Introduction

心衰和心梗【所有入院首诊为心衰心梗的病人，不限于ICU，若多次因为首诊心衰或者心梗入院，则提取第一次入院信息】：

1. 人口学基本信息和共病情况 data_info4
2. 实验室指标 data_chart & data_lab
3. 用药信息 data_pharmacy
4. 在icu停留的记录 data_icustay

Tasks

1. 把实验室指标、用药信息 长数据转成宽数据
2. 把上述表格合并成一个宽数据的表格

In [ ]:
```
import numpy as np
import pandas as pd
```

In [ ]:
```
data_info4 = pd.read_csv('data_info4.csv')
data_chart = pd.read_csv('data_chart.csv')
data_lab = pd.read_csv('data_lab.csv')
data_pharmacy = pd.read_csv('data_pharmacy.csv')
data_icustay = pd.read_csv('data_icustay.csv')
```

0. 基本信息

In [55]:
```
data_info4
```

Out[55]:

	subject_id	hadm_id	admittime	dischtime	admission_type	admission_location	discharge_location	insurance	language	marital_status	...	pept
0	10000980	29654838	3/1/2188 17:41:00	5/1/2188 17:30:00	EW EMER.	EMERGENCY ROOM	HOME HEALTH CARE	Medicare	ENGLISH	MARRIED	...	
1	10001492	27463908	23/9/2136 18:02:00	25/9/2136 17:45:00	EW EMER.	EMERGENCY ROOM	HOME	Medicare	ENGLISH	MARRIED	...	
2	10001877	21320596	21/11/2150 23:02:00	23/11/2150 16:46:00	EU OBSERVATION	EMERGENCY ROOM	NaN	Other	ENGLISH	MARRIED	...	
3	10002013	24760295	10/7/2160 19:33:00	12/7/2160 12:30:00	EW EMER.	EMERGENCY ROOM	HOME	Medicare	ENGLISH	SINGLE	...	
4	10002155	23822395	4/8/2129 12:44:00	18/8/2129 16:53:00	EW EMER.	PROCEDURE SITE	CHRONIC/LONG TERM ACUTE CARE	Other	ENGLISH	MARRIED	...	
...	...	...	...	...	...	...	...	...	...	...	...	...
12423	19996783	21880161	9/5/2188 15:55:00	19/5/2188 18:09:00	OBSERVATION ADMIT	WALK-IN/SELF REFERRAL	HOSPICE	Other	?	MARRIED	...	
12424	19997367	22967208	24/5/2127 18:33:00	27/5/2127 15:30:00	URGENT	TRANSFER FROM HOSPITAL	HOME HEALTH CARE	Medicare	ENGLISH	MARRIED	...	
12425	19997473	27787494	11/9/2173 00:53:00	2/10/2173 15:50:00	URGENT	TRANSFER FROM HOSPITAL	SKILLED NURSING FACILITY	Medicare	ENGLISH	MARRIED	...	
12426	19998330	23151993	20/9/2178 20:20:00	23/9/2178 18:30:00	EW EMER.	EMERGENCY ROOM	HOME	Other	ENGLISH	MARRIED	...	
12427	19998497	29288061	1/7/2139 16:19:00	5/7/2139 13:00:00	URGENT	TRANSFER FROM HOSPITAL	SKILLED NURSING FACILITY	Other	ENGLISH	WIDOWED	...	

