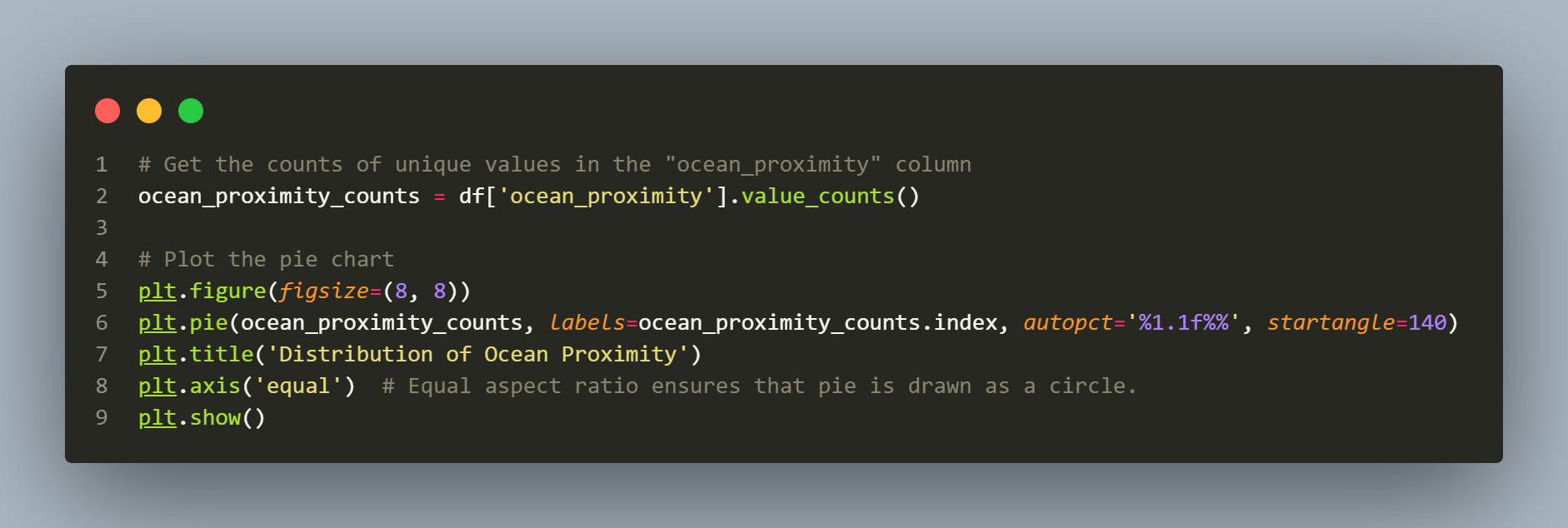
به نظر می‌رسد توزیع میانه درآمد کمی چوله به راست است، اما به طور نرمال‌تر از سایر ویژگی‌ها توزیع می‌شود، که نشان می‌دهد سطوح درآمد به طور یکنواخت‌تر در میان مجموعه داده‌ها توزیع می‌شود، البته با دنباله‌ای به سمت درآمدهای بالاتر.

**Median House Value:**

هیستوگرام مقدار متوسط خانه یک توزیع چند وجهی با چندین قله را نشان می دهد، که نشان می دهد خانه ها در دسته های ارزش مجزا قرار می گیرند. این نشان‌دهنده قیمت‌های متنوع مسکن، با اوج‌ها در محدوده‌های ارزشی خاص است.

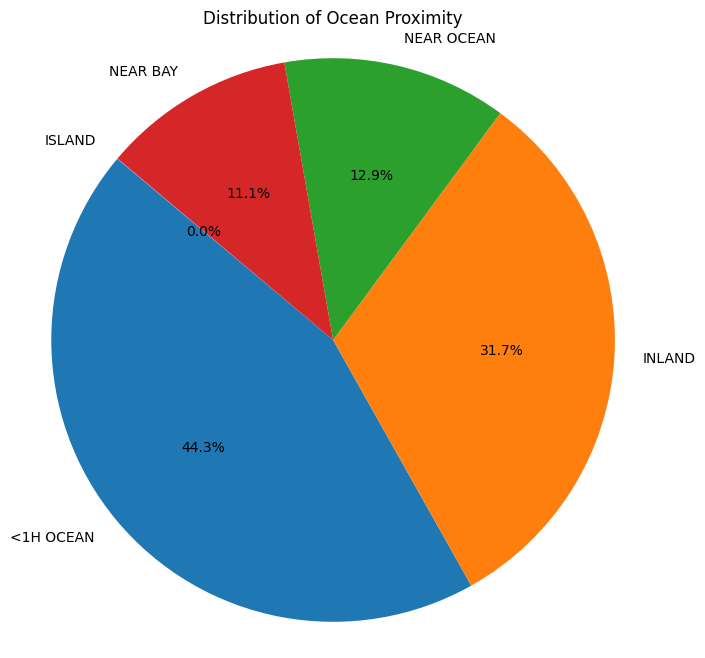
### 2-2-1- رسم نمودار دایره‌ای و تفسیر آن

برای تنها ستون داده‌های دسته‌ای این دیتافریم یک نمودار دایره‌ای رسم می‌کنیم. نحوه رسم این نمودار به این صورت است که ابتدا جمعیت هر دسته را با متد .value\_counts() محاسبه می‌کنیم سپس با استفاده از کتابخانه matplotlib نمودار را رسم می‌کنیم. شکل (15) نمودار رسم شده را نمایش می‌دهد.



شکل 15: کد رسم نمودار دایره‌ای

همچنین شکل (16) خروجی این کد را نمایش می‌دهد.



شکل 16: نمودار دایره‌ای

**تفسیر:**

نزدیکی به اقیانوس: داده‌ها نشان می‌دهند که طبقه‌بندی واضحی از مسکن‌ها بر اساس نزدیکی به اقیانوس وجود دارد و اکثر خانه‌ها به اقیانوس نزدیک هستند.

همچنین بخش قابل توجهی از خانه‌ها از اقیانوس دور هستند که نشان می‌دهد دیتاست پراکندگی خوبی دارد.

