

In [1]:

```
import pandas as pd
import numpy as np
```

In [2]:

```
diagnosis = pd.read_csv(r'E:\MIMIC\eICU\eICU\diagnosis.csv')
diagnosis.head()
```

Out[2]:

	diagnosisid	patientunitstayid	activeupondischarge	diagnosisoffset	diagnosisstring	icd9code	diagnosispriority
0	4222318	141168	False	72	cardiovascular chest pain / ASHD coronary arte...	414.00, I25.10	Other
1	3370568	141168	True	118	cardiovascular ventricular disorders cardiomyo...	NaN	Other
2	4160941	141168	False	72	pulmonary disorders of the airways COPD	491.20, J44.9	Other
3	4103261	141168	True	118	pulmonary disorders of the airways COPD	491.20, J44.9	Other
4	3545241	141168	True	118	cardiovascular ventricular disorders congestiv...	428.0, I50.9	Other

In [3]:

```
sepsis= diagnosis[diagnosis['diagnosisstring'].str.contains('sepsis')]
sepsis.head()
```

Out[3]:

	diagnosisid	patientunitstayid	activeupondischarge	diagnosisoffset	diagnosisstring	icd9code	diagnosispriority
29	3803853	141227	True	18	cardiovascular shock / hypotension sepsis	038.9, A41.9	Primary
54	3486058	141288	False	575	cardiovascular shock / hypotension sepsis	038.9, A41.9	Primary
58	3831497	141288	False	19	cardiovascular shock / hypotension sepsis	038.9, A41.9	Primary
64	3814467	141288	False	415	cardiovascular shock / hypotension sepsis	038.9, A41.9	Primary
68	4091397	141288	False	899	cardiovascular shock / hypotension sepsis	038.9, A41.9	Primary

In [4]:

```
sepsis.shape
```

Out[4]:

```
(115375, 7)
```

In [5]:

```
sepsis1=sepsis[['patientunitstayid','diagnosisoffset','diagnosisstring']]
sepsis1.head()
```

Out[5]:

	patientunitstayid	diagnosisoffset	diagnosisstring
29	141227	18	cardiovascular shock / hypotension sepsis
54	141288	575	cardiovascular shock / hypotension sepsis
58	141288	19	cardiovascular shock / hypotension sepsis
64	141288	415	cardiovascular shock / hypotension sepsis
68	141288	899	cardiovascular shock / hypotension sepsis

In [6]:

```
sepsis2 = sepsis1.groupby(['patientunitstayid'])['diagnosisoffset'].idxmin()
sepsis3 = sepsis1.loc[sepsis2]
```

In [7]:

```
sepsis3.head()
```

Out[7]:

	patientunitstayid	diagnosisoffset	diagnosisstring
29	141227	18	cardiovascular shock / hypotension sepsis
58	141288	19	cardiovascular shock / hypotension sepsis
74	141289	13	cardiovascular shock / hypotension sepsis
83	141297	25	cardiovascular shock / hypotension sepsis
88	141304	23	cardiovascular shock / hypotension sepsis

In [8]:

```
sepsis3.shape
```

Out[8]:

```
(23479, 3)
```

In [9]:

```
sepsis3.patientunitstayid.nunique()
```

Out[9]:

```
23479
```

In [20]:

```
icustay_detai = pd.read_parquet(r'E:\MIMIC\MIMIC_BIG_DATA\MIMIC_BIG_DATA\MIMIC_IV_Parquet\eICU\icustay_detail.parquet')
icustay_detai.head()
```

Out[20]:

	uniquepid	patienthealthsystemstayid	patientunitstayid	unitvisitnumber	hospitalid	region	unitttype	hospitaladmitoffset	hospitaldischargeoffset	unitadmitoffset	...	apache_iv	hospitaldischarg
0	NaN	193705	224606	1	71	NaN	NaN	-179	8183	0	...	NaN	
1	NaN	178200	204602	1	66	NaN	NaN	-4854	18918	0	...	57.0	
2	NaN	141169	157016	1	63	NaN	NaN	-3331	4172	0	...	29.0	
3	NaN	183274	211144	1	71	NaN	NaN	-67	9697	0	...	NaN	
4	NaN	190893	221005	1	71	NaN	NaN	-2140	7772	0	...	42.0	

5 rows × 21 columns

In [21]:

```
df= sepsis3.merge(icustay_detai, on=['patientunitstayid'], how='left')
```

In [22]:

```
df.shape
```

Out[22]:

```
(23479, 23)
```

住院大于48h

In []:

```
df1=df.query('icu_los_hours >= 48')
df1.shape
```

Out[]:

```
(5935, 69)
```

年龄大于18

In [26]:

```
df2=df1.query('age >= 18')
df2.shape
```

Out[26]:

```
(22391, 23)
```

第一次住ICU

In []:

```
df3=df2.query('unitvisitnumber == 1')
df3.shape
```

Out[]:

```
(19092, 23)
```

生存率大于7天

```
In [ ]: df3['hospitaldischargeoffset'] = pd.to_numeric(df3['hospitaldischargeoffset'], errors='coerce')

# 使用布尔索引来筛选不满足条件的行
df4= df3[~((df3['hosp_mort'] == 1) & (df3['hospitaldischargeoffset'] < 10080))]

# filtered_df现在包含了不满足条件的行

In [19]: df4.shape

Out[19]: (9786, 23)

In [29]: labsfirstday= pd.read_csv(r'E:\MIMIC\eICU\eICU\labsfirstday.csv')
labsfirstday.head()
```

	uniquepid	patienthealthsystemstayid	patientunitstayid	aniongap_min	aniongap_max	albumin_min	albumin_max	bands_min	bands_max	bicarbonate_min	...	inr_min	inr_max	pt_min	pt_max
0	002-10009	193705	224606	9.0	22.0	2.2	2.2	1.0	9.0	18.0	...	1.5	1.5	16.1	16.1
1	002-10018	178200	204602	12.0	19.0	1.8	2.0	1.0	44.0	17.0	...	NaN	NaN	NaN	NaN
2	002-10034	141169	157016	11.0	16.0	2.0	2.2	NaN	NaN	24.0	...	NaN	NaN	NaN	NaN
3	002-10050	183274	211144	10.0	10.0	NaN	NaN	NaN	NaN	30.0	...	1.0	1.1	10.7	11.0
4	002-10050	190893	221004	12.0	12.0	NaN	NaN	NaN	NaN	28.0	...	1.0	1.0	10.3	10.3

5 rows × 43 columns

```
In [30]: print(np.intersect1d(df4.columns,labsfirstday.columns))

['patienthealthsystemstayid' 'patientunitstayid' 'uniquepid']

In [31]: labsfirstday.drop(['patienthealthsystemstayid', 'uniquepid'], axis=1, inplace=True)

In [32]: print(np.intersect1d(df4.columns,labsfirstday.columns))

['patientunitstayid']

In [33]: df5= df4.merge(labsfirstday, on=['patientunitstayid'], how='left')

In [34]: df5.shape

Out[34]: (19092, 63)

In [35]: lab= pd.read_csv(r'E:\MIMIC\eICU\eICU\lab.csv')
lab.head()
```

	labid	patientunitstayid	labresultoffset	labtypeid	labname	labresult	labresulttext	labmeasurenamesystem	labmeasurenameinterface	labresultrevisedoffset
0	52307161	141168	2026	3	fibrinogen	177.0	177	mg/dL	mg/dL	2219
1	50363251	141168	1133	3	PT - INR	2.5	2.5	ratio	NaN	1208
2	49149139	141168	2026	1	magnesium	2.0	2.0	mg/dL	mg/dL	2090
3	50363250	141168	1133	3	PT	26.6	26.6	sec	sec	1208
4	66695374	141168	2141	7	pH	7.2	7.20	NaN	Units	2155

```
In [36]: creatinine= lab[lab['labname'].str.contains('creatinine')]
creatinine.head()
```

	labid	patientunitstayid	labresultoffset	labtypeid	labname	labresult	labresulttext	labmeasurenamesystem	labmeasurenameinterface	labresultrevisedoffset
6	58077793	141168	1701	4	urinary creatinine	173.12	173.12	mg/dL	mg/dL	1757
54	49485523	141168	2026	1	creatinine	2.95	2.95	mg/dL	mg/dL	2090
91	43871514	141168	1133	1	creatinine	2.30	2.30	mg/dL	mg/dL	1240
108	47753719	141168	516	1	creatinine	1.95	1.95	mg/dL	mg/dL	540
159	48867924	141178	-280	1	creatinine	0.70	0.70	mg/dL	mg/dL	-212

```
In [37]: creatinineonly = creatinine[creatinine['labname'].str.fullmatch('creatinine', na=False)]

In [38]: creatinineonly.head()
```

	labid	patientunitstayid	labresultoffset	labtypeid	labname	labresult	labresulttext	labmeasurenamesystem	labmeasurenameinterface	labresultrevisedoffset
54	49485523	141168	2026	1	creatinine	2.95	2.95	mg/dL	mg/dL	2090
91	43871514	141168	1133	1	creatinine	2.30	2.30	mg/dL	mg/dL	1240
108	47753719	141168	516	1	creatinine	1.95	1.95	mg/dL	mg/dL	540
159	48867924	141178	-280	1	creatinine	0.70	0.70	mg/dL	mg/dL	-212
196	49190158	141179	82	1	creatinine	0.70	0.70	mg/dL	mg/dL	106

```
In [39]: creatinine1 = creatinineonly.dropna(subset=['labresult'])

In [40]: creatinine2=creatinine1[['patientunitstayid','labresultoffset','labresult']]
creatinine2.head()
```

	patientunitstayid	labresultoffset	labresult
54	141168	2026	2.95
91	141168	1133	2.30
108	141168	516	1.95
159	141178	-280	0.70
196	141179	82	0.70

```
In [ ]: creatinine2.rename(columns={'labresult': 'creatinine'}, inplace=True)
creatinine2.head()
```

```
In [42]: creatininemax = creatinine2.groupby(['patientunitstayid'])['creatinine'].idxmax()
creatininemax1= creatinine2.loc[creatininemax]
creatininemax1.head()
```

```
Out[42]:
```

	patientunitstayid	labresultoffset	creatinine
54	141168	2026	2.95
159	141178	-280	0.70
196	141179	82	0.70
461	141194	-923	2.94
547	141196	3600	0.89

```
In [43]: creatininemax1.rename(columns={'creatinine': 'creatininemax'}, inplace=True)
```

```
In [44]: creatininemax1.rename(columns={'labresultoffset': 'labresultoffsetmax'}, inplace=True)
```

```
In [45]: creatininemax1.head()
```

```
Out[45]:
```