12428 rows × 31 columns

In [65]:
```
data_info4.columns.tolist()
```

Out[65]:
```
['subject_id',
 'hadm_id',
 'admittime',
 'dischtime',
 'admission_type',
 'admission_location',
 'discharge_location',
 'insurance',
 'language',
 'marital_status',
 'ethnicity',
 'status',
 'gender',
 'age',
 'myocardial_infarct',
 'congestive_heart_failure',
 'peripheral_vascular_disease',
 'cerebrovascular_disease',
 'dementia',
 'chronic_pulmonary_disease',
 'rheumatic_disease',
 'peptic_ulcer_disease',
 'mild_liver_disease',
 'diabetes_without_cc',
 'diabetes_with_cc',
 'paraplegia',
 'renal_disease',
 'malignant_cancer',
 'severe_liver_disease',
 'metastatic_solid_tumor',
 'aids']
```

0.1 详细

info4 更多 (https://mimic.mit.edu/docs/iv/modules/hosp/admissions/%5D(https://mimic.mit.edu/docs/iv/modules/icu/chartevents/): 有关住院的详细信息。

> 入院表提供了关于病人入院的信息。由于病人的每一次独特的医院访问都被分配了一个独特的hdm_id，所以入院表可以被认为是hdm_id的定义表。可用的信息包括入院和出院的时间信息、人口统计信息、入院的来源等等。

- 数据来自医院的入院、出院和转院数据库（通常称为“ADT”数据）。
- 有时会为在医院死亡的患者创建器官捐献者账户。这些是不同的住院时间，住院时间很短，有时甚至是负数。此外，他们的死亡时间通常与较早入院患者的死亡时间相同。

- subject_id,hadm_id: 该表的每一行都包含一个唯一的 hadm_id，代表单个患者入院。 hadm_id的取值范围是2000000-2999999，这张表subject_id可能重复，说明一个病人多次入院。可以使用 subject_id 将 ADMISSIONS 表链接到 PATIENTS 表。
- admittime dischtime, (入院时间、离院时间、死亡时间): admittime提供病人入院的日期和时间，dischtime则提供病人出院的日期和时间。如果适用，deathtime提供病人在医院内死亡的时间。请注意，只有当病人在医院内死亡时才会出现deathtime，而且几乎总是与病人的dischtime相同。然而，由于打字错误，可能会有一些差异。

> 💡 deathtime 无

- admission_type用于对入院的紧迫性进行分类。有9种可能性。门诊观察"、"直接急诊"、"直接观察"、"择期"、"紧急观察"、"观察入院"、"当天手术入院"、"紧急"。
- admission_location提供病人到达医院之前的位置信息。请注意，由于急诊室在技术上是一个诊所，通过急诊室入院的病人通常将其作为入院地点。
- 同样，discharge_location 是患者出院后的处置情况。

- admission_type用于对入院的紧迫性进行分类。有9种可能性。门诊观察"、"直接急诊"、"直接观察"、"择期"、"紧急观察"、"观察入院"、"当天手术入院"、"紧急"。
- insurance, language, marital_status, ethnicity (保险、语言、婚姻状况、种族)： 保险、语言、婚姻状况和种族列提供有关给定住院的患者人口统计信息。请注意，由于每次入院都会记录此数据，因此它们可能会因住院而异。

> 💡 edregtime, edouttime (暂无) 患者在急诊室登记和出院的日期和时间。

🎯 未完成的注释

- `status` :
- `gender` :
- `age` :
- `myocardial_infarct` :
- `congestive_heart_failure` : 心衰

- peripheral_vascular_disease :
- cerebrovascular_disease :
- dementia :
- chronic_pulmonary_disease :
- rheumatic_disease :
- peptic_ulcer_disease : xx溃疡
- mild_liver_disease :
- diabetes_without_cc :
- diabetes_with_cc :
- paraplegia :
- renal_disease :
- malignant_cancer :,
- severe_liver_disease : 严重的肺x
- metastatic_solid_tumor :
- aids :

Links to:

- patients on `subject_id`
- admissions on `hadm_id`
- icustays on `stay_id`
- d_items on `itemid`