خانه‌های ساخته شده در جزیره در این مجموعه داده یا وجود ندارند یا بسیار نادر هستند.

### 3-2-1-رسم و تفسیر نمودارهای جعبه‌ای

رسم نمودارهای جعبه‌ای برای ستونهای عددی همانند رسم هیستوگرام‌ها است با این تفاوت که از تابع boxplot() برای رسم نمودار استفاده می‌شود. شکل (17) کد رسم کننده این نمودارها را نمایش می‌دهد. همچنین شکل (18) خروجی این کد را نمایش می‌دهد.

**تفسیر:**

**Longitude and Latitude:**

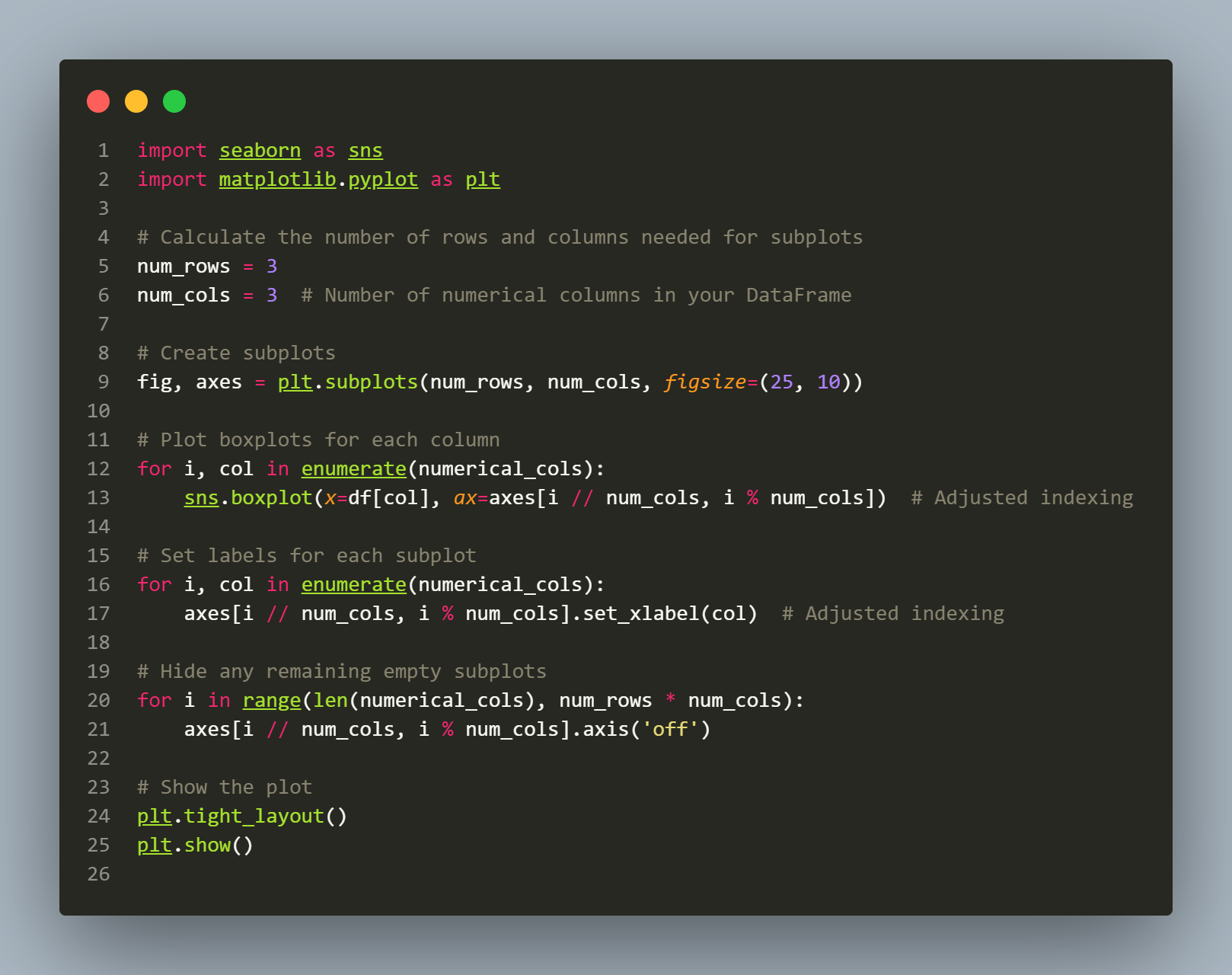
این دو نمودار موقعیت جغرافیایی املاک را نشان می دهند. جعبه ها طول کمی دارند و محدوده متمرکزی از مقادیر را با چند نقطه پرت نشان می دهند. این گسترش حاکی از یک منطقه جغرافیایی محدود در حال مطالعه است، مانند یک منطقه یا ایالت خاص.

Housing Median Age:

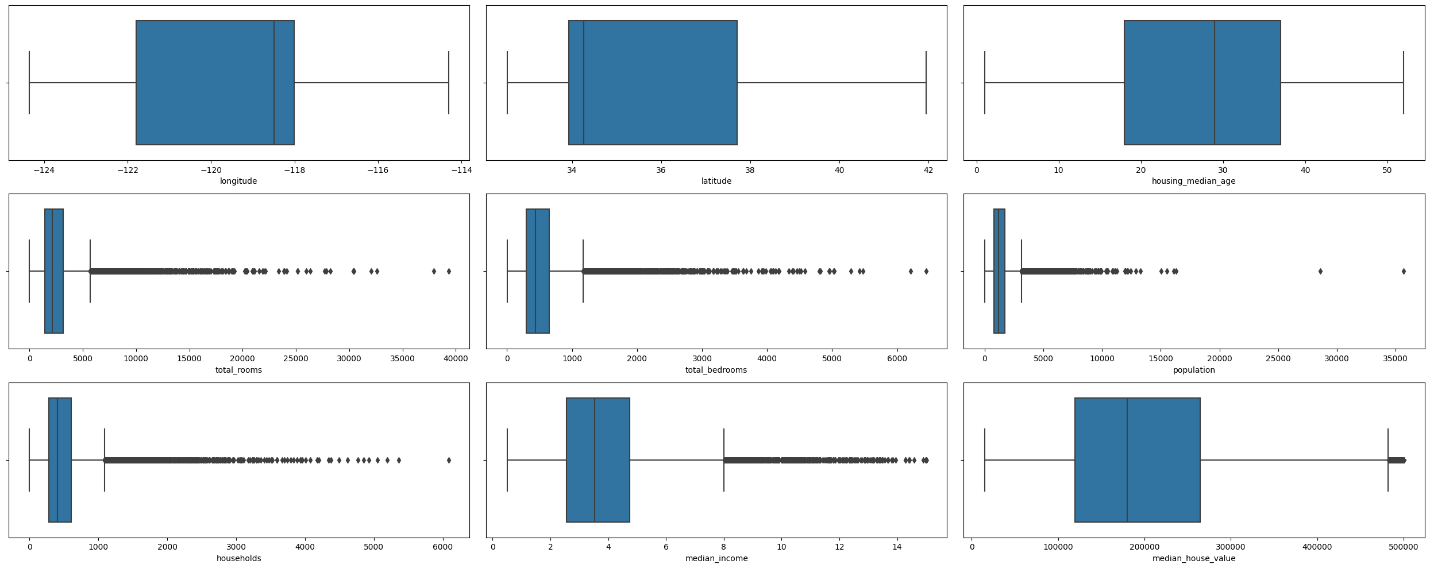
توزیع نسبتاً یکنواخت به نظر می رسد، با تمرکز جزئی در سنین میانی. این نشان دهنده تنوع سن خانه‌هاست.

Total Rooms, Total Bedrooms, Population, و Households:

این نمودارها توزیعی به سمت راست را نشان می‌دهند که از سبیل‌های بلندی که به سمت راست امتداد می‌یابند و خط میانی به سمت چپ جعبه مشهود است.



شکل 17: کد رسم کننده نمودارهای جعبه‌ای



شکل 18: نمودارهای جعبه‌ای

سبیل‌های بلند محدوده ای از مقادیر بزرگتر را نشان می دهد، در حالی که کادر نشان می دهد که قسمت عمده داده ها کجا قرار دارد. این نشان می‌دهد که در حالی که اکثر املاک دارای محدوده کمتری از اتاق‌ها، اتاق‌خواب‌ها، جمعیت و اندازه‌های خانوار هستند، اما نقاط پرت قابل توجهی با مقادیر بسیار بالا وجود دارد.

این الگو اغلب نشان می‌دهد که چند ملک بسیار بزرگ یا مناطق پرجمعیت وجود دارد، اما اکثریت خانه‌ها، اتاق‌های کمتر و جمعیت/خانوارهای کوچک‌تری دارند.

Median Income:

درآمد متوسط دارای چولگی به راست است اما چولگی کمتری نسبت به کل اتاق ها یا اتاق خواب ها دارد. مقادیر پرت کمتری وجود دارد که نشان دهنده توزیع یکنواخت درآمد در میان مجموعه داده است.

Median House Value:

توزیع ارزش خانه دارای دامنه وسیعی است، با این جعبه که بسیار گسترده است، که نشان می دهد تغییرات قابل توجهی در قیمت خانه وجود دارد. میانه به چارک پایین نزدیکتر است، که نشان می دهد خانه های بیشتری کمتر از میانه قیمت گذاری شده اند.

سبیل بلند سمت راست نشان می دهد که خانه هایی با ارزش به طور قابل توجهی بالاتر از میانه وجود دارد که نشان دهنده وجود املاک لوکس یا با ارزش بالا است.

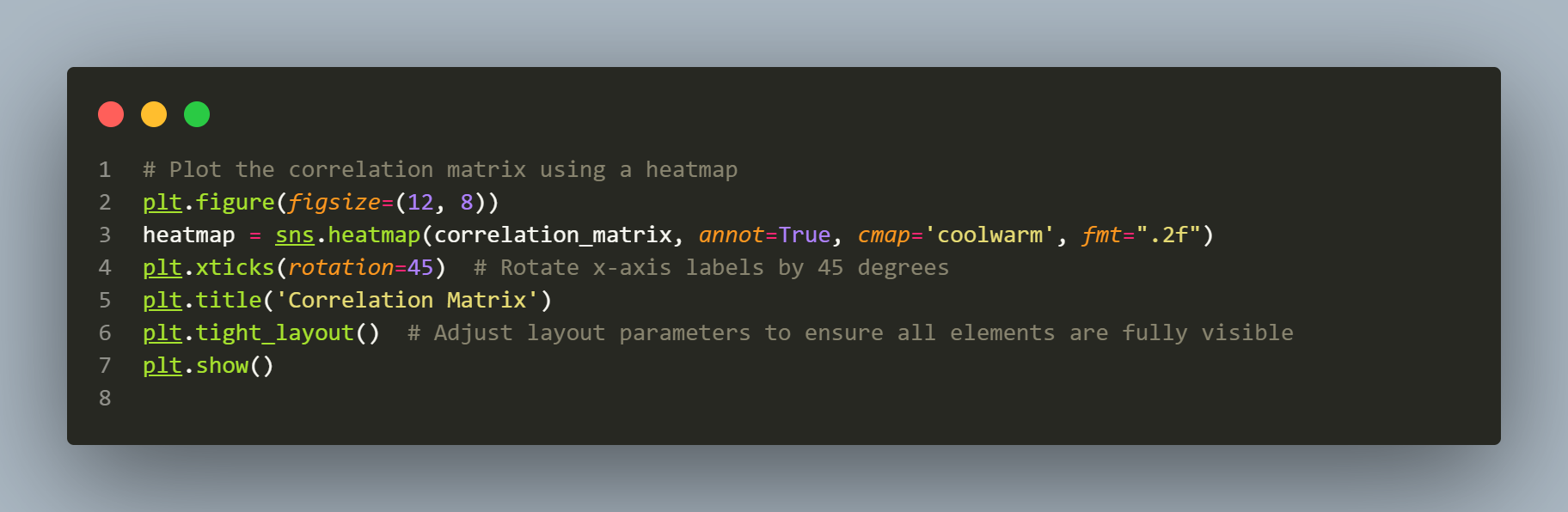
4-2-1-رسم و تفسیر ماتریس همبستگی

برای رسم ماتریس، ابتدا باید داده‌های دسته‌ای را one hot encode کرد. برای این منظور از متد get\_dummies() استفاده می‌کنیم. سپس ستون هدف و ستون ocean\_proximity را حذف می‌کنیم و ابتدا ستون‌های انکود شده و سپس ستون هدف را به دیتافریم می‌چسبانیم. برای این منظور از تابع pd.concat() استفاده می‌کنیم. در ادامه با استفاده از متد corr() ماتریس همبستگی را ایجاد می‌کنیم. شکل (19) نحوه انجام این کار را نمایش می‌دهد.