	patientunitstayid	labresultoffsetmax	creatininemax
54	141168	2026	2.95
159	141178	-280	0.70
196	141179	82	0.70
461	141194	-923	2.94
547	141196	3600	0.89

```
In [46]: creatininemin = creatinine2.groupby(['patientunitstayid'])['creatinine'].idxmin()
creatinininemin1= creatinine2.loc[creatinininemin]
creatinininemin1.head()
```

```
Out[46]:
```

	patientunitstayid	labresultoffset	creatinine
108	141168	516	1.95
159	141178	-280	0.70
196	141179	82	0.70
224	141194	10397	1.74
584	141196	628	0.80

```
In [47]: creatininemin1.rename(columns={'creatinine': 'creatinininemin'}, inplace=True)
```

```
In [48]: creatininemin1.rename(columns={'labresultoffset': 'labresultoffsetmin'}, inplace=True)
```

```
In [49]: creatininemin1.head()
```

```
Out[49]:
```

	patientunitstayid	labresultoffsetmin	creatinininemin
108	141168	516	1.95
159	141178	-280	0.70
196	141179	82	0.70
224	141194	10397	1.74
584	141196	628	0.80

```
In [50]: scr= pd.merge(creatinininemin1, creatininemax1, on=['patientunitstayid'])
scr.head()
```

```
Out[50]:
```

	patientunitstayid	labresultoffsetmin	creatinininemin	labresultoffsetmax	creatininemax
0	141168	516	1.95	2026	2.95
1	141178	-280	0.70	-280	0.70
2	141179	82	0.70	82	0.70
3	141194	10397	1.74	-923	2.94
4	141196	628	0.80	3600	0.89

```
In [51]: scr['creatinine_diff'] = scr['creatininemax'] - scr['creatinininemin']
selected_rows = scr[scr['creatinine_diff'] >= 0.3]
selected_rows.head()
```

```
Out[51]:
```

	patientunitstayid	labresultoffsetmin	creatinininemin	labresultoffsetmax	creatininemax	creatinine_diff
0	141168	516	1.95	2026	2.95	1.00
3	141194	10397	1.74	-923	2.94	1.20
6	141203	-1580	0.33	1091	0.71	0.38
8	141227	-1566	1.40	1362	1.90	0.50
10	141233	11142	0.80	19652	1.10	0.30

```
In [52]: aki=selected_rows.query('creatininemax > 1.5')
aki.head()
```

```
Out[52]:
```

	patientunitstayid	labresultoffsetmin	creatinininemin	labresultoffsetmax	creatininemax	creatinine_diff
0	141168	516	1.95	2026	2.95	1.00
3	141194	10397	1.74	-923	2.94	1.20
8	141227	-1566	1.40	1362	1.90	0.50
16	141266	6557	1.20	-703	1.90	0.70
21	141296	-1827	4.00	3933	7.16	3.16

```
In [53]: aki.shape
```

```
Out[53]: (51992, 6)
```

```
In [54]: aki.patientunitstayid.nunique()
```

```
Out[54]: 51992
```

```
In [55]: aki1=aki[['patientunitstayid']]
```

```
In [56]: aki2= aki1.merge(creatinine2, on=['patientunitstayid'], how='left')
```

```
In [57]: aki2.head()
```

Out[57]:

	patientunitstayid	labresultoffset	creatinine
0	141168	2026	2.95
1	141168	1133	2.30
2	141168	516	1.95
3	141194	11658	1.75
4	141194	10397	1.74

In [58]:

aki2.shape

Out[58]:

(549353, 3)

In [59]:

aki2.patientunitstayid.nunique()

Out[59]:

51992

In [60]:

df6=df5[['patientunitstayid']]

In [61]:

patient_ids_in_scr2 = aki2['patientunitstayid'].unique()
saki= df6[df6['patientunitstayid'].isin(patient_ids_in_scr2)]

In [62]:

saki.head()

Out[62]:

	patientunitstayid
0	141227
2	141304
3	141392
6	141470
7	141751

In [63]:

saki.shape

Out[63]:

(9208, 1)

In [64]:

saki.patientunitstayid.nunique()

Out[64]:

9208

In [65]:

saki1=saki.merge(aki2, on=['patientunitstayid'], how='left')
saki1.head()

Out[65]:

	patientunitstayid	labresultoffset	creatinine
0	141227	1362	1.9
1	141227	-1566	1.4
2	141227	-13	1.5
3	141227	937	1.6
4	141227	-128	1.4

In [66]:

saki1.shape

Out[66]:

(113697, 3)

In [67]:

saki1.patientunitstayid.nunique()

Out[67]:

9208

In [68]:

creatinine_min=saki1.groupby(['patientunitstayid'])['creatinine'].idxmin()
Scr_baseline= saki1.loc[creatinine_min]
Scr_baseline.head()

Out[68]:

	patientunitstayid	labresultoffset	creatinine
1	141227	-1566	1.40
9	141304	5800	1.24
17	141392	7802	0.76
21	141470	8864	0.92
40	141751	10390	0.90

In [69]:

Scr_baseline.rename(columns={'creatinine':'Scr_baseline'}, inplace=True)
Scr_baseline.head()

Out[69]:

	patientunitstayid	labresultoffset	Scr_baseline
1	141227	-1566	1.40
9	141304	5800	1.24
17	141392	7802	0.76
21	141470	8864	0.92
40	141751	10390	0.90

In [70]:

Scr_baseline.drop(['labresultoffset'], axis=1, inplace=True)

In [71]:

saki2=saki1.query('creatinine > 1')
saki2.shape

Out[71]:

(91529, 3)

In [72]:

akimintime = saki2.groupby(['patientunitstayid'])['labresultoffset'].idxmin()
akimintime1= saki2.loc[akimintime]
akimintime1.head()

Out[72]:

	patientunitstayid	labresultoffset	creatinine
1	141227	-1566	1.40
5	141304	-127	2.34
19	141392	-468	1.89
23	141470	-271	2.44
39	141751	-101	2.10

In [73]:

akimintime1.shape

Out[73]:

(9208, 3)

In [74]:

akimintime1.rename(columns={'labresultoffset': 'akimintime'}, inplace=True)
akimintime1.head()

Out[74]:

	patientunitstayid	akimintime	creatinine
1	141227	-1566	1.40
5	141304	-127	2.34
19	141392	-468	1.89
23	141470	-271	2.44
39	141751	-101	2.10

In [75]:

akimaxtime = saki2.groupby(['patientunitstayid'])['labresultoffset'].idxmax()
akimaxtime1= saki2.loc[akimaxtime]
akimaxtime1.shape

Out[75]:

(9208, 3)

In [76]:

akimaxtime1.rename(columns={'labresultoffset': 'akimaxtime'}, inplace=True)
akimaxtime1.head()

Out[76]:

	patientunitstayid	akimaxtime	creatinine
0	141227	1362	1.90
11	141304	7293	1.29
18	141392	1639	1.09
22	141470	6083	1.06
27	141751	6042	1.10

In [77]:

akd= pd.merge(akimintime1,akimaxtime1,on=['patientunitstayid'])
akd.head()

Out[77]:

	patientunitstayid	akimintime	creatinine_x	akimaxtime	creatinine_y
0	141227	-1566	1.40	1362	1.90
1	141304	-127	2.34	7293	1.29
2	141392	-468	1.89	1639	1.09
3	141470	-271	2.44	6083	1.06
4	141751	-101	2.10	6042	1.10

In [78]:

akd.drop(['creatinine_x', 'creatinine_y'], axis=1, inplace=True)
akd.head()

Out[78]:

	patientunitstayid	akimintime	akimaxtime
0	141227	-1566	1362
1	141304	-127	7293
2	141392	-468	1639
3	141470	-271	6083
4	141751	-101	6042

In [79]:

akd['los_aki']=akd['akimaxtime']-akd['akimintime']

In [80]:

akd.head()

Out[80]:

	patientunitstayid	akimintime	akimaxtime	los_aki
0	141227	-1566	1362	2928
1	141304	-127	7293	7420
2	141392	-468	1639	2107
3	141470	-271	6083	6354
4	141751	-101	6042	6143

In [81]:

akd['AKD'] = (akd['los_aki'] > 10080).astype(int)
akd.head()

Out[81]:

	patientunitstayid	akimintime	akimaxtime	los_aki	AKD
0	141227	-1566	1362	2928	0
1	141304	-127	7293	7420	0
2	141392	-468	1639	2107	0
3	141470	-271	6083	6354	0
4	141751	-101	6042	6143	0

In [82]:

akd.AKD.value_counts()

Out[82]:

AKD
0 5939
1 3269
Name: count, dtype: int64

In [83]:

akd['cxx_aki'] = (akd['los_aki'] > 2880).astype(int)
akd.head()

Out[83]:

	patientunitstayid	akimintime	akimaxtime	los_aki	AKD	cxx_aki
0	141227	-1566	1362	2928	0	1
1	141304	-127	7293	7420	0	1
2	141392	-468	1639	2107	0	0
3	141470	-271	6083	6354	0	1
4	141751	-101	6042	6143	0	1

In [84]:

akd.cxx_aki.value_counts()

```
Out[84]: cxx_aki
1      6778
0      2430
Name: count, dtype: int64
```

```
In [85]: saki3=pd.merge(Scr_baseline,akd,on=['patientunitstayid'])
saki3.head()
```

	patientunitstayid	Scr_baseline	akimintime	akimaxtime	los_aki	AKD	cxx_aki
0	141227	1.40	-1566	1362	2928	0	1
1	141304	1.24	-127	7293	7420	0	1
2	141392	0.76	-468	1639	2107	0	0
3	141470	0.92	-271	6083	6354	0	1
4	141751	0.90	-101	6042	6143	0	1

```
In [86]: saki3.shape
```

```
Out[86]: (9208, 7)
```

```
In [87]: print(np.intersect1d(saki3.columns,df5.columns))
```