1. Chart表

In [52]:
```
data_chart
```

Out[52]:

	subject_id	hadm_id	charttime	storetime	itemid	value	valuenum
0	10464977	23034003	7/11/2140 20:58:00	7/11/2140 20:58:00	223956	Ventricular Demand	NaN
1	10464977	23034003	7/11/2140 20:58:00	7/11/2140 20:58:00	223961	Attached-Pacer	NaN
2	10464977	23034003	7/11/2140 20:58:00	7/11/2140 20:58:00	224390	No	NaN
3	10464977	23034003	7/11/2140 20:58:00	7/11/2140 20:58:00	224834	Yes	NaN
4	10464977	23034003	7/11/2140 20:58:00	7/11/2140 20:58:00	224835	Yes	NaN
...	...	...	...	...	...	...	...
18504547	10464977	23034003	3/11/2140 00:00:00	3/11/2140 03:12:00	227947	Confusion	NaN
18504548	10464977	23034003	3/11/2140 00:00:00	3/11/2140 03:12:00	227948	Soft limb	NaN
18504549	10464977	23034003	3/11/2140 00:00:00	3/11/2140 03:12:00	227949	Done	NaN
18504550	10464977	23034003	3/11/2140 00:00:00	3/11/2140 03:12:00	227950	Both arms	NaN
18504551	10464977	23034003	3/11/2140 00:00:00	3/11/2140 03:12:00	227951	Circulation adequate	NaN

18504552 rows × 7 columns

1.1 详细

chartevent 更多 (https://mimic.mit.edu/docs/iv/modules/icu/chartevents/): 在 ICU 逗留期间发生的图表项目。包含 ICU 中记录的大部分信息。

> charterevents 包含所有可用于患者的图表数据。在他们入住 ICU 期间，患者信息的主要存储库是他们的电子图表。电子图表显示患者的常规生命体征和与他们的护理相关的任何其他信息：**呼吸机**设置、**实验室**值、**代码**状态、**精神**状态等。因此，有关患者住院的大部分信息都包含在图表事件中。此外，即使在其他地方（labevents）捕获了实验室值，它们也经常在 charterevents 中重复。发生这种情况是因为希望在患者的电子图表上显示实验室值，因此将值从存储实验室值的数据库复制到存储图表事件的数据库。

- 指定患者的标识符： `subject_id` 对于患者是唯一的， `hadm_id` 对于患者住院是唯一的， `stay_id` 对于患者住院是唯一的。有关这些标识符的更多信息，请参见此处。
- `charttime` 记录进行观察的时间，通常是最接近实际测量数据的时间。 `storetime` 记录临床工作人员手动输入或手动验证观察结果的时间。
- 数据库中单一测量类型的标识符。与一个 `ITEMID` （如212）相关的每一行都对应于同一测量（如心率）的一个实例。
- `value` 包含为 ITEMID 标识的概念测量的值。如果此值为数字，则 valuenum 包含数字格式的相同数据。如果此数据不是数字，则 valuenum 为空。在某些情况下（例如 Glasgow Coma Scale、Richmond Sedation Agititation Scale 和 Code Status 等分数），valuenum 包含分数，value 包含分数和描述分数含义的文本.

Links to:

- patients on `subject_id`
- admissions on `hadm_id`
- icustays on `stay_id`
- d_items on `itemid`

2. Lab表

In [53]:
```
data_lab
```

Out[53]:

	subject_id	hadm_id	charttime	storetime	itemid	value	valuenum
0	10013653	26666796	26/10/2182 07:40:00	26/10/2182 09:02:00	50960	2.0	2.00
1	10013653	26666796	26/10/2182 00:57:00	26/10/2182 01:02:00	50802	-2	-2.00
2	10013653	26666796	26/10/2182 00:57:00	26/10/2182 01:02:00	50804	23	23.00
3	10013653	26666796	26/10/2182 00:57:00	26/10/2182 01:02:00	50818	37	37.00
4	10013653	26666796	26/10/2182 00:57:00	26/10/2182 01:02:00	50820	7.38	7.38
...	...	...	...	...	...	...	...
3197437	16221250	23282331	26/9/2140 22:22:00	26/9/2140 22:50:00	51508	Red	NaN
3197438	12741325	20860534	26/1/2121 12:35:00	26/1/2121 14:16:00	51508	PINK	NaN
3197439	19119896	28318588	2/4/2195 16:00:00	2/4/2195 16:54:00	51508	Red	NaN
3197440	19506871	21026986	11/4/2118 22:50:00	11/4/2118 23:30:00	51508	DKAMB	NaN
3197441	18426598	22940206	24/4/2173 01:30:00	24/4/2173 02:38:00	51508		NaN

