به نام خدا

عنوان پروژه: پیشپردازش و مصورسازی دیتاست

دانشجو: امیرحسین بیاتی

شماره دانشجویی: 986127012

داکیومنت آپلود شده در سایت ککل

پروژه آپلود شده در گیت هاب

خلاصه کارهای انجام شده:

1. پاکسازی دیتاهای نامناسب
2. انجام عملیات های ریاضی بر داده های موجود
3. رسم توزیع آماری
4. رسم هیستوگرام
5. مقاسیه و نرمالایز کردن داده ها
6. مقایسه همبستگی(کارولویشن) بین ویژگی های مختلف
7. ترسیم scatter plot برای مقایسه داده ها

پاکسازی دیتاهای نامناسب

در این قسمت داده داده های گم شده، غلط، تکراری و پرت شناسایی و حذف شدند تا با مقایسه بهتر به توان به نتایج درست تری دست پیدا کرد.

برای پاکسازی دیتاهای غلط ابتدا داده ها و خانه های خالی شناسایی و حذف شدند،سپس داده هایی که فرمت درستی نداشتند به عنوان مثال کارکتری از حروف بودند حذف شدند.

```
data.dropna(inplace=True)
```

Remove Wrong Data Formats

```
for i in data.index:
    try:
        data.loc[i] = pd.to_numeric(data.loc[i])
    except:
        data.drop(i, inplace=True)
```

علاوه بر آن برخی ستون ها (ستون های A2-A5) مجاز به استفاده از رقم 0 نبودند و آنها نیز حذف شدند.

Remove Wrong Data

```
for x in data.index:
    if data.loc[x, "output"] > 1 or data.loc[x, "output"] < 0:
        data.drop(x, inplace=True)

    if data.loc[x, "A4"] == 0 or data.loc[x, "A2"] == 0 or data.loc[x, "A3"] == 0 or data.lo
c[x, "A5"] == 0:
        data.drop(x, inplace=True)
```

برای حذف داده های پرت نیز از فرمت iqr استفاده شد تا دیتاهای قابل اعتماد تری برای بررسی داشته باشیم.

در روش iqr ابتدا 25درصد حد بالایی و پایینی داده ها محاسبه میشوند و داده های خارج از محدوده بین آنها به عنوان داده پرت در نظر گرفته میشوند.

Remove Outlier Data

```
for col in data:
    # Computing IQR
    Q1 = data[col].describe()[4]
    Q3 = data[col].describe()[6]
    IQR = data[col].describe()[5]
    lower = Q1 - 1.5 * IQR
    upper = Q3 + 1.5 * IQR
    for row in data.index:
        item = data.loc[row][col]
        if item < lower or item > upper:
            data.drop(row, inplace=True)
```

انجام عملیات های ریاضی بر داده های موجود

برای بررسی و پیدا کردن دید کلی نسبت دیتا، عملیات های ریاضی به ترتیب: مینیمم ، ماکسیسمم، میانه، میانگین، انحراف معیار و واریانس بر روی دیتا ها انجام شد.

Calculate Minimum, Maximum, Median, Average, Standard deviation and Variance

```
res = []
for column in data:
    row = {
        "minimum": data[column].min(),
        "maximum": data[column].max(),
        "median": data[column].median(),
        "average": data[column].mean(),
        "standard-deviation": data[column].std(),
        "variance": data[column].var()
    }
    res.append(row)

resDf = pd.DataFrame(res, index=[col for col in data])
```

	minimum	maximum	median	average	standard-deviation	variance
A1	0.000	10.000	2.000	2.755020	2.570032	6.605065
A2	56.000	195.000	107.000	111.843373	26.154233	684.043918
A3	24.000	110.000	70.000	69.008032	12.471578	155.540258
A4	10.000	60.000	27.000	27.726908	10.526355	110.804152
A5	15.000	180.000	94.000	97.819277	41.873122	1753.358337
A6	18.200	67.100	31.600	31.922088	6.951882	48.328663
A7	0.085	1.096	0.412	0.441948	0.228017	0.051992
A8	21.000	81.000	26.000	28.779116	8.736938	76.334078
output	0.000	1.000	0.000	0.216867	0.412942	0.170521