['patientunitstayid']

```
In [88]: df7=saki3.merge(df5, on=['patientunitstayid'], how='left')
df7.head()
```

	patientunitstayid	Scr_baseline	akimintime	akimaxtime	los_aki	AKD	cxx_aki	diagnosisoffset	diagnosisstring	uniquepid	...	inr_min	inr_max	pt_min	pt_max	sodium_min	sodium_max
0	141227	1.40	-1566	1362	2928	0	1	18	cardiovascular shock / hypotension sepsis	NaN	...	1.2	1.2	13.3	13.3	132.0	134.0
1	141304	1.24	-127	7293	7420	0	1	23	cardiovascular shock / hypotension sepsis	NaN	...	1.0	1.0	10.3	10.3	140.0	141.0
2	141392	0.76	-468	1639	2107	0	0	38	cardiovascular shock / hypotension sepsis	NaN	...	2.7	3.2	29.7	34.9	135.0	137.0
3	141470	0.92	-271	6083	6354	0	1	13	cardiovascular shock / hypotension sepsis	NaN	...	NaN	NaN	NaN	NaN	131.0	138.0
4	141751	0.90	-101	6042	6143	0	1	45	cardiovascular shock / hypotension signs and s...	NaN	...	1.8	1.8	18.7	18.7	109.0	143.0

5 rows × 69 columns



```
In [89]: df7.shape
```

```
Out[89]: (9208, 69)
```

CRRT

```
In [81]: treatment= pd.read_csv(r'E:\MIMIC\eICU\eICU\treatment.csv')
treatment.head()
```

	treatmentid	patientunitstayid	treatmentoffset	treatmentstring	activeupondischarge
0	8399138	242040	198	cardiovascular hypertension angiotensin II rec...	False
1	8626134	242040	198	cardiovascular myocardial ischemia / infarctio...	False
2	8517569	242040	198	infectious diseases medications therapeutic an...	False
3	9597686	242040	616	cardiovascular non-operative procedures diagno...	False
4	9334096	242040	618	infectious diseases medications therapeutic an...	True

```
In [82]: CRRT= treatment[treatment['treatmentstring'].str.contains('dialysis')]
CRRT.head()
```

	treatmentid	patientunitstayid	treatmentoffset	treatmentstring	activeupondischarge
637	8619386	242787	1473	renal dialysis hemodialysis for chronic renal ...	True
649	8711073	242787	672	renal dialysis hemodialysis for chronic renal ...	False
655	9353682	242787	1230	renal dialysis hemodialysis for chronic renal ...	False
1307	9638036	243334	1038	renal dialysis hemodialysis for chronic renal ...	False
1324	8133609	243334	126	renal dialysis hemodialysis for chronic renal ...	False

```
In [83]: CRRT1=CRRT[['patientunitstayid','treatmentoffset']]
```

```
In [84]: CRRT2= CRRT1.groupby(['patientunitstayid'])['treatmentoffset'].min().reset_index()
```

```
In [85]: CRRT2.head()
```

	patientunitstayid	treatmentoffset
0	242787	672
1	243334	126
2	243709	2137
3	243838	144
4	244150	2474

```
In [86]: df7['window_start7']=df7['akimintime'] + 10080
```

```
In [87]: merged_df1 = df7.merge(CRRT2, on=['patientunitstayid'], how='left')
merged_df1['CRRT'] = (merged_df1['treatmentoffset'] < merged_df1['window_start7']).astype(int)

# 保留 df11 原本的数据
df8 = df7.merge(merged_df1[['patientunitstayid', 'CRRT']], on=['patientunitstayid'], how='left')

df8.head()
```

Out[87]:

	patientunitstayid	Scr_baseline	akimintime	akimaxtime	los_aki	AKD	cxr_aki	diagnosisoffset	diagnosisstring	uniquepid	...	pt_min	pt_max	sodium_min	sodium_max	bun_min	bun_max
0	141304	1.24	-127	7293	7420	0	1	23	cardiovascular shock / hypotension sepsis	NaN	...	10.3	10.3	140.0	141.0	41.0	48.0
1	141751	0.90	-101	6042	6143	0	1	45	cardiovascular shock / hypotension signs and s...	NaN	...	18.7	18.7	109.0	143.0	39.0	62.0
2	141920	1.03	-113	2469	2582	0	0	15	cardiovascular shock / hypotension sepsis	NaN	...	NaN	NaN	134.0	138.0	32.0	34.0
3	141945	2.17	-205	6539	6744	0	1	25	cardiovascular shock / hypotension sepsis	NaN	...	10.8	11.4	138.0	139.0	23.0	54.0
4	142388	1.11	-94	7122	7216	0	1	73	cardiovascular shock / hypotension sepsis	NaN	...	9.7	9.7	133.0	139.0	17.0	21.0

5 rows × 71 columns

In [88]:

df8.CRRT.value_counts()

Out[88]:

CRRT
0 4385
1 920
Name: count, dtype: int64

In [89]:

apachepatientresult= pd.read_csv(r'E:\MIMIC\eICU\eICU\apachepatientresult.csv')
apachepatientresult.head()

Out[89]:

	apachepatientresultsid	patientunitstayid	physicianspeciality	physicianinterventioncategory	acutephysiologyscore	apachescore	apacheversion	predictedicumortality	actualicumortality	predictedic
0	26570	141168	critical care medicine (CCM)	Unknown	49	65	IV	0.026988	EXPIRED	3.0
1	26571	141168	critical care medicine (CCM)	Unknown	49	65	IVa	0.028889	EXPIRED	3.0
2	53135	141194	critical care medicine (CCM)	Unknown	57	70	IV	0.037888	ALIVE	4.6
3	53136	141194	critical care medicine (CCM)	Unknown	57	70	IVa	0.046448	ALIVE	4.1
4	8	141203	hospitalist	I	73	90	IVa	0.291609	ALIVE	8.6

5 rows × 23 columns

In [90]:

print(list(apachepatientresult.columns))

['apachepatientresultsid', 'patientunitstayid', 'physicianspeciality', 'physicianinterventioncategory', 'acutephysiologyscore', 'apachescore', 'apacheversion', 'predictedicumortality', 'actualicumortality', 'predictediculos', 'actualiculos', 'predictedhospitalmortality', 'actualhospitalmortality', 'predictedhospitallos', 'actualhospitallos', 'preopmi', 'preopcardiaccath', 'ptcawithin24h', 'unabridgedunitlos', 'unabridgedhospos', 'actualventdays', 'predventdays', 'unabridgedactualventdays']

In [91]:

print(np.intersect1d(df8.columns,apachepatientresult.columns))

['patientunitstayid']

In [92]:

apachepatientresult1 = apachepatientresult.groupby(['patientunitstayid'])['apachepatientresultsid'].idxmin()
apachepatientresult2= apachepatientresult.loc[apachepatientresult1]
apachepatientresult2.shape

Out[92]:

(148532, 23)

In [93]:

df9=df8.merge(apachepatientresult2, on=['patientunitstayid'], how='left')

In [94]:

df9.shape

Out[94]:

(5305, 93)

In [95]:

apacheapsvar= pd.read_csv(r'E:\MIMIC\eICU\eICU\apacheapsvar.csv')
apacheapsvar.head()

Out[95]:

	apacheapsvarid	patientunitstayid	intubated	vent	dialysis	eyes	motor	verbal	meds	urine	...	ph	hematocrit	creatinine	albumin	pao2	pco2	bun	glucose	bilirubin	fio2
0	2	141168	0	0	0	4	6	5	0	-1.0	...	-1.00	40.1	2.30	3.1	-1.0	-1.0	27.0	95.0	4.1	-1.0
1	2399659	141178	0	0	0	-1	-1	-1	-1	-1.0	...	-1.00	-1.0	-1.00	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
2	167885	141194	0	0	0	3	6	4	0	-1.0	...	-1.00	27.4	2.51	2.3	-1.0	-1.0	31.0	168.0	0.4	-1.0
3	70691	141197	0	0	0	4	6	5	0	-1.0	...	-1.00	-1.0	-1.00	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
4	2279689	141203	0	1	0	1	3	1	0	-1.0	...	7.45	36.9	0.56	-1.0	51.0	37.0	9.0	145.0	-1.0	100.0

5 rows × 26 columns

In [96]:

print(list(apacheapsvar.columns))

['apacheapsvarid', 'patientunitstayid', 'intubated', 'vent', 'dialysis', 'eyes', 'motor', 'verbal', 'meds', 'urine', 'wbc', 'temperature', 'respiratoryrate', 'sodium', 'heartrate', 'meanbp', 'ph', 'hematocrit', 'creatinine', 'albumin', 'pao2', 'pco2', 'bun', 'glucose', 'bilirubin', 'fio2']

In [97]:

print(np.intersect1d(df9.columns,apacheapsvar.columns))

['patientunitstayid']

In [98]:

apacheapsvar1 = apacheapsvar.groupby(['patientunitstayid'])['apacheapsvarid'].idxmin()
apacheapsvar2= apacheapsvar.loc[apacheapsvar1]
apacheapsvar2.shape

Out[98]:

(171177, 26)

In [99]:

df10=df9.merge(apacheapsvar2, on=['patientunitstayid'], how='left')

In [100]:

df10.shape

Out[100]:

(5305, 118)

In [101]:

oasis= pd.read_csv(r'E:\MIMIC\eICU\eICU\pivoted_oasis.csv')
oasis.head()

Out[101...

	patientunitstayid	pre_icu_los_oasis	age_oasis	gcs_oasis	heartrate_oasis	map_oasis	respiratoryrate_oasis	temperature_oasis	urineoutput_oasis	vent_oasis	electivesurgery_oasis	oasis	oasis_prob
0	141168	5	6	NaN	6.0	0.0	0.0	4.0	NaN	0	6	27	0.061104
1	141178	3	3	NaN	0.0	0.0	0.0	0.0	NaN	0	6	12	0.009521
2	141179	3	3	4.0	1.0	2.0	0.0	2.0	1.0	0	6	22	0.033259
3	141194	0	6	0.0	3.0	3.0	1.0	2.0	NaN	0	6	21	0.029391
4	141196	3	6	0.0	1.0	0.0	0.0	2.0	10.0	0	6	28	0.068841

In [102...

```
print(list(oasis.columns))
```

['patientunitstayid', 'pre_icu_los_oasis', 'age_oasis', 'gcs_oasis', 'heartrate_oasis', 'map_oasis', 'respiratoryrate_oasis', 'temperature_oasis', 'urineoutput_oasis', 'vent_oasis', 'electivesurgery_oasis', 'oasis', 'oasis_prob']

In [103...

```
oasis1=oasis[['patientunitstayid','oasis']]
```

In [104...

```
oasis1.head()
```

Out[104...

	patientunitstayid	oasis
0	141168	27
1	141178	12
2	141179	22
3	141194	21
4	141196	28

In [105...

```
oasis1.shape
```

Out[105...