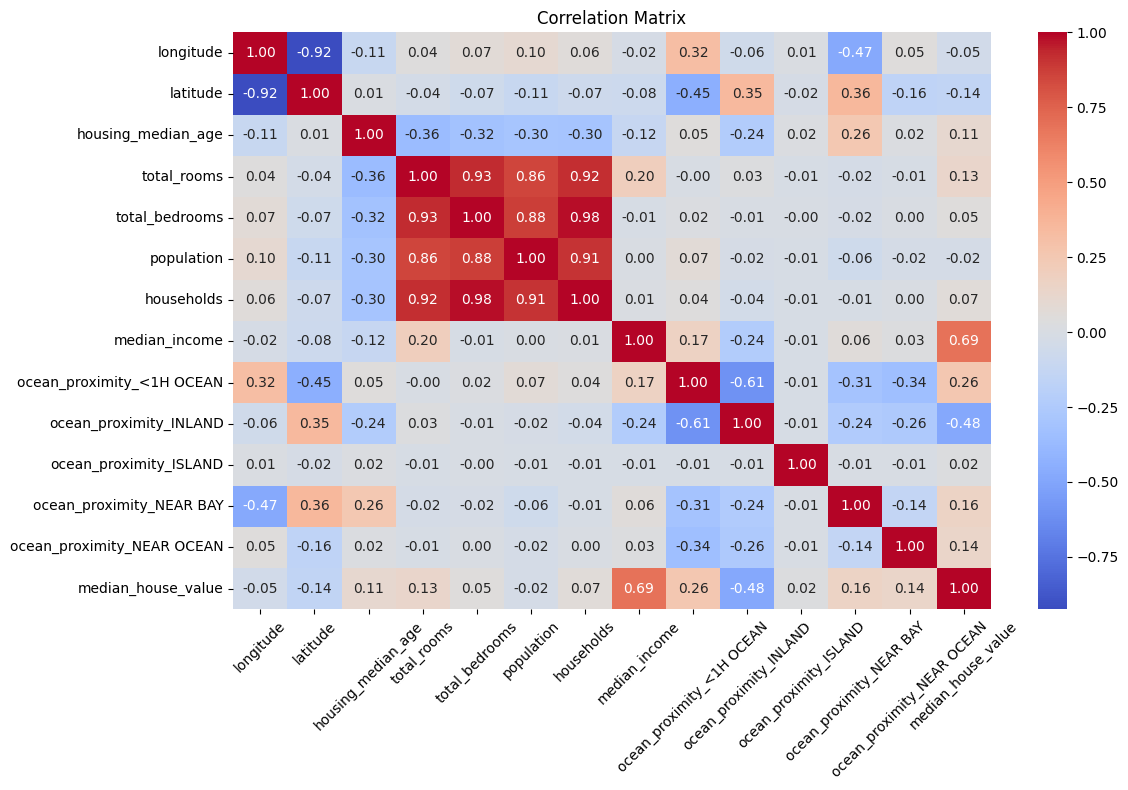
شکل 19: آماده‌سازی دیتافریم برای محاسبه ماتریس همبستگی

سپس با استفاده از کتابخانه seaborn و تابع heatmap() ماتریس ایجاد شده را رسم می‌کنیم. شکل (20) نحوه رسم این ماتریس را نمایش می‌دهد.



شکل 20: رسم ماتریس همبستگی

همچنین شکل (21) ماتریس رسم شده را نمایش می‌دهد.



شکل 21: ماتریس همبستگی رسم شده

**تفسیر:**

طول و عرض جغرافیایی: یک همبستگی منفی قوی بین طول و عرض جغرافیایی وجود دارد، که نشان می دهد در این مجموعه داده جغرافیایی، با رفتن به سمت غرب (با افزایش طول جغرافیایی)، فرد تمایل به رفتن به سمت جنوب (کاهش عرض جغرافیایی) دارد یا برعکس.

میانه سن مسکن: به نظر می‌رسد که با اکثر متغیرهای دیگر همبستگی کمی یا بدون وجود دارد، که نشان می‌دهد سن خانه‌ها رابطه خطی قوی با ویژگی‌هایی مانند اتاق‌ها، اتاق‌خواب‌ها یا اندازه خانه ندارد.

مجموع اتاق‌ها/اتاق‌های خواب و خانوارها/جمعیت: همبستگی‌های مثبت قوی در این گروه‌ها وجود دارد، که شهودی است زیرا اتاق‌های بیشتر معمولاً با تعداد اتاق‌خواب‌های بیشتر مرتبط هستند و اندازه جمعیت بزرگ‌تر با تعداد خانوارهای بیشتر مرتبط است.

درآمد متوسط و میانه ارزش خانه: یک همبستگی مثبت قابل توجه وجود دارد، که نشان می دهد مناطق درآمد متوسط بالاتر با ارزش خانه بالاتر مرتبط هستند، که یک الگوی اقتصادی رایج است.

نزدیکی اقیانوس: دسته‌های مختلف نزدیکی اقیانوس درجات مختلفی از همبستگی با میانگین ارزش خانه دارند. این نشان می دهد که نزدیکی به اقیانوس ممکن است قیمت مسکن را به طور متفاوت تحت تاثیر قرار دهد، به طوری که برخی از مقوله ها دارای همبستگی مثبت و برخی دیگر منفی هستند.