3197442 rows × 7 columns

2.1详细

labevents (https://mimic.mit.edu/docs/iv/modules/hosp/labevents/): 来自患者样本的实验室测量。

> labevents 表存储为单个患者进行的所有实验室测量的结果。这些包括血液学测量、血气、化学面板和不太常见的测试，如基因检测。

`hadm_id` 是使用转院表分配给靠近住院的实验室的。然而，这并不总是能完美地捕捉到与住院时间相近的实验室。具体来说，从v2.1版开始，通过连接入院病人的 `subject_id` 、入院时间和 `dischtime` ，可以为59,327,830个观测值分配 `hadm_id` 。然而，这些观测值中只有58,131,956个（98%）有一个 `hadm_id` 实际存储在**labevents**表中。希望确保捕获与住院时间相近的实验室的用户应该意识到这一点，并在必要时使用带有时间的连接。

- `charttime` : 绘制实验室测量值的时间。这通常是获取标本的时间，通常明显早于可进行测量的时间。
- `storetime` : 在实验室系统中提供测量值的时间。这是信息提供给护理提供者的时间。
- `itemid` : 一个标识符，唯一地表示实验室概念
- `value & valuenum` : 实验室测量的结果，如果是数字，则转换为数字数据类型的值。

Links to

- d_labitems(另外一张表) on `itemid`

3. 用药表

In [56]:
```
data_pharmacy
```

Out[56]:

	subject_id	hadm_id	starttime	stoptime	medication	frequency
0	15346117	24544765	13/1/2201 00:00:00	18/1/2201 14:00:00	Sodium Chloride 0.9% Flush	Q8H
1	15346117	24544765	13/1/2201 00:00:00	21/1/2201 19:00:00	Heparin	TID
2	15346117	24544765	13/1/2201 01:00:00	13/1/2201 16:00:00	Acetaminophen	Q6H:PRN
3	15346117	24544765	13/1/2201 01:00:00	21/1/2201 19:00:00	Amlodipine	DAILY
4	15346117	24544765	13/1/2201 01:00:00	21/1/2201 19:00:00	Aspirin	DAILY
...	...	...	...	...	...	...
667601	17614063	26391110	22/4/2177 14:00:00	22/4/2177 20:00:00	Fluzone Trival	ONCE
667602	12361593	28465050	16/7/2147 13:00:00	16/7/2147 19:00:00	Fluzone Trival	ONCE
667603	17050261	28079544	20/4/2184 15:00:00	20/4/2184 20:00:00	Fluzone Trival	ONCE
667604	18749775	26585465	22/7/2147 04:00:00	24/7/2147 12:00:00	COQ10 (300 mg)	Q 12H
667605	16112663	24076527	8/12/2167 08:00:00	8/12/2167 17:00:00	Tenecteplase	2X

667606 rows × 6 columns

3.1 详细

pharmacy 更多指标解释 (https://mimic.mit.edu/docs/iv/modules/hosp/pharmacy/): 处方药的处方、剂量和其他信息。

> 药房表提供了有关开给患者的填充药物的详细信息。药房信息包括药物剂量、处方剂量数、给药频率、用药途径和处方持续时间。

- `subject_id` : 是一个唯一的标识符，它指定了一个单独的病人。任何与单个subject_id相关的行都与同一个人有关。
- `hadm_id` : 是一个整数标识符，对每个病人的住院治疗是唯一的。
- `starttime` `stoptime` : 给定的处方药的开始和停止时间
- `medication` : 提供的药物名称。
- `frequency` : 应向患者给药的频率。许多常用的短手都用在频率栏中。 Q#表示每隔#小时；例如“Q6”或“Q6H”是每 6 小时一班。