(200859, 2)

In [106...

```
oasis11 = oasis1.groupby(['patientunitstayid'])['oasis'].idxmax()  
oasis12= oasis1.loc[oasis11]  
oasis12.shape
```

Out[106...

(200859, 2)

In [107...

```
df11=df10.merge(oasis12, on=['patientunitstayid'], how='left')  
df11.shape
```

Out[107...

(5305, 119)

In [108...

```
vasopressor= pd.read_csv(r'E:\MIMIC\eICU\pivoted_treatment_vasopressor.csv')  
vasopressor.head()
```

Out[108...

	patientunitstayid	chartoffset	vasopressor
0	242082	107	1
1	242082	240	1
2	242082	903	1
3	242290	162	1
4	242290	165	1

In [109...

```
vasopressor1=vasopressor[['patientunitstayid','chartoffset']]
```

In [110...

```
vasopressor2= vasopressor1.groupby(['patientunitstayid'])['chartoffset'].min().reset_index()
```

In [111...

```
merged_df1 = df11.merge(vasopressor2, on=['patientunitstayid'], how='left')  
merged_df1['vasopressor'] = (merged_df1['chartoffset'] < merged_df1['window_start7']).astype(int)  
  
# 保留 df11 原本的数据  
df12 = df11.merge(merged_df1[['patientunitstayid', 'vasopressor']], on=['patientunitstayid'], how='left')  
  
df12.head()
```

Out[111...

	patientunitstayid	Scr_baseline	akimintime	akimaxtime	los_aki	AKD	cxx_aki	diagnosisoffset	diagnosisstring	uniquepid	...	creatinine	albumin	pao2	pco2	bun	glucose	bilirubin	fio2
0	141304	1.24	-127	7293	7420	0	1	23	cardiovascular shock / hypotension sepsis	NaN	...	2.05	-1.0	-1.0	-1.0	48.0	164.0	-1.0	-1.0
1	141751	0.90	-101	6042	6143	0	1	45	cardiovascular shock / hypotension signs and s...	NaN	...	1.70	1.5	78.0	29.0	51.0	825.0	1.3	40.0
2	141920	1.03	-113	2469	2582	0	0	15	cardiovascular shock / hypotension sepsis	NaN	...	1.43	2.7	-1.0	-1.0	34.0	136.0	0.6	-1.0
3	141945	2.17	-205	6539	6744	0	1	25	cardiovascular shock / hypotension sepsis	NaN	...	2.17	-1.0	-1.0	-1.0	23.0	79.0	-1.0	-1.0
4	142388	1.11	-94	7122	7216	0	1	73	cardiovascular shock / hypotension sepsis	NaN	...	1.48	-1.0	-1.0	-1.0	21.0	398.0	-1.0	-1.0

5 rows × 120 columns

In [112...

```
df12.vasopressor.value_counts()
```

Out[112...

```
vasopressor  
0    3058  
1     2247  
Name: count, dtype: int64
```

In [113...

```
treatment1= pd.read_csv(r'E:\MIMIC\eICU\ICU\treatment.csv')  
treatment1.head()
```

Out[113...

	treatmentid	patientunitstayid	treatmentoffset	treatmentstring	activeupondischarge
0	8399138	242040	198	cardiovascular hypertension angiotensin II rec...	False
1	8626134	242040	198	cardiovascular myocardial ischemia / infarctio...	False
2	8517569	242040	198	infectious diseases medications therapeutic an...	False
3	9597686	242040	616	cardiovascular non-operative procedures diagno...	False
4	9334096	242040	618	infectious diseases medications therapeutic an...	True

In [114...

```
ACEI= treatment1[treatment1['treatmentstring'].str.contains('ACE inhibitor')]  
ACEI.head()
```

Out[114...

	treatmentid	patientunitstayid	treatmentoffset	treatmentstring	activeupondischarge
	350	8327139	242401	1006 cardiovascular hypertension ACE inhibitor lisi...	True
	384	8434432	242476	1140 cardiovascular hypertension ACE inhibitor lisi...	True
	396	8206929	242476	111 cardiovascular hypertension ACE inhibitor lisi...	False
	397	9130356	242476	110 cardiovascular hypertension ACE inhibitor lisi...	False
	866	8004622	243026	1374 cardiovascular hypertension ACE inhibitor lisi...	False

In [115...

ACEI.shape

Out[115...

(13279, 5)

In [116...

ACEI1=ACEI[['patientunitstayid','treatmentoffset']]

In [117...

ACEI2= ACEI1.groupby(['patientunitstayid'])['treatmentoffset'].min().reset_index()

In [118...

```
merged_df1 = df12.merge(ACEI2, on=['patientunitstayid'], how='left')
merged_df1['ACEI'] = (merged_df1['treatmentoffset'] < merged_df1['window_start7']).astype(int)

# 保留 df11 原本的数据
df13 = df12.merge(merged_df1[['patientunitstayid', 'ACEI']], on=['patientunitstayid'], how='left')
df13.head()
```

Out[118...

	patientunitstayid	Scr_baseline	akimintime	akimaxtime	los_aki	AKD	cxx_aki	diagnosisoffset	diagnosisstring	uniquepid	...	albumin	pao2	pco2	bun	glucose	bilirubin	fio2	oasis	vas
0	141304	1.24	-127	7293	7420	0	1	23	cardiovascular shock / hypotension sepsis	NaN	...	-1.0	-1.0	-1.0	48.0	164.0	-1.0	-1.0	28	
1	141751	0.90	-101	6042	6143	0	1	45	cardiovascular shock / hypotension signs and s...	NaN	...	1.5	78.0	29.0	51.0	825.0	1.3	40.0	49	
2	141920	1.03	-113	2469	2582	0	0	15	cardiovascular shock / hypotension sepsis	NaN	...	2.7	-1.0	-1.0	34.0	136.0	0.6	-1.0	35	
3	141945	2.17	-205	6539	6744	0	1	25	cardiovascular shock / hypotension sepsis	NaN	...	-1.0	-1.0	-1.0	23.0	79.0	-1.0	-1.0	20	
4	142388	1.11	-94	7122	7216	0	1	73	cardiovascular shock / hypotension sepsis	NaN	...	-1.0	-1.0	-1.0	21.0	398.0	-1.0	-1.0	54	

5 rows × 121 columns

In [119...

ARB= treatment1[treatment1['treatmentstring'].str.contains('angiotensin II receptor blocker')]
ARB.head()

Out[119...

	treatmentid	patientunitstayid	treatmentoffset	treatmentstring	activeupondischarge
0	8399138	242040	198	cardiovascular hypertension angiotensin II rec...	False
24	9659024	242040	617	cardiovascular hypertension angiotensin II rec...	False
26	10022944	242040	616	cardiovascular hypertension angiotensin II rec...	False
38	9306475	242040	616	cardiovascular hypertension angiotensin II rec...	False
68	8530639	242040	618	cardiovascular hypertension angiotensin II rec...	True

In [120...

ARB.shape

Out[120...

(3400, 5)

In [121...

ARB1=ARB[['patientunitstayid','treatmentoffset']]

In [122...

ARB2= ARB1.groupby(['patientunitstayid'])['treatmentoffset'].min().reset_index()

In [123...

```
merged_df1 = df13.merge(ARB2, on=['patientunitstayid'], how='left')
merged_df1['ARB'] = (merged_df1['treatmentoffset'] < merged_df1['window_start7']).astype(int)

# 保留 df11 原本的数据
df14= df13.merge(merged_df1[['patientunitstayid', 'ARB']], on=['patientunitstayid'], how='left')
df14.head()
```

Out[123...

	patientunitstayid	Scr_baseline	akimintime	akimaxtime	los_aki	AKD	cxx_aki	diagnosisoffset	diagnosisstring	uniquepid	...	pao2	pco2	bun	glucose	bilirubin	fio2	oasis	vasopressor
0	141304	1.24	-127	7293	7420	0	1	23	cardiovascular shock / hypotension sepsis	NaN	...	-1.0	-1.0	48.0	164.0	-1.0	-1.0	28	0
1	141751	0.90	-101	6042	6143	0	1	45	cardiovascular shock / hypotension signs and s...	NaN	...	78.0	29.0	51.0	825.0	1.3	40.0	49	0
2	141920	1.03	-113	2469	2582	0	0	15	cardiovascular shock / hypotension sepsis	NaN	...	-1.0	-1.0	34.0	136.0	0.6	-1.0	35	0
3	141945	2.17	-205	6539	6744	0	1	25	cardiovascular shock / hypotension sepsis	NaN	...	-1.0	-1.0	23.0	79.0	-1.0	-1.0	20	0
4	142388	1.11	-94	7122	7216	0	1	73	cardiovascular shock / hypotension sepsis	NaN	...	-1.0	-1.0	21.0	398.0	-1.0	-1.0	54	0

5 rows × 122 columns

In [124...

df14['ACEI/ARB'] = (df14['ACEI'] == 1) | (df14['ARB'] == 1)
df14['ACEI/ARB'] = df14['ACEI/ARB'].astype(int)
删除原本的 'ACEI'和 'arb'列
df14.drop(['ACEI', 'ARB'], axis=1, inplace=True)

In [125...

df14.shape

Out[125...

(5305, 121)

In [126...

df14['ACEI/ARB'].value_counts()

Out[126...