### 5-2-1-رسم و تفسیر نمودار همبستگی با ستون هدف

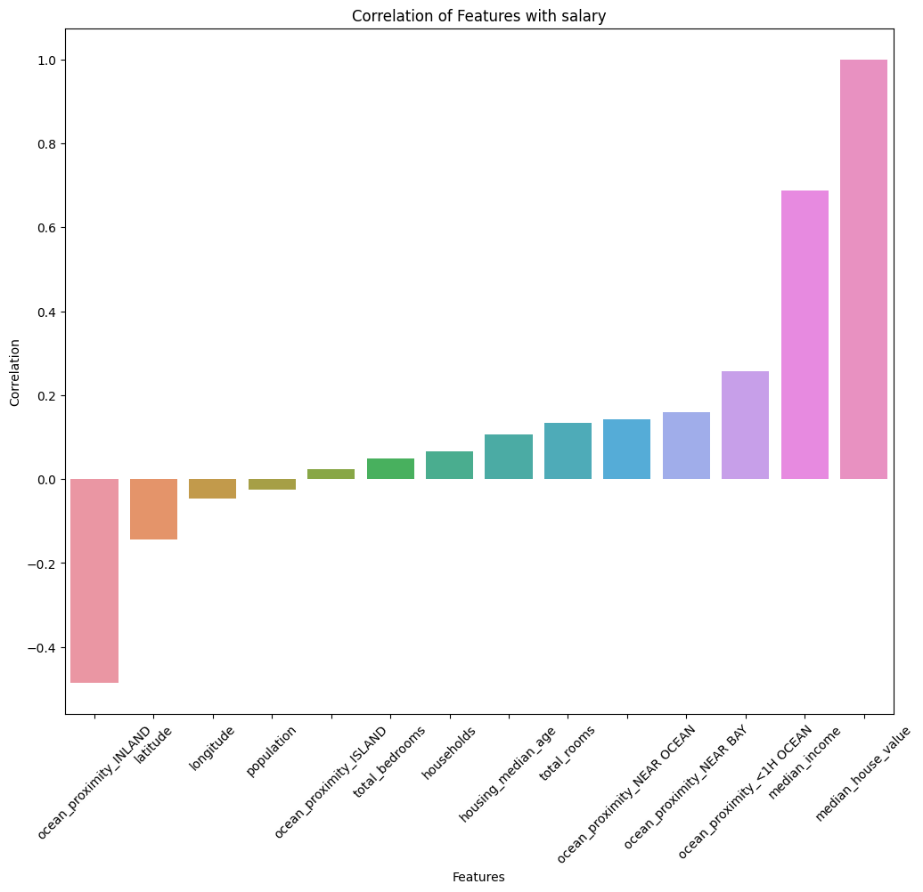
برای رسم این نمودار، ابتدا با استفاده از متد corrwith() همبستگی ستون‌ها با ستون هدف را محاسبه می‌کنیم. سپس این مقادیر را در متغیر correlation ذخیره می‌کنیم. سپس مقادیر موجود در این لیست را با متد sort\_values() به طور صعودی مرتب می‌کنیم. و با استفاده از تابع barplot() از کتابخانه seaborn از این مقادیر نمودار میله‌ای رسم می‌کنیم. شکل (22) کدی که وظیفه‌ی انجام این عمل را به عهده دارد را نمایش می‌دهد. همچنین شکل (23) خروجی این کد را نمایش می‌دهد.

تفسیر:

همبستگی منفی (نوارهای قرمز و نارنجی):



شکل 22: کد رسم کننده نمودار میزان همبستگی با سطر هدف



شکل 23: نمودار میزان همبستگی با سطر هدف

ویژگی‌های «ocean\_proximity\_INLAND» و «Latitude» همبستگی منفی با ستون هدف دارند، که نشان می‌دهد با افزایش این مقادیر ویژگی، حقوق تمایل به کاهش دارد.

'ocean\_proximity\_INLAND' قوی ترین همبستگی منفی را در بین ویژگی های نمایش داده شده دارد.

همبستگی ضعیف تا بدون (نوارهای زرد تا آبی روشن):

ویژگی‌هایی مانند «longitude»، «population»، «ocean\_proximity\_ISLAND»، «households»، «total\_bedrooms»، « housing\_median\_age» و «total\_rooms» همبستگی ضعیف و بدون هیچ ارتباطی با ستون هدف نشان می‌دهند.

همبستگی مثبت (نوارهای آبی تا صورتی):

ویژگی‌های «ocean\_proximity\_NEAR BAY» و «ocean\_proximity\_<1H OCEAN» همبستگی مثبتی دارند، که نشان می‌دهد با افزایش نزدیکی به این مناطق، میزان ستون هدف نیز افزایش می‌یابد.

« median\_income» و « median\_house\_value» همبستگی مثبت قوی با ستون هدف نشان می‌دهند.

قوی ترین همبستگی ها (نوارهای صورتی):

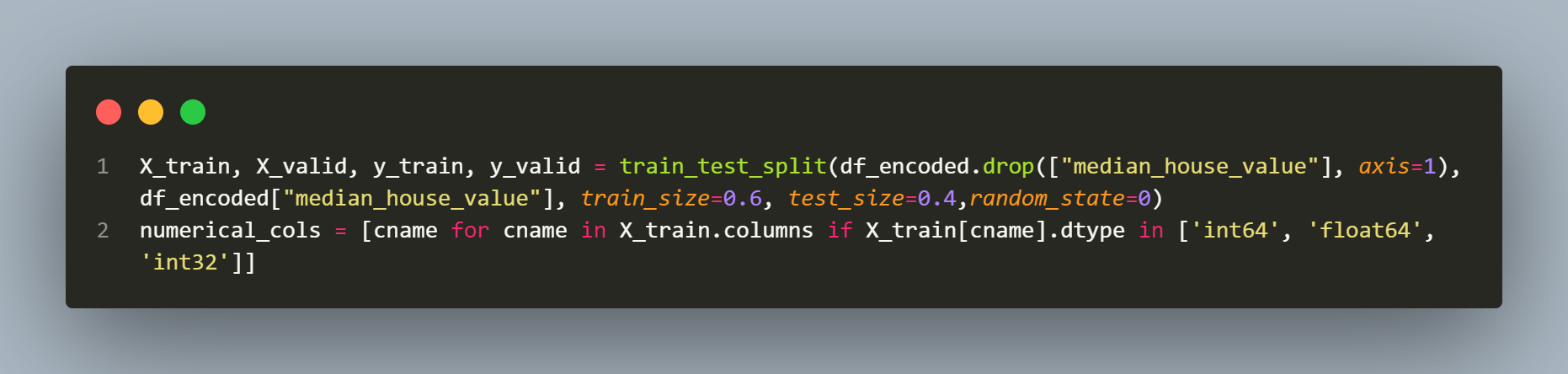
میله 'median\_income' بسیار بلند است، که یک همبستگی مثبت قوی را نشان می دهد، که به طور بالقوه نشان می دهد که این ویژگی می تواند پیش بینی خوبی برای ستون هدف باشد.