Links to

- poe on `poe_id`
- prescriptions on `pharmacy_id`
- emar on `pharmacy_id`

4. icu停留 表

In [57]:
```
data_icustay
```

Out[57]:

	subject_id	hadm_id	stay_id	first_careunit	last_careunit	intime	outtime	los
0	11679839	29636865	30021881	Coronary Care Unit (CCU)	Coronary Care Unit (CCU)	19/6/2112 14:41:32	20/6/2112 21:50:49	1.298113
1	13787728	27446327	30024343	Coronary Care Unit (CCU)	Coronary Care Unit (CCU)	17/2/2123 18:52:51	18/2/2123 13:39:32	0.782419
2	19352034	21509557	30058866	Coronary Care Unit (CCU)	Coronary Care Unit (CCU)	18/4/2114 02:02:38	30/4/2114 21:41:14	12.818472
3	18817948	23379351	30061662	Coronary Care Unit (CCU)	Coronary Care Unit (CCU)	2/11/2142 22:18:49	4/11/2142 00:46:50	1.102789
4	14964123	25103078	30081334	Surgical Intensive Care Unit (SICU)	Surgical Intensive Care Unit (SICU)	15/6/2137 16:33:00	17/6/2137 18:28:59	2.080544
...	...	...	...	...	...	...	...	...
4821	19420059	22049257	39920422	Cardiac Vascular Intensive Care Unit (CVICU)	Cardiac Vascular Intensive Care Unit (CVICU)	17/1/2114 08:55:42	22/1/2114 13:16:30	5.181111
4822	12673141	21284928	39921830	Cardiac Vascular Intensive Care Unit (CVICU)	Coronary Care Unit (CCU)	19/4/2131 05:28:27	23/4/2131 21:17:05	4.658773
4823	10836444	25551438	39934059	Cardiac Vascular Intensive Care Unit (CVICU)	Cardiac Vascular Intensive Care Unit (CVICU)	9/12/2170 13:50:58	10/12/2170 17:23:26	1.147546
4824	17014465	25960753	39966638	Cardiac Vascular Intensive Care Unit (CVICU)	Cardiac Vascular Intensive Care Unit (CVICU)	5/2/2112 09:23:57	6/2/2112 17:00:33	1.317083
4825	12275003	22562812	39992247	Cardiac Vascular Intensive Care Unit (CVICU)	Cardiac Vascular Intensive Care Unit (CVICU)	15/8/2182 09:37:33	16/8/2182 17:25:44	1.325127

4826 rows × 8 columns

详细

ICU stays 更多 (https://mimic.mit.edu/docs/iv/modules/icu/icustays/): ICU 停留的跟踪信息，包括入院和出院时间。

- `stay_id` 是一个生成的标识符，不是基于任何原始数据标识符。医院和ICU的数据库没有内在的联系，所以没有任何ICU遭遇标识符的概念。
- `icustays` 表是由 `transfer` 表衍生出来的。具体来说，它排除了病房不是ICU的转院记录。
- `transfer` 表中多个连续的ICU住院条目被合并为 `transfer` 表中的一个条目。
- 心率测量被用来确定重症监护室住院是否有效。如果 `transfer` 表中的重症监护室停留的心率测量值不可用，重症监护室停留将不包括在 `icustays` 表中（~5%）。

- `subject_id & hadm_id` : 指定患者的标识符： `subject_id` 对于患者是唯一的， `hadm_id` 对于患者住院是唯一的， `stay_id` 对于患者住院是唯一的。
- `first_careunit & last_careunit` : `FIRST_CAREUNIT` 和 `LAST_CAREUNIT` 分别包含病人被护理的第一个和最后一个ICU类型。由于一个stay_id将所有在24小时内入院的重症监护室分组，所以病人有可能从一种重症监护室转到另一种重症监护室而拥有相同的 `stay_id` 。
- `intime & outtime` : INTIME 提供患者被转移到 ICU 的日期和时间。 OUTTIME 提供患者转出 ICU 的日期和时间。
- `los` : LOS 是患者在给定的 ICU 住院期间的住院时间，可能包括一个或多个 ICU 病房。停留时间以小数天数计算。