ACEI/ARB
0 5236
1 69
Name: count, dtype: int64

In [127...

print(list(df14.columns))

```
['patientunitstayid', 'Scr_baseline', 'akimintime', 'akimaxtime', 'los_aki', 'AKD', 'cxx_aki', 'diagnosisoffset', 'diagnosisstring', 'uniquepid', 'patienthealthsystemstayid', 'unitvisitnumbe
r', 'hospitalid', 'region', 'unittype', 'hospitaladmitoffset', 'hospitaldischargeoffset', 'unitadmitoffset', 'unitdischargeoffset', 'apache_iv', 'hospitaldischargeyear', 'age', 'hosp_mort', 'g
ender', 'ethnicity', 'admissionheight', 'admissionweight', 'dischargeweight', 'icu_los_hours', 'aniongap_min', 'aniongap_max', 'albumin_min', 'albumin_max', 'bands_min', 'bands_max', 'bicarbon
ate_min', 'bicarbonate_max', 'hco3_min', 'hco3_max', 'bilirubin_min', 'bilirubin_max', 'creatinine_min', 'creatinine_max', 'chloride_min', 'chloride_max', 'glucose_min', 'glucose_max', 'hemato
crit_min', 'hematocrit_max', 'hemoglobin_min', 'hemoglobin_max', 'lactate_min', 'lactate_max', 'platelet_min', 'platelet_max', 'potassium_min', 'potassium_max', 'ptt_min', 'ptt_max', 'inr_mi
n', 'inr_max', 'pt_min', 'pt_max', 'sodium_min', 'sodium_max', 'bun_min', 'bun_max', 'wbc_min', 'wbc_max', 'window_start7', 'CRRT', 'apachepatientresultsid', 'physicianspeciality', 'physiciani
nterventioncategory', 'acutephysiologyscore', 'apachescore', 'apacheversion', 'predictedicumortality', 'actualicumortality', 'predictediculos', 'actualiculos', 'predictedhospitalmortality', 'a
ctualhospitalmortality', 'predictedhospitallos', 'actualhospitallos', 'preopmi', 'preopcardiaccath', 'ptcawithin24h', 'unabridgedunitlos', 'unabridgedhosplos', 'actualventdays', 'predventday
s', 'unabridgedactualventdays', 'apacheapsvarid', 'intubated', 'vent', 'dialysis', 'eyes', 'motor', 'verbal', 'meds', 'urine', 'wbc', 'temperature', 'respiratoryrate', 'sodium', 'heartrate',
'meanbp', 'ph', 'hematocrit', 'creatinine', 'albumin', 'pao2', 'pco2', 'bun', 'glucose', 'bilirubin', 'fio2', 'oasis', 'vasopressor', 'ACEI/ARB']
```

```
In [128... df14.replace(-1, np.nan, inplace=True)
```

```
In [129... df14[['eyes', 'motor', 'verbal']].head()
```

	eyes	motor	verbal
0	4.0	6.0	5.0
1	4.0	6.0	4.0
2	4.0	6.0	5.0
3	4.0	6.0	5.0
4	1.0	1.0	1.0

```
In [130... df14['GCS']=df14['eyes'] + df14['motor'] + df14['verbal']
```

```
In [131... print(list(df14.columns))

['patientunitstayid', 'Scr_baseline', 'akimintime', 'akimaxtime', 'los_aki', 'AKD', 'cxx_aki', 'diagnosisoffset', 'diagnosisstring', 'uniquepid', 'patienthealthsystemstayid', 'unitvisitnumbe
r', 'hospitalid', 'region', 'unittype', 'hospitaladmitoffset', 'hospitaldischargeoffset', 'unitadmitoffset', 'unitdischargeoffset', 'apache_iv', 'hospitaldischargeyear', 'age', 'hosp_mort', 'g
ender', 'ethnicity', 'admissionheight', 'admissionweight', 'dischargeweight', 'icu_los_hours', 'aniongap_min', 'aniongap_max', 'albumin_min', 'albumin_max', 'bands_min', 'bands_max', 'bicarbon
ate_min', 'bicarbonate_max', 'hco3_min', 'hco3_max', 'bilirubin_min', 'bilirubin_max', 'creatinine_min', 'creatinine_max', 'chloride_min', 'chloride_max', 'glucose_min', 'glucose_max', 'hemato
crit_min', 'hematocrit_max', 'hemoglobin_min', 'hemoglobin_max', 'lactate_min', 'lactate_max', 'platelet_min', 'platelet_max', 'potassium_min', 'potassium_max', 'ptt_min', 'ptt_max', 'inr_mi
n', 'inr_max', 'pt_min', 'pt_max', 'sodium_min', 'sodium_max', 'bun_min', 'bun_max', 'wbc_min', 'wbc_max', 'window_start7', 'CRRT', 'apachepatientresultsid', 'physicianspeciality', 'physiciani
nterventioncategory', 'acutephysiologyscore', 'apachescore', 'apacheversion', 'predictedicumortality', 'actualicumortality', 'predictediculos', 'actualiculos', 'predictedhospitalmortality', 'a
ctualhospitalmortality', 'predictedhospitallos', 'actualhospitallos', 'preopmi', 'preopcardiaccath', 'ptcawithin24h', 'unabridgedunitlos', 'unabridgedhosplos', 'actualventdays', 'predventday
s', 'unabridgedactualventdays', 'apacheapsvarid', 'intubated', 'vent', 'dialysis', 'eyes', 'motor', 'verbal', 'meds', 'urine', 'wbc', 'temperature', 'respiratoryrate', 'sodium', 'heartrate',
'meanbp', 'ph', 'hematocrit', 'creatinine', 'albumin', 'pao2', 'pco2', 'bun', 'glucose', 'bilirubin', 'fio2', 'oasis', 'vasopressor', 'ACEI/ARB', 'GCS']
```

```
In [ ]: vital= pd.read_csv(r'E:\MIMIC\eICU\eICU\pivoted_vital.csv')
vital.head()
```

```
In [133... vital1=vital[['patientunitstayid','chartoffset','heartrate','respiratoryrate','spo2','nibp_systolic','nibp_diastolic','temperature']]
```

```
In [134... vital1.head()
```

	patientunitstayid	chartoffset	heartrate	respiratoryrate	spo2	nibp_systolic	nibp_diastolic	temperature
0	141168	6	140.0	NaN	NaN	82.0	59.0	NaN
1	141168	21	70.0	NaN	94.0	NaN	NaN	NaN
2	141168	36	70.0	NaN	94.0	105.0	59.0	NaN
3	141168	51	140.0	NaN	94.0	103.0	65.0	NaN
4	141168	66	140.0	NaN	92.0	99.0	68.0	NaN

```
In [135... unique_patientunitstayid_df14 = df14['patientunitstayid'].unique()
vital2 = vital1[vital1['patientunitstayid'].isin(unique_patientunitstayid_df14)]
vital2.shape
```

Out[135... (1571741, 8)

```
In [136... vital2.patientunitstayid.nunique()
```

Out[136... 5174

```
In [137... vital3=vital2.query('chartoffset >= 0')
vital3.shape
```

Out[137... (1518781, 8)

```
In [138... vital3.patientunitstayid.nunique()
```

Out[138... 5173

missing_ratio = vital3.isnull().mean(axis=1) vital4= vital3[missing_ratio < 0.5] vital4.shape

vital4.patientunitstayid.nunique()

```
In [139... min_chartoffsets = vital3.groupby('patientunitstayid')['chartoffset'].min().reset_index()

# 步骤2和3: 检查缺失值并替换
def fill_missing_values(row, df):
    for col in ['heartrate', 'respiratoryrate', 'spo2', 'nibp_systolic', 'nibp_diastolic', 'temperature']:
        if pd.isna(row[col]):
            # 获取该patientunitstayid的下一个chartoffset的行
            next_rows = df[(df['patientunitstayid'] == row['patientunitstayid']) & (df['chartoffset'] > row['chartoffset'])]
            for _, next_row in next_rows.iterrows():
                if not pd.isna(next_row[col]):
                    row[col] = next_row[col]
                    break
    return row

# 应用函数
result_list = []

for _, row in min_chartoffsets.iterrows():
    patient_data = vital3[vital3['patientunitstayid'] == row['patientunitstayid']]
    min_chartoffset_row = patient_data[patient_data['chartoffset'] == row['chartoffset']].iloc[0]
    filled_row = fill_missing_values(min_chartoffset_row, patient_data)
    result_list.append(filled_row)

# 使用concat合并DataFrame
result = pd.concat(result_list, axis=1).transpose()

result[['patientunitstayid', 'chartoffset', 'heartrate', 'respiratoryrate', 'spo2', 'nibp_systolic', 'nibp_diastolic', 'temperature']]
```

Out[139...

	patientunitstayid	chartoffset	heartrate	respiratoryrate	spo2	nibp_systolic	nibp_diastolic	temperature	
	2452	141304.0	14.0	118.0	28.0	95.0	106.0	55.0	38.5
	9863	141751.0	14.0	95.0	25.0	94.0	77.0	48.0	31.2
	14702	141920.0	6.0	112.0	28.0	97.0	103.0	49.0	38.8
	15232	141945.0	12.0	78.0	14.0	69.0	97.0	53.0	36.7
	21301	142388.0	7.0	122.0	22.0	97.0	211.0	111.0	37.4
	...	...	...	...	...	...	...	...	...
	21019457	3352634.0	9.0	149.0	26.0	96.0	70.0	46.0	38.6
	21020490	3352721.0	8.0	68.0	29.0	100.0	79.0	52.0	33.5
	21023337	3352827.0	0.0	67.0	16.0	100.0	164.0	75.0	36.7
	21035224	3353197.0	1.0	107.0	16.0	100.0	100.0	50.0	37.1
	21037634	3353251.0	6.0	100.0	24.0	100.0	134.0	70.0	33.4

5173 rows × 8 columns

In [140...

```
result.head()
```

Out[140...

	patientunitstayid	chartoffset	heartrate	respiratoryrate	spo2	nibp_systolic	nibp_diastolic	temperature	
	2452	141304.0	14.0	118.0	28.0	95.0	106.0	55.0	38.5
	9863	141751.0	14.0	95.0	25.0	94.0	77.0	48.0	31.2
	14702	141920.0	6.0	112.0	28.0	97.0	103.0	49.0	38.8
	15232	141945.0	12.0	78.0	14.0	69.0	97.0	53.0	36.7
	21301	142388.0	7.0	122.0	22.0	97.0	211.0	111.0	37.4

In [141...

```
print(np.intersect1d(df14.columns,result.columns))
```

['heartrate' 'patientunitstayid' 'respiratoryrate' 'temperature']

In [142...

```
missing_rate = result['temperature'].isna().mean()  
missing_rate
```

Out[142...

0.009278948385849604

In [143...

```
missing_rate = df14['temperature'].isna().mean()  
missing_rate
```

Out[143...

0.04957587181903864

In [144...

```
df14.drop(['heartrate','respiratoryrate','temperature'], axis=1, inplace=True)
```

In [145...

```
result['patientunitstayid'] = result['patientunitstayid'].astype(int)
```

In [146...

```
df15 = df14.merge(result, on=['patientunitstayid'], how='left')  
df15.head()
```

Out[146...

	patientunitstayid	Scr_baseline	akimintime	akimaxtime	los_aki	AKD	cxx_aki	diagnosisoffset	diagnosisstring	uniquepid	...	vasopressor	ACEI/ARB	GCS	chartoffset	heartrate	respirator
0	141304	1.24	-127.0	7293	7420	0	1	23.0	cardiovascular shock / hypotension sepsis	NaN	...	0	0	15.0	14.0	118.0	
1	141751	0.90	-101.0	6042	6143	0	1	45.0	cardiovascular shock / hypotension signs and s...	NaN	...	0	0	14.0	14.0	95.0	
2	141920	1.03	-113.0	2469	2582	0	0	15.0	cardiovascular shock / hypotension sepsis	NaN	...	0	0	15.0	6.0	112.0	
3	141945	2.17	-205.0	6539	6744	0	1	25.0	cardiovascular shock / hypotension sepsis	NaN	...	0	0	15.0	12.0	78.0	
4	142388	1.11	-94.0	7122	7216	0	1	73.0	cardiovascular shock / hypotension sepsis	NaN	...	0	0	3.0	7.0	122.0	

5 rows × 126 columns



In [147...

```
df15.shape
```

Out[147...