## 3-1-پر کردن مقادیر خالی

برای پر کردن مقادیر خالی دو روش را با هم مقایسه می‌کنیم. اولین روش پر کردن مقادیر با استفاده از KNN است و روش دوم استفاده از میانگین است. دیتاست حاصل از این دو روش را با الگوریتم رگرسور جنگل تصادفی و ارزیابی مقدار R2 مقایسه می‌کنیم.

### 1-3-1-تقسیم بندی به دیتاست آموزشی و تستی

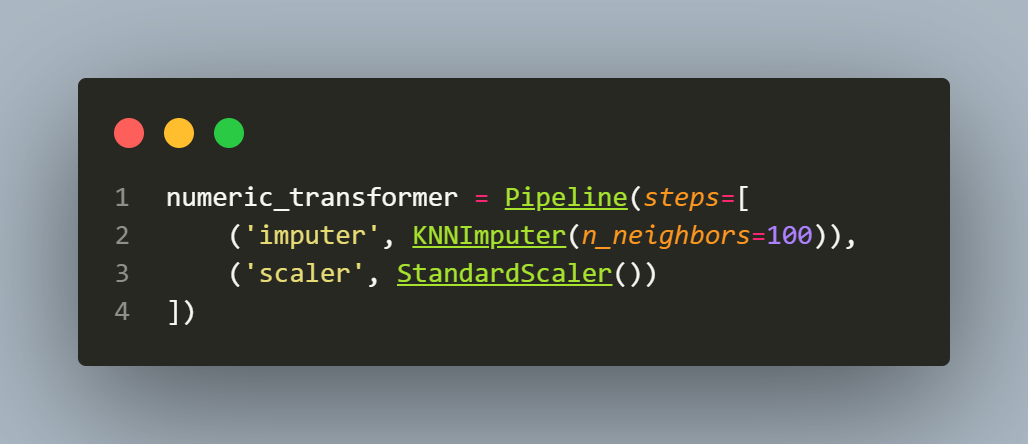
از آنجایی که در مرحله قبلی ستون دسته‌ای را انکد کردیم، می‌توان از همان دیتاست قسمت قبلی برای تقسیم بندی داده‌ها استفاده کرد. حال بار دیگر لیست ستون‌های عددی را تعریف می‌کنیم تا ستون‌های انکود شده هم در این لیست قرار بگیرند. شکل (24) نحوه انجام این کار را نمایش می‌دهد.



شکل 24: تقسیم داده‌ها به دیتاست تمرینی و تستی

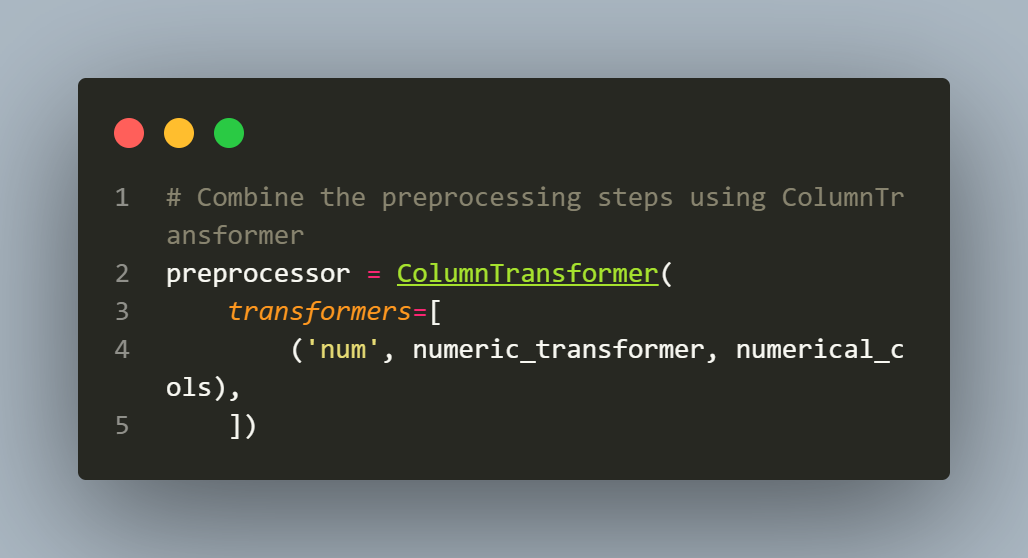
### 2-3-1-تعریف شی پیش‌پردازش برای پر کردن مقادیر خالی با KNN

در ادامه برای انجام راحت پیش‌پردازش یک شی از کلاس پایپ‌لاین تعریف می‌کنیم. این شی دسته‌ای از عملیات‌ها را به ترتیب روی هدف انجام می‌دهد. شی تعریف شده در این قسمت، ابتدا مقادیر خالی را با استفاده از تابع KNNImputer() پر می‌کند و سپس داده‌ها را استاندارد سازی می‌کند. شکل (25) کد مربوط به این شی را نمایش می‌دهد.