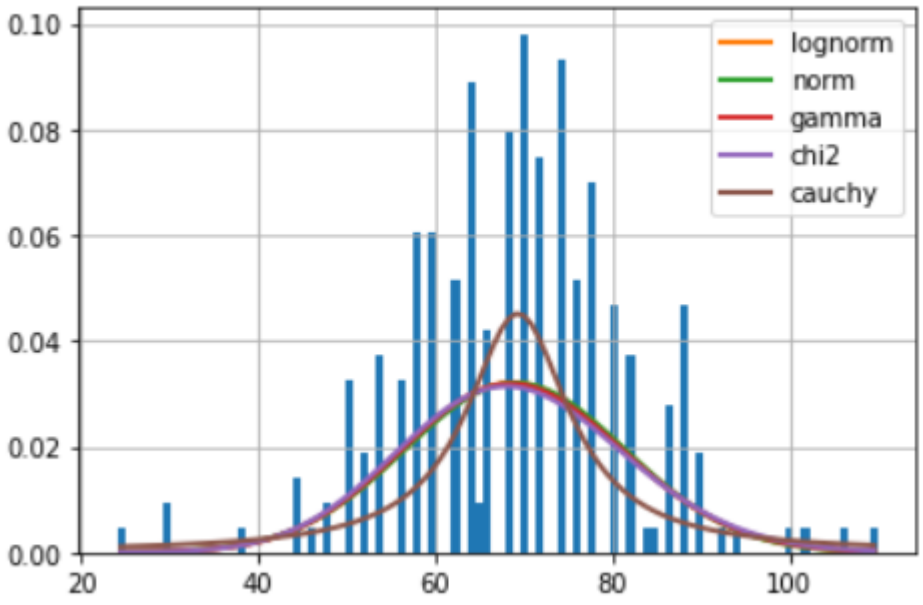
رسم توزیع آماری

برای رسم توزیع آماری ابتدا توسط کتابخانه fitter بهترین روش های توزیع آماری بدست آورده شد و سپس نمودار آنها نیز رسم شد.

Draw Distribution

```
for col in data:
    distribut = Fitter(data[col], distributions=get_common_distributions())
    distribut.fit()
    distribut.summary()
    plt.show()
```

نمونه ای از نمودارهای خروجی



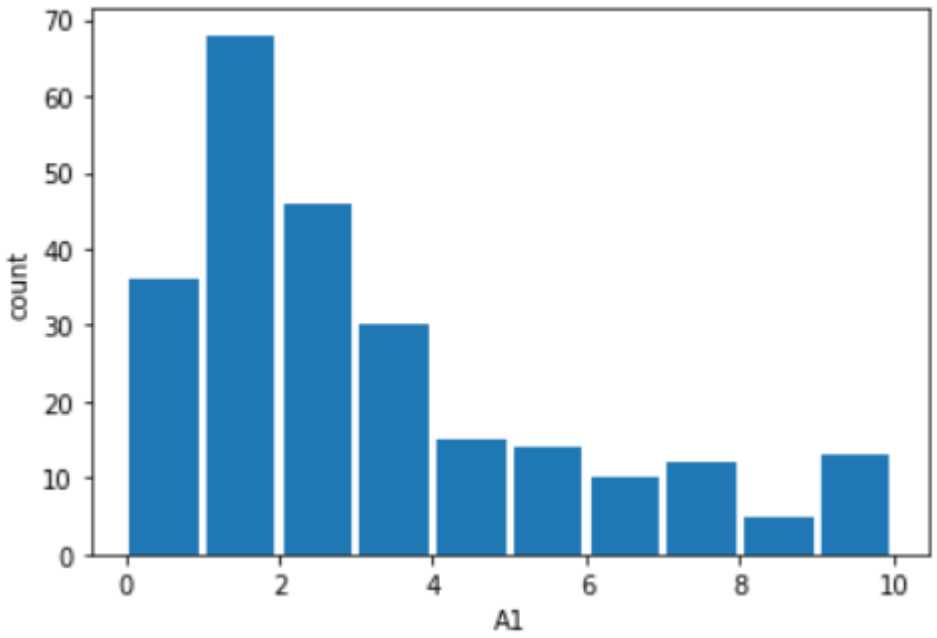
رسم هیستوگرام

برای بررسی دقیق تر و بهتر داده ها نمودار هیستوگرام مربوط به هر یک از ویژگی ها نیز رسم شدند.