(5305, 126)

In [148...

```
print(list(df15.columns))
```

['patientunitstayid', 'Scr_baseline', 'akimintime', 'akimaxtime', 'los_aki', 'AKD', 'cxx_aki', 'diagnosisoffset', 'diagnosisstring', 'uniquepid', 'patienthealthsystemstayid', 'unitvisitnumbe
r', 'hospitalid', 'region', 'unitttype', 'hospitaladmitoffset', 'hospitaldischargeoffset', 'unitadmitoffset', 'unitdischargeoffset', 'apache_iv', 'hospitaldischargeyear', 'age', 'hosp_mort', 'g
ender', 'ethnicity', 'admissionheight', 'admissionweight', 'dischargeweight', 'icu_los_hours', 'aniongap_min', 'aniongap_max', 'albumin_min', 'albumin_max', 'bands_min', 'bands_max', 'bicarbon
ate_min', 'bicarbonate_max', 'hco3_min', 'hco3_max', 'bilirubin_min', 'bilirubin_max', 'creatinine_min', 'creatinine_max', 'chloride_min', 'chloride_max', 'glucose_min', 'glucose_max', 'hemato
crit_min', 'hematocrit_max', 'hemoglobin_min', 'hemoglobin_max', 'lactate_min', 'lactate_max', 'platelet_min', 'platelet_max', 'potassium_min', 'potassium_max', 'ptt_min', 'ptt_max', 'inr_mi
n', 'inr_max', 'pt_min', 'pt_max', 'sodium_min', 'sodium_max', 'bun_min', 'bun_max', 'wbc_min', 'wbc_max', 'window_start7', 'CRRT', 'apachepatientresultsid', 'physicianspeciality', 'physiciani
nterventioncategory', 'acutephysiologyscore', 'apachescore', 'apacheversion', 'predictedicumorality', 'actualicumortality', 'predictediculos', 'actualiculos', 'predictedhospitalmortality', 'a
ctualhospitalmortality', 'predictedhospitallos', 'actualhospitallos', 'preopmi', 'preopcardiaccath', 'ptcawithin24h', 'unabridgedunitlos', 'unabridgedhosplos', 'actualventdays', 'predventday
s', 'unabridgedactualventdays', 'apacheapsvarid', 'intubated', 'vent', 'dialysis', 'eyes', 'motor', 'verbal', 'meds', 'urine', 'wbc', 'sodium', 'meanbp', 'ph', 'hematocrit', 'creatinine', 'alb
umin', 'pao2', 'pco2', 'bun', 'glucose', 'bilirubin', 'fio2', 'oasis', 'vasopressor', 'ACEI/ARB', 'GCS', 'chartoffset', 'heartrate', 'respiratoryrate', 'spo2', 'nibp_systolic', 'nibp_diastoli
c', 'temperature']

In [149...

```
import matplotlib.pyplot as plt  
from missingno import missingno as msno
```

In [150...

```
# 计算每列的缺失率  
missing_rate = df15.isnull().mean()  
  
# 找出缺失率大于30%的列  
cols_to_drop = missing_rate[missing_rate > 0.30].index  
  
cols_to_drop  
# 显示结果
```

Out[150...

Index(['uniquepid', 'region', 'unitttype', 'ethnicity', 'dischargeweight',
 'bands_min', 'bands_max', 'hco3_min', 'hco3_max', 'ptt_min', 'ptt_max',
 'actualventdays', 'predventdays', 'unabridgedactualventdays', 'urine',
 'ph', 'albumin', 'pao2', 'pco2', 'bilirubin', 'fio2'],
 dtype='object')

In [151...

```
df15.drop(cols_to_drop, axis=1, inplace=True)
```

In []:

```
msno.bar(df15)
```

In [153...

```
pt= lab[lab['labname'].str.fullmatch('PT', na=False)]  
pt.head()
```

Out[153...

	labid	patientunitstayid	labresultoffset	labtypeid	labname	labresult	labresulttext	labmeasurenamesystem	labmeasurenameinterface	labresultrevisedoffset	
	3	50363250	141168	1133	3	PT	26.6	26.6	sec	sec	1208
	11	56072055	141168	231	3	PT	17.1	17.1	sec	sec	268
	41	50567433	141168	2026	3	PT	43.3	43.3	sec	sec	2219
	83	56391148	141168	516	3	PT	20.0	20.0	sec	sec	534
	94	56035789	141168	2026	3	PT	46.1	46.1	sec	sec	2069

In [154...

```
pt = pt.dropna(subset=['labresult'])
```

In [155...

```
pt1=pt[['patientunitstayid','labresultoffset','labresult']]
pt1.head()
```

Out[155...

	patientunitstayid	labresultoffset	labresult	
	3	141168	1133	26.6
	11	141168	231	17.1
	41	141168	2026	43.3
	83	141168	516	20.0
	94	141168	2026	46.1

In [] :

```
pt1.rename(columns={'labresult': 'PT'}, inplace=True)
pt1.head()
```

In [157...

```
pt2= pt1.groupby(['patientunitstayid'])['labresultoffset'].idxmin()
pt3= pt1.loc[pt2]
pt3.head()
```

Out[157...

	patientunitstayid	labresultoffset	PT
	11	141168	231 17.1
	242	141194	-923 12.1
	657	141203	-1580 80.6
	866	141227	-13 13.3
	998	141229	-106 34.5

In [158...

```
pt3 = pt3.drop('labresultoffset', axis=1)
```

In [159...

```
df16 = df15.merge(pt3, on=['patientunitstayid'], how='left')
df16.shape
```

Out[159...

```
(5305, 106)
```

In [160...

```
missing_rate = df16['PT'].isna().mean()
missing_rate
```

Out[160...

```
0.15626767200754005
```

In [] :

```
pa02= lab[lab['labname'].str.fullmatch('pa02', na=False)]
pa02= pa02.dropna(subset=['labresult'])
pa021=pa02[['patientunitstayid','labresultoffset','labresult']]
pa021.rename(columns={'labresult': 'pa02'}, inplace=True)
pa021.head()
```

In [162...

```
pa022= pa021.groupby(['patientunitstayid'])['labresultoffset'].idxmin()
pa023= pa021.loc[pa022]
pa023.head()
```

Out[162...

	patientunitstayid	labresultoffset	pa02	
	89	141168	1805	41.0
	334	141194	2102	100.0
	560	141196	1030	70.0
	712	141203	1	51.0
	836	141227	-13	64.0

In [163...

```
pa023= pa023.drop('labresultoffset', axis=1)
```

In [] :

In [164...

```
df17 = df16.merge(pa023, on=['patientunitstayid'], how='left')
df17.shape
```

Out[164...

```
(5305, 107)
```

In [165...

```
missing_rate = df17['pa02'].isna().mean()
missing_rate
```

Out[165...

```
0.2524033930254477
```

In [] :

```
Fi02= lab[lab['labname'].str.fullmatch('Fi02', na=False)]
Fi02= Fi02.dropna(subset=['labresult'])
Fi021=Fi02[['patientunitstayid','labresultoffset','labresult']]
Fi021.rename(columns={'labresult': 'Fi02'}, inplace=True)
Fi022= Fi021.groupby(['patientunitstayid'])['labresultoffset'].idxmin()
Fi023= Fi021.loc[Fi022]
Fi023.head()
```

In [167...

```
Fi023= Fi023.drop('labresultoffset', axis=1)
```

In [168...

```
df18 = df17.merge(Fi023, on=['patientunitstayid'], how='left')
df18.shape
```

Out[168...

```
(5305, 108)
```

In [169...

```
missing_rate = df18['Fi02'].isna().mean()
missing_rate
```

Out[169...

```
0.30763430725730445
```

In [170...

```
df18['Pa02/Fi02'] = df18.apply(lambda row: row['pa02'] / row['Fi02'] if row['Fi02'] != 0 else None, axis=1)
```