شکل 25: ایجاد شی از کلاس پایپ‌لاین

در ادامه یک شی از کلاس ColumnTransformer() به نام preprocessor تعریف می‌کنیم. این شی می‌تواند یک یا چند عمل را ستون به ستون روی دیتافریم انجام دهد. سپس پایپ‌لاین ایجاد شده را به این شی پاس می‌دهیم. این کار باعث می‌شود که در صورت اجرای متد fit() از این شی، عملیات‌های تعریف شده در پایپ‌لاین ستون به ستون روی دیتافریم اجرا شوند. شکل (26) این کلاس و شی ایجاد شده از آن را نمایش می‌دهد.

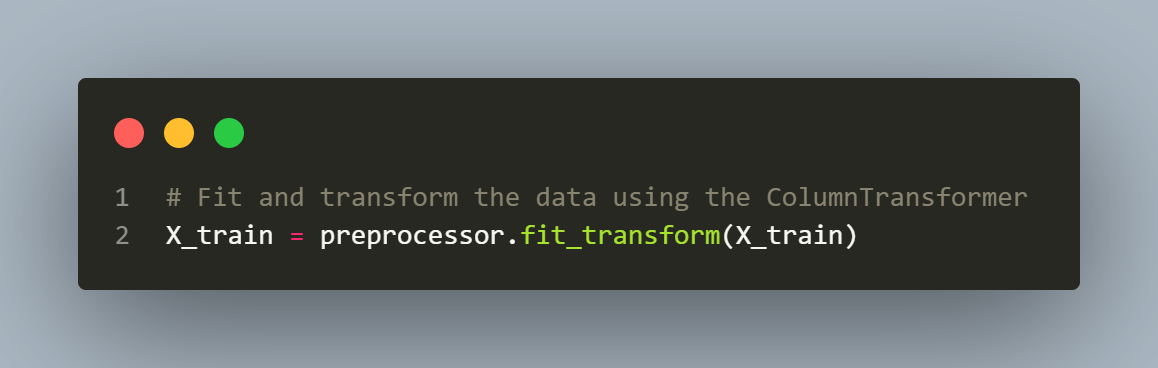


شکل 26: ایجاد شی از کلاس ColumnTransformer

### 3-3-1-انجام پیش پردازش و تعریف مدل

در ادامه از شی preprocessor متد .fit\_transform() را فراخوانی می‌کنیم و دیتاست آموزشی را به آن پاس می‌دهیم. این متد پارامتر دیتاستی که به عنوان آرگومان دریافت کرده را یاد می‌گیرد، سپس با توجه به پارامترهایی که محاسبه کرده، عملیات‌های تعریف شده را انجام می‌دهد. و سپس نتیجه را در همان متغیر قبلی ذخیره می‌کنیم. شکل (27) نحوه عملکرد این کد را نمایش می‌دهد.

سپس برای پیش‌پردازش داده‌های تست، از متد transform() شی استفاده می‌کنیم. دلیل استفاده از این متد این است که این متد با استفاده از پارامترهای یادگرفته شده از داده‌های آموزشی، عملیات را روی داده‌های تستی انجام می‌دهد. توجه شود که اینجا از متد قبلی استفاده نمی‌شود



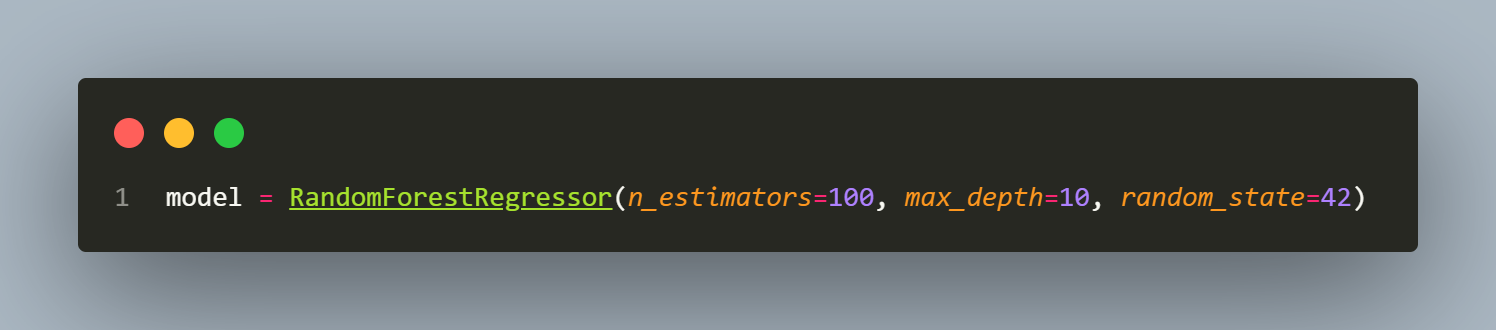
شکل 27: پیش‌پردازش داده‌های آموزشی

زیرا فرض بر این است که پارامترهای داده‌های تست را نداریم و باید با استفاده از برآوردی که در داده‌های آموزشی انجام دادیم، عملیات‌ها را روی داده‌های تست انجام دهیم. عدم رعایت این مورد، موجب ایجاد نشت داده می‌شود. شکل (28) نحوه پیش‌پردازش داده‌های تست را نمایش می‌دهد.



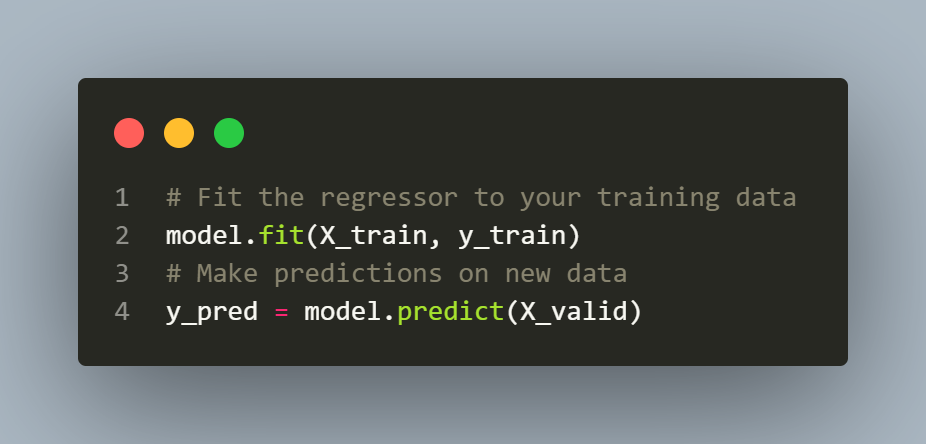
شکل 28: پیش‌پردازش داده‌های تست

سپس مدل را تعریف می‌کنیم. یک شی از کلاس RandomForestRegressor() ایجاد می‌کنیم و آن را model می‌نامیم. حداکثر عمق درخت‌ها را 10، و تعداد برآورد کننده ها را برابر 100 قرار می‌دهیم. همچنین برای امکانپذیری بازتولید نتایج، حالت رندوم را برابر 42 قرار می‌دهیم. شکل (29) تعریف مدل را نمایش می‌دهد.