Draw Histogram

```
for column in data:
    plt.hist(data[column], rwidth=0.9)
    plt.xlabel(column)
    plt.ylabel("count")
    plt.show()
```

نمونه ای نمودارهای رسم شده:



مقاسیه و نرمالایز کردن داده ها:

در این قسمت ابتدا داده ها مقایسه و نرمالایز شدند و دوباره عملیات های ریاضی مثل میانگین، واریانس، انحراف معیار و ... بر روی آنها اعمال شد.

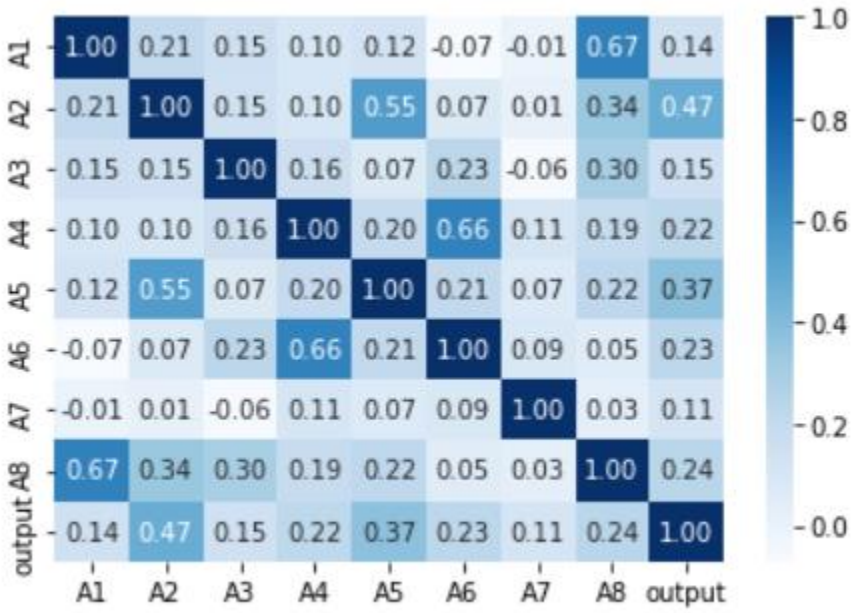
```
normalized_df = (data-data.min())/(data.max()-data.min())
```

	minimum	maximum	median	average	standard-deviation	variance
A1	0.0	1.0	0.200000	0.275502	0.257003	0.066051
A2	0.0	1.0	0.366906	0.401751	0.188160	0.035404
A3	0.0	1.0	0.534884	0.523349	0.145018	0.021030
A4	0.0	1.0	0.340000	0.354538	0.210527	0.044322
A5	0.0	1.0	0.478788	0.501935	0.253776	0.064403
A6	0.0	1.0	0.274029	0.280615	0.142165	0.020211
A7	0.0	1.0	0.323442	0.353064	0.225536	0.050866
A8	0.0	1.0	0.083333	0.129652	0.145616	0.021204
output	0.0	1.0	0.000000	0.216867	0.412942	0.170521

مقایسه همبستگی(کارولویشن) بین ویژگی های مختلف

در این قسمت همبستگی یا کارولویشن داده ها توسط کتابخانه seaborn محاسبه و رسم شدند.

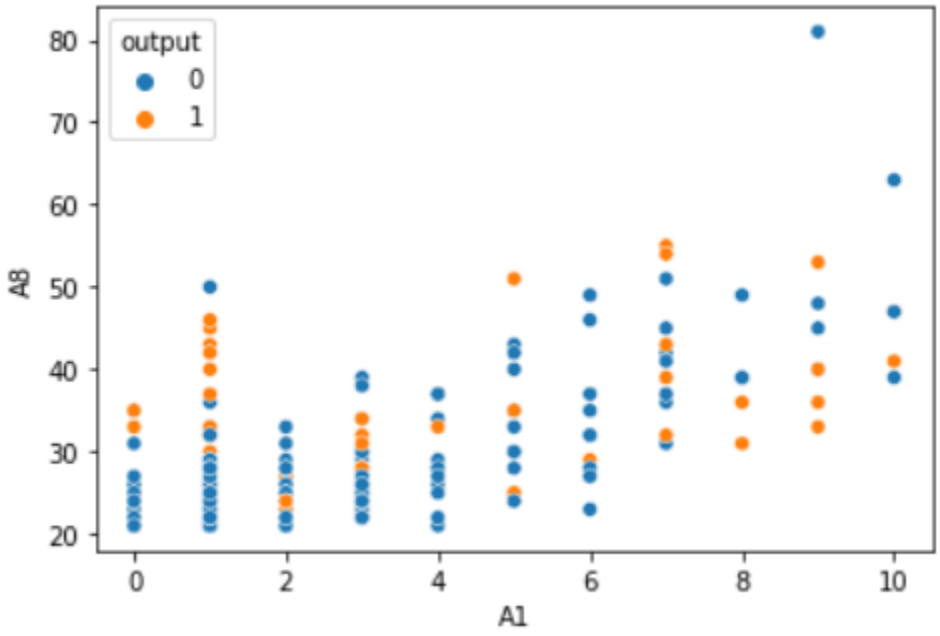
```
sn.heatmap(data.corr(), annot=True, fmt=".2f", cmap="Blues")
plt.show()
```