In [171...

```
df18['Pa02/Fi02'] = df18['Pa02/Fi02'] * 100
```

In [172...

```
df18['bun'] = df18['bun'] * 0.375
df18['bun'] = df18['bun'].round(2)
```

```
In [173... # 定义计算LODS评分的函数
def calculate_lods(row):
    # GCS评分
    if row['GCS'] <= 5:
        gcs_score = 5
    elif 6 <= row['GCS'] <= 8:
        gcs_score = 3
    elif 9 <= row['GCS'] <= 13:
        gcs_score = 1
    else:
        gcs_score = 0

    # 收缩压评分
    if row['nibp_systolic'] < 40:
        sbp_score = 5
    elif 40 <= row['nibp_systolic'] <= 69:
        sbp_score = 3
    elif 70 <= row['nibp_systolic'] <= 89:
        sbp_score = 1
    elif 90 <= row['nibp_systolic'] <= 239:
        sbp_score = 0
    elif 240 <= row['nibp_systolic'] <= 269:
        sbp_score = 1
    else:
        sbp_score = 3

    # 尿素氮评分
    if row['bun'] < 6:
        bun_score = 0
    elif 6 <= row['bun'] <= 9.9:
        bun_score = 1
    elif 10 <= row['bun'] <= 19.9:
        bun_score = 3
    else:
        bun_score = 5

    # 机械通气及PaO2/FiO2评分

    if row['vent'] == 0:
        vent_score = 0
    elif row['vent'] == 1:
        if pd.isna(row['PaO2/FiO2']):
            vent_score = 1
        elif row['PaO2/FiO2'] < 150:
            vent_score = 3
        else:
            vent_score = 1
    else:
        vent_score = 0

    # 白细胞计数评分
    if row['wbc'] < 1:
        wbc_score = 3
    elif 1 <= row['wbc'] <= 2.4:
        wbc_score = 1
    elif 2.5 <= row['wbc'] <= 49.9:
        wbc_score = 0
    else:
        wbc_score = 1

    # PT延长评分
    if row['PT'] > 16:
        pt_score = 3
    else:
        pt_score = 0

    # 总LODS评分
    total_lods = gcs_score + sbp_score + bun_score + vent_score + wbc_score + pt_score
    return total_lods

# 应用函数计算每个病人的LODS值
df18['LODS'] = df18.apply(calculate_lods, axis=1)
```

```
In [174... df18['LODS'].value_counts()
```

```
Out[174... LODS
6      868
9      591
7      583
4      505
5      486
3      481
8      472
10     304
11     222
1      196
12     196
2      146
13      86
14      61
0       54
15      34
16       11
17        9
Name: count, dtype: int64
```

```
In [175... df18.shape
```

```
Out[175... (5305, 110)
```

```
In [ ]: BaseExcess= lab[lab['labname'].str.fullmatch('Base Excess', na=False)]
BaseExcess= BaseExcess.dropna(subset=['labresult'])
BaseExcess1=BaseExcess[['patientunitstayid','labresultoffset','labresult']]
BaseExcess1.rename(columns={'labresult': 'BaseExcess'}, inplace=True)
BaseExcess2= BaseExcess1.groupby(['patientunitstayid'])['labresultoffset'].idxmin()
BaseExcess3= BaseExcess1.loc[BaseExcess2]
BaseExcess3.head()
```

```
In [177... BaseExcess3= BaseExcess3.drop('labresultoffset', axis=1)
df19= df18.merge(BaseExcess3, on=['patientunitstayid'], how='left')
df19.shape
```

```
Out[177... (5305, 111)
```

```
In [178... missing_rate = df19['BaseExcess'].isna().mean()
missing_rate
```

```
Out[178... 0.3768143261074458
```

```
In [179... print(list(df19.columns))
```


In [195...

print(list(charlson.columns))

['patientunitstayid', 'mets6', 'aids6', 'liver3', 'stroke2', 'renal2', 'dm', 'cancer2', 'leukemia2', 'lymphoma2', 'mi1', 'chf1', 'pvd1', 'tia1', 'dementia1', 'copd1', 'ctd1', 'pud1', 'liver1', 'age_score_charlson', 'final_charlson_score']

In [196...

df21= df20.merge(charlson, on=['patientunitstayid'], how='left')
df21.shape

Out[196...

(5305, 133)

In [197...

df21['Cerebrovascular_disease'] = df21.apply(lambda row: 0 if row['stroke2'] == 0 and row['tia1'] == 0 else 1, axis=1)

In [198...

df21['Cerebrovascular_disease'].value_counts()

Out[198...

Cerebrovascular_disease
0 4613
1 692
Name: count, dtype: int64

In [199...

paCO2= lab[lab['labname'].str.fullmatch('paCO2', na=False)]
paCO2= paCO2.dropna(subset=['labresult'])

In [200...

paCO2.head()

Out[200...

	labid	patientunitstayid	labresultoffset	labtypeid	labname	labresult	labresulttext	labmeasurenamesystem	labmeasurenameinterface	labresultrevisedoffset
39	67446289	141168	2010	7	paCO2	44.0	44	mm Hg	mm Hg	2026
40	66695373	141168	2141	7	paCO2	37.0	37	mm Hg	mm Hg	2155
42	67301509	141168	2026	7	paCO2	31.0	31	mm Hg	mm Hg	2054
77	67217623	141168	1805	7	paCO2	46.0	46	mm Hg	mm Hg	1811
362	66794223	141194	2102	7	paCO2	26.0	26	mm Hg	mm Hg	2114

In [] :

paCO21=paCO2[['patientunitstayid','labresultoffset','labresult']]
paCO21.rename(columns={'labresult': 'paCO2'}, inplace=True)
paCO21.head()

In [202...

paCO22= paCO21.groupby(['patientunitstayid'])['labresultoffset'].idxmin()
paCO23= paCO21.loc[paCO22]
paCO23.head()

Out[202...

	patientunitstayid	labresultoffset	paCO2
77	141168	1805	46.0
362	141194	2102	26.0
579	141196	1030	45.0
700	141203	1	37.0
828	141227	-13	20.0

In [203...

paCO23= paCO23.drop('labresultoffset', axis=1)

In [204...

df22 = df21.merge(paCO23, on=['patientunitstayid'], how='left')
df22.shape

Out[204...

(5305, 135)

In [205...

df22.head()

Out[205...

	patientunitstayid	Scr_baseline	akimintime	akimaxtime	los_aki	AKD	cxx_aki	diagnosisoffset	diagnosisstring	patienthealthsystemstayid	...	tia1	dementia1	copd1	ctd1	pud1	liver1	ag
0	141304	1.24	-127.0	7293	7420	0	1	23.0	cardiovascular shock / hypotension sepsis	129031	...	0	0	0	0	0	0	
1	141751	0.90	-101.0	6042	6143	0	1	45.0	cardiovascular shock / hypotension signs and s...	129381	...	0	0	0	0	0	0	
2	141920	1.03	-113.0	2469	2582	0	0	15.0	cardiovascular shock / hypotension sepsis	129506	...	0	0	0	0	0	0	
3	141945	2.17	-205.0	6539	6744	0	1	25.0	cardiovascular shock / hypotension sepsis	129526	...	0	0	0	0	0	0	
4	142388	1.11	-94.0	7122	7216	0	1	73.0	cardiovascular shock / hypotension sepsis	129864	...	0	0	0	0	0	0	

5 rows × 135 columns

In [206...

print(list(df22.columns))

['patientunitstayid', 'Scr_baseline', 'akimintime', 'akimaxtime', 'los_aki', 'AKD', 'cxx_aki', 'diagnosisoffset', 'diagnosisstring', 'patienthealthsystemstayid', 'unitvisitnumber', 'hospitalid', 'hospitaladmitoffset', 'hospitaldischargeoffset', 'unitadmitoffset', 'unitdischargeoffset', 'apache_iv', 'hospitaldischargeyear', 'age', 'hosp_mort', 'gender', 'admissionheight', 'admissionweight', 'icu_los_hours', 'aniongap_min', 'aniongap_max', 'albumin_min', 'albumin_max', 'bicarbonate_min', 'bicarbonate_max', 'bilirubin_min', 'bilirubin_max', 'creatinine_min', 'creatinine_max', 'chloride_min', 'chloride_max', 'glucose_min', 'glucose_max', 'hematocrit_min', 'hematocrit_max', 'hemoglobin_min', 'hemoglobin_max', 'lactate_min', 'lactate_max', 'platelet_min', 'platelet_max', 'potassium_min', 'potassium_max', 'inr_min', 'inr_max', 'pt_min', 'pt_max', 'sodium_min', 'sodium_max', 'bun_min', 'bun_max', 'wbc_min', 'wbc_max', 'window_start7', 'CRRT', 'apachepatientresultsid', 'physicianspeciality', 'physicianinterventioncategory', 'acutephysiologyscore', 'apachescore', 'apacheversion', 'predictedicumortality', 'actualicumortality', 'predictediculos', 'actualiculos', 'predictedhospitalmortality', 'actualhospitalmortality', 'predictedhospitallos', 'actualhospitallos', 'preopmi', 'preopcardiaccath', 'ptcawithin24h', 'unabridgedunitlos', 'unabridgedhospos', 'apacheapsvarid', 'intubated', 'vent', 'dialysis', 'eyes', 'motor', 'verbal', 'meds', 'wbc', 'sodium', 'meanbp', 'hematocrit', 'creatinine', 'bun', 'glucose', 'oasis', 'vasopressor', 'ACEI/ARB', 'GCS', 'chartoffset', 'heartrate', 'respiratoryrate', 'spo2', 'nibp_systolic', 'nibp_diastolic', 'temperature', 'PT', 'paO2', 'FiO2', 'PaO2/FiO2', 'LODS', 'BaseExcess', 'drugstartoffset', 'Los_inf_AB', 'mets6', 'aids6', 'liver3', 'stroke2', 'renal2', 'dm', 'cancer2', 'leukemia2', 'lymphoma2', 'mi1', 'chf1', 'pvd1', 'tia1', 'dementia1', 'copd1', 'ctd1', 'pud1', 'liver1', 'age_score_charlson', 'final_charlson_score', 'Cerebrovascular_disease', 'paCO2']

In [207...

paraplegia= diagnosis[diagnosis['diagnosisstring'].str.contains('paraplegia')]
paraplegia.head()

Out[207...

	diagnosisid	patientunitstayid	activeupondischarge	diagnosisoffset	diagnosisstring	icd9code	diagnosispriority
103	4250611	141314	False	91	neurologic disorders of the spinal cord and pe...	344.1, G82.21	Other
106	4071439	141314	False	63	neurologic disorders of the spinal cord and pe...	344.1, G82.21	Other
152	4276075	141314	True	125	neurologic disorders of the spinal cord and pe...	344.1, G82.21	Other
158	3887340	141314	False	125	neurologic disorders of the spinal cord and pe...	344.1, G82.21	Other
163	3636936	141314	False	28	neurologic disorders of the spinal cord and pe...	344.1, G82.21	Other

In [208...

paraplegia1=paraplegia[['patientunitstayid','diagnosisoffset','diagnosisstring']]
paraplegia1.head()

Out[208...

	patientunitstayid	diagnosisoffset	diagnosisstring
103	141314	91	neurologic disorders of the spinal cord and pe...
106	141314	63	neurologic disorders of the spinal cord and pe...
152	141314	125	neurologic disorders of the spinal cord and pe...
158	141314	125	neurologic disorders of the spinal cord and pe...
163	141314	28	neurologic disorders of the spinal cord and pe...

In [209...

paraplegia1.shape

Out[209...

(1071, 3)

In [210...

paraplegia1.patientunitstayid.nunique()

Out[210...

220

In [211...

paraplegia2 = paraplegia1.groupby(['patientunitstayid'])['diagnosisoffset'].idxmin()
paraplegia3 = paraplegia1.loc[paraplegia2]

In [212...

paraplegia3.head()

Out[212...

	patientunitstayid	diagnosisoffset	diagnosisstring
163	141314	28	neurologic disorders of the spinal cord and pe...
10578	156903	21	neurologic disorders of the spinal cord and pe...
10618	156988	30	neurologic disorders of the spinal cord and pe...
20484	169229	78	neurologic disorders of the spinal cord and pe...
21603	171138	27	neurologic disorders of the spinal cord and pe...