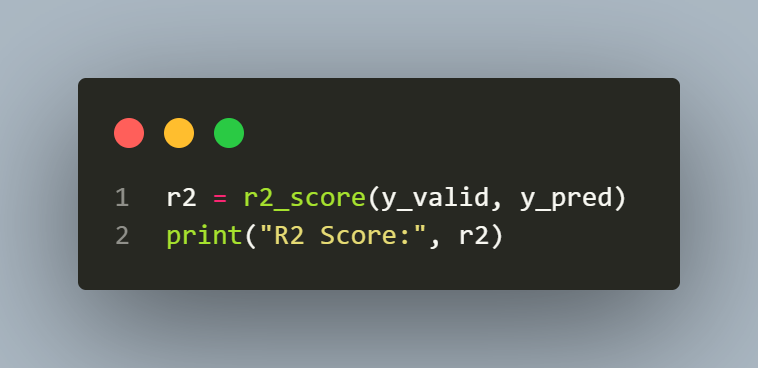
شکل 29: تعریف مدل

سپس مدل را با استفاده از متد .fit() روی داده‌های آموزشی آموزش می‌دهیم و با پاس دادن داده‌های تست به متد .predict() ، پیش‌بینی مدل را در متغیر y\_pred ذخیره می‌کنیم. شکل (30) نحوه انجام این دو عملیات را نمایش می‌دهد.



شکل 30: آموزش و تست مدل

سپس مقدار R2 را محاسبه می‌کنیم. شکل (31) نحوه انجام این کار با تابع r2\_score() را نمایش می‌دهد.

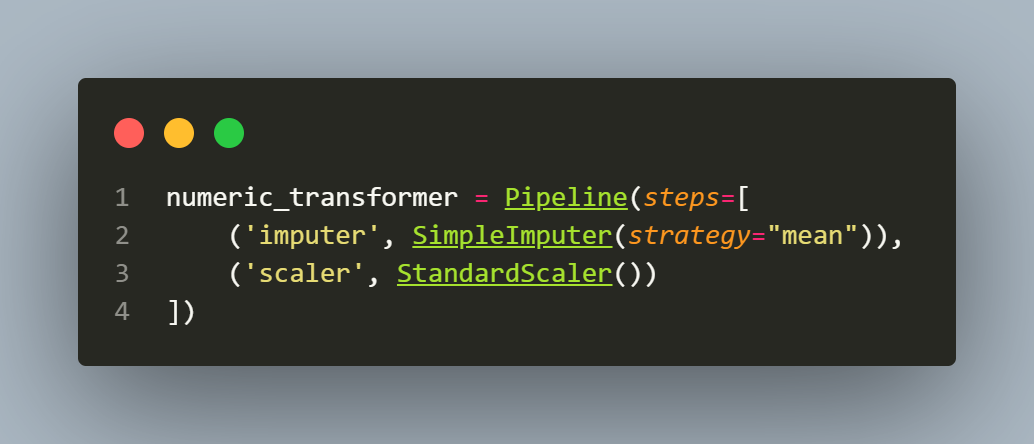


شکل 31: محاسبه و نمایش R2

مقدار R2 از این روش برابر 0.778 می‌شود. در قدم بعدی مقادیر خالی را با میانگین پر می‌کنیم.

### 4-3-1-تعریف شی پیش‌پردازش برای پر کردن مقادیر خالی با میانگین

حال شی تعریف شده برای پیش‌پردازش را به نحوی تغییر می‌دهیم که مقادیر خالی را با میانه پر کند. برای این منظور، در شی پایپ‌لاین، به جای KNNImputer() کلاس SimpleImputer() با آرگومان strategy = ‘mean’ را پاس می‌دهیم. شکل (32) پایپ‌لاین به روز شده را نمایش می‌دهد.



شکل 32: پایپ‌لاین به روز شده

سایر مراحل را مشابه بخش قبلی انجام می‌دهیم. و مدل را آموزش و تمرین می‌دهیم و مقدار R2 را محاسبه می‌کنیم. در این حالت مقدار R2 برابر 0.7777 خواهد شد. بنابراین برای پر کردن مقادیر خالی از میانگین استفاده می‌کنیم.

## 4-1-تشخیص داده‌های پرت

برای تشخیص داده‌های پرت از روش شش سیگما استفاده می‌کنیم. نحوه کارکرد این روش به این صورت است که برای هر ستون میانگین و واریانس محاسبه می‌کنیم و برای هر ستون، هر داده‌ای که کمتر از میانگین منهای سه سیگما و بزرگتر از میانگین به علاوه سه سیگما بود را به عنوان داده پرت شناسایی می‌کنیم. برای مدیریت داده‌های پرت دو رویکرد را با هم مقایسه می‌کنیم. این دو رویکرد عبارت است از حذف داده‌های پرت یا جایگزینی آنها با حد پایین یا حد بالا.

روی دیتاست‌های به دست آمده از هر روش، مدل جنگل تصادفی با پارامترهای بخش قبلی را آموزش می‌دهیم و مقدار R2 را محاسبه می‌کنیم.

1-4-1-جایگزینی داده‌ها با حد پایین یا حد بالا

ابتدا از دیتافریم به دست آمده از بخش قبلی، یک کپی در متغیر cleaned\_df ذخیره می‌کنیم، حالا با استفاده از یک حلقه، برای هر ستون میانگین، واریانس، حد پایین و حد بالا را محاسبه می‌کنیم.