ترسیم scatter plot برای مقایسه داده ها

در این قسمت ابتدا نمودار scatter plot مربوط به output را به ازای ویژگی های A1(تعداد دفعات بارداری) و A8(سن) رسم شد.

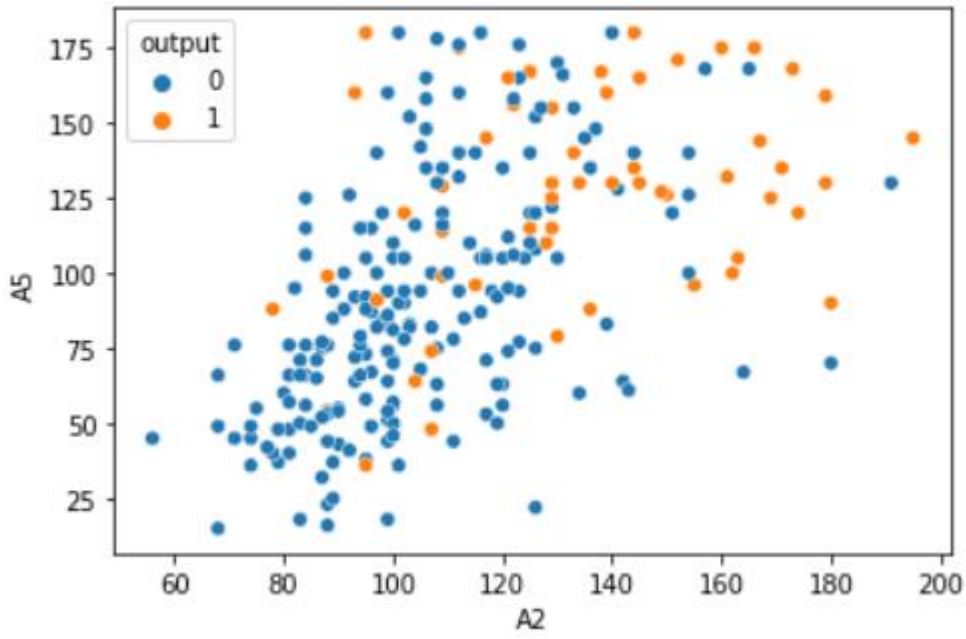
```
plt.clf()
sn.scatterplot(data=data, x="A1", y="A8", hue="output")
plt.show()
```



همانطور که در نمودار میبینید، با افزایش تعداد بارداری و به ویژه سن افراد، احتمال ابتلا به دیابت افزایش پیدا میکند.

با توجه بیشتر به نمودار میبیبینیم که افراد بالای 40 سال بیشتر به ابتلا به این بیماری نزدیکند که مراقبت های لازم را باید بیشتر در نظر گیرند.

برای بررسی های بهتر نمودارهای مختلف از جمله نمودار مربوط به A2(غلظت گلوکز) و A5(انسولین) نیز رسم شدند.



همانطور که در نمودار هم مشاهده میکنید با افزایش غلظت گلوکز و انسولین خطر ابتلا به دیابت در افراد بیشتر میشود.

با بررسی دقیق تر میبینید که افرادی که غلظت گلوکز بالای 120 و همچنین انسولین بالای 100 دارند احتمال خیلی بیشتری در ابتلا دارند.