In [213...

paraplegia3['paraplegia']=1

In []]:

df23 = df22.merge(paraplegia3[['patientunitstayid', 'paraplegia']], on='patientunitstayid', how='left')
df23['paraplegia'].fillna(0, inplace=True)

In [215...

df23['paraplegia'] = df23['paraplegia'].astype(int)

In []]:

df23

df23.to_csv('250510 EICU数据 saki.csv')

In []]:

In [218...

final=df23[['patientunitstayid', 'Scr_baseline', 'AKD', 'cxx_aki', 'apache_iv', 'age', 'CRRT', 'LODS', 'BaseExcess', 'Los_inf_AB', 'vent',
'temperature', 'respiratoryrate', 'ACEI/ARB', 'vasopressor', 'bun', 'glucose', 'oasis', 'wbc', 'sodium', 'acutephysiologyscore',
'Cerebrovascular_disease', 'admissionweight', 'paO2', 'paCO2', 'spo2']]

In []]:

In [219...

final1=final[['Scr_baseline', 'AKD', 'age', 'CRRT', 'LODS', 'BaseExcess', 'Los_inf_AB',
'temperature', 'respiratoryrate', 'ACEI/ARB', 'vasopressor', 'bun', 'glucose', 'oasis', 'wbc', 'sodium', 'acutephysiologyscore',
'Cerebrovascular_disease', 'admissionweight', 'paO2', 'paCO2']]

In [220...

rename_mapping = {
 'Scr_baseline': 'Scr_baseline',
 'age': 'Age',
 'CRRT': 'CRRT',
 'LODS': 'LODS',
 'BaseExcess': 'Baseexcess',
 'Los_inf_AB': 'Los_inf_AB',
 'temperature': 'Temperature',
 'respiratoryrate': 'Resp_rate',
 'ACEI/ARB': 'ACEI/ARB',
 'vasopressor': 'Vasoactive_agent',
 'bun': 'Bun',
 'glucose': 'Glucose',
 'oasis': 'OASIS',
 'wbc': 'WBC',
 'sodium': 'Sodium',
 'acutephysiologyscore': 'APS_III',
 'Cerebrovascular_disease': 'Cerebrovascular_disease',
 'admissionweight': 'Weight',
 'paO2': 'Po2',
 'paCO2': 'Pco2'
}

使用rename方法替换变量名
final2= final1.rename(columns=rename_mapping)

In [221...

final2.shape

Out[221...

(5305, 21)

In [222...

final2.head()

Out[222...

	Scr_baseline	AKD	Age	CRRT	LODS	Baseexcess	Los_inf_AB	Temperature	Resp_rate	ACEI/ARB	...	Bun	Glucose	OASIS	WBC	Sodium	APS_III	Cerebrovascular_disease	Weight	Po2	Pco2
0	1.24	0	70.0	0	4	6.0	0.00	38.5	28.0	0	...	18.00	164.0	28	12.8	140.0	41.0	0	NaN	29.0	64.0
1	0.90	0	60.0	0	8	NaN	0.19	31.2	25.0	0	...	19.12	825.0	49	20.0	127.0	NaN	0	NaN	140.0	26.0
2	1.03	0	81.0	0	4	NaN	0.00	38.8	28.0	0	...	12.75	136.0	35	22.4	138.0	41.0	0	53.4	NaN	NaN
3	2.17	0	72.0	0	1	NaN	0.00	36.7	14.0	0	...	8.62	79.0	20	11.1	139.0	40.0	0	74.8	NaN	NaN
4	1.11	0	67.0	0	7	NaN	0.86	37.4	22.0	0	...	7.88	398.0	54	18.9	139.0	101.0	0	96.6	NaN	NaN

5 rows × 21 columns

◀

▶

In []]:

msno.bar(final2)

In [224...

删除每行缺失值超过30%的行
threshold = 0.9 * len(final2.columns) # 保留至少70%的非缺失值
final2_cleaned = final2.dropna(thresh=threshold)

final2_cleaned.shape

Out[224...

(3669, 21)

In [225...

final2_cleaned['AKD'].value_counts()

```
Out[225...      AKD
      1      2006
      0      1663
      Name: count, dtype: int64
```

```
In [ ]]: msno.bar(final2_cleaned)
```

```
In [227... threshold = 1 * len(final2.columns) # 保留至少70%的非缺失值
final3 = final2.dropna(thresh=threshold)

final3.shape
```

```
Out[227... (2282, 21)
```

```
In [228... final3['AKD'].value_counts()
```

```
Out[228...      AKD
      1      1307
      0       975
      Name: count, dtype: int64
```

```
In [ ]]: msno.bar(final3)
```

```
In [ ]]: final3['Glucose'] = final3['Glucose'] / 18.0182
```

```
In [ ]]: final3['Scr_baseline'] = final3['Scr_baseline'] * 88.4
```

```
In [232... column_order = [
    'ACEI/ARB', 'APS_III', 'Age', 'Baseexcess', 'Bun', 'CRRT',
    'Cerebrovascular_disease', 'Glucose', 'LODS', 'Los_inf_AB', 'OASIS', 'Pco2',
    'Po2', 'Resp_rate', 'Scr_baseline', 'Sodium', 'Temperature',
    'Vasoactive_agent', 'WBC', 'Weight', 'AKD'
]

# 按指定顺序排列列
final3 = final3[column_order]
```

```
In [233... final3.head()
```

Out[233...

	ACEI/ARB	APS_III	Age	Baseexcess	Bun	CRRT	Cerebrovascular_disease	Glucose	LODS	Los_inf_AB	...	Pco2	Po2	Resp_rate	Scr_baseline	Sodium	Temperature	Vasoactive_agent	WBC
6	0	63.0	60.0	0.0	10.50	0	0	21.311785	7	0.28	...	54.0	233.0	20.0	162.656	131.0	37.1	0	16.5
7	0	74.0	65.0	3.0	21.75	0	0	4.051459	9	0.00	...	46.0	74.0	22.0	238.680	140.0	39.5	0	16.0
9	0	48.0	72.0	5.0	10.50	0	0	22.366274	6	0.00	...	50.0	84.0	27.0	120.224	142.0	37.4	0	17.7
14	0	127.0	67.0	1.0	18.38	0	0	11.266386	11	0.16	...	31.0	86.0	31.0	71.604	136.0	36.8	0	13.3
22	0	40.0	71.0	1.0	11.62	0	0	5.050449	4	2.58	...	53.0	229.0	12.0	175.916	138.0	38.3	0	15.7

5 rows × 21 columns



```
In [234... y_val= final3['AKD']
X_val= final3.drop('AKD',axis=1)
```

```
In [235... import streamlit as st
import joblib
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

```
In [ ]]: model = joblib.load('E:\MIMIC\医学期刊Web部署全攻略：打造多样化数据展示与预测系统教程 2025-3-16 8395 1\saki_akd_web部署\model.pkl')
scaler = joblib.load('E:\MIMIC\MIMIC_BIG_DATA\MIMIC_BIG_DATA\case\stroke\saki_scaler.pkl')
```

```
In [237... X_val= scaler.transform(X_val)
```

```
In [ ]]: from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, mean_squared_error, roc_auc_score, roc_curve, auc
import numpy as np

# 使用模型进行预测
y_pred = model.predict(X_val)
y_pred_proba = model.predict_proba(X_val)[:, 1] # 对于二分类问题，获取正类的预测概率

# 计算性能指标
accuracy = accuracy_score(y_val, y_pred)
precision = precision_score(y_val, y_pred)
recall = recall_score(y_val, y_pred)
f1 = f1_score(y_val, y_pred)

# 注意：均方误差通常用于回归问题，不适用于分类问题
# 如果你想计算分类问题的损失，可以使用交叉熵损失
# mean_squared_error在这里不适用，因此下面的代码是错误的，仅作为示例
mse = mean_squared_error(y_val, y_pred) # 错误的使用，分类问题不适用

# 计算AUC值
auc_score = roc_auc_score(y_val, y_pred_proba)

# 打印性能指标
print(f'Accuracy: {accuracy}')
print(f'Precision: {precision}')
print(f'Recall: {recall}')
print(f'F1 Score: {f1}')
print(f'AUC: {auc_score}')

# 如果需要，也可以绘制ROC曲线
import matplotlib.pyplot as plt

fpr, tpr, thresholds = roc_curve(y_val, y_pred_proba)
plt.figure()
plt.plot(fpr, tpr, color='darkorange', lw=2, label=f'ROC curve (area = {auc_score:.2f})')
plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver Operating Characteristic (ROC) Curve')
plt.legend(loc="lower right")
plt.show()
```

```
In [239... X_test=X_val
y_test=y_val
```

```
In [ ]]: import joblib
import pandas as pd
from sklearn.metrics import (
    accuracy_score,
    precision_score,
    recall_score,
    f1_score,
    mean_squared_error,
    roc_auc_score,
```

```
        confusion_matrix,
        roc_curve,
        auc
    )

# 3. 进行预测（确保特征处理与训练时一致）
y_pred = model.predict(X_test) # 类别预测
y_proba = model.predict_proba(X_test)[:, 1] # 正类概率（适用于二分类）

# 4. 计算性能指标
# 基础分类指标
accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred)
recall = recall_score(y_test, y_pred)
f1 = f1_score(y_test, y_pred)

# 均方误差（使用概率计算）
mse = mean_squared_error(y_test, y_proba)

# AUC-ROC
auc_roc = roc_auc_score(y_test, y_proba)

# 混淆矩阵
conf_matrix = confusion_matrix(y_test, y_pred)

# 打印结果
print(f"准确率 (Accuracy): {accuracy:.4f}")
print(f"精确率 (Precision): {precision:.4f}")
print(f"召回率 (Recall): {recall:.4f}")
print(f"F1分数 (F1-score): {f1:.4f}")
print(f"均方误差 (MSE): {mse:.4f}")
print(f"AUC-ROC: {auc_roc:.4f}")
print("\n混淆矩阵:")
print(conf_matrix)

# 可选：绘制ROC曲线
import matplotlib.pyplot as plt

fpr, tpr, thresholds = roc_curve(y_test, y_proba)
roc_auc = auc(fpr, tpr)

plt.figure()
plt.plot(fpr, tpr, color='darkorange', lw=2, label=f'ROC曲线 (AUC = {roc_auc:.2f})')
plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')
plt.xlabel('假正率')
plt.ylabel('真正率')
plt.title('接收者操作特征曲线 (ROC)')
plt.legend(loc="lower right")
plt.show()
```