Links to:

- patients on `subject_id`
- admissions on `hadm_id`

In [45]:
```
print('chart表，实验室表，以及用药表的行数:{}, {}, {}'.format(data_chart.shape[0], data_lab.shape[0], data_pharmacy.shape[0]))
```

```
chart表，实验室表，以及用药表的行数:18504552, 3197442, 667606
```

In [58]:
```
print('Char表、实验表、用药表中的 住院号数量: {}, {}, {}'.format(len(data_chart['hadm_id'].value_counts()), len(data_lab['hadm_i
```

```
Char表、实验表、用药表中的 住院号数量: 4297, 12257, 12352
```

In [59]:
```
print('Char表、实验表、用药表中的 病人数量: {}, {}, {}'.format(len(data_chart['subject_id'].value_counts()), len(data_lab['subje
```

```
Char表、实验表、用药表中的 病人数量: 4297, 12257, 12352
```

案例分析

但病人住院号分析

In [49]:
```
data_chart[data_chart['hadm_id'] == 23034003]
```

Out[49]:

	subject_id	hadm_id	charttime	storetime	itemid	value	valuenum
0	10464977	23034003	7/11/2140 20:58:00	7/11/2140 20:58:00	223956	Ventricular Demand	NaN
1	10464977	23034003	7/11/2140 20:58:00	7/11/2140 20:58:00	223961	Attached-Pacer	NaN
2	10464977	23034003	7/11/2140 20:58:00	7/11/2140 20:58:00	224390	No	NaN
3	10464977	23034003	7/11/2140 20:58:00	7/11/2140 20:58:00	224834	Yes	NaN
4	10464977	23034003	7/11/2140 20:58:00	7/11/2140 20:58:00	224835	Yes	NaN
...	...	...	...	...	...	...	...
18504547	10464977	23034003	3/11/2140 00:00:00	3/11/2140 03:12:00	227947	Confusion	NaN
18504548	10464977	23034003	3/11/2140 00:00:00	3/11/2140 03:12:00	227948	Soft limb	NaN
18504549	10464977	23034003	3/11/2140 00:00:00	3/11/2140 03:12:00	227949	Done	NaN
18504550	10464977	23034003	3/11/2140 00:00:00	3/11/2140 03:12:00	227950	Both arms	NaN
18504551	10464977	23034003	3/11/2140 00:00:00	3/11/2140 03:12:00	227951	Circulation adequate	NaN

14032 rows × 7 columns

In [50]:
```
data_lab[data_lab['hadm_id'] == 23034003]
```

Out[50]:

	subject_id	hadm_id	charttime	storetime	itemid	value	valuenum
15521	10464977	23034003	7/11/2140 18:59:00	7/11/2140 19:03:00	50817	98	98.0
15618	10464977	23034003	2/11/2140 23:55:00	3/11/2140 01:39:00	50868	13	13.0
15619	10464977	23034003	2/11/2140 23:55:00	3/11/2140 01:39:00	50882	25	25.0
15620	10464977	23034003	2/11/2140 23:55:00	3/11/2140 01:39:00	50893	8.1	8.1
15621	10464977	23034003	2/11/2140 23:55:00	3/11/2140 01:39:00	50902	105	105.0
...	...	...	...	...	...	...	...
2940296	10464977	23034003	10/11/2140 00:20:00	10/11/2140 01:00:00	50920	NaN	NaN
2940555	10464977	23034003	2/11/2140 23:55:00	3/11/2140 01:39:00	50920	NaN	NaN
2942371	10464977	23034003	9/11/2140 02:22:00	9/11/2140 03:41:00	51301	14.2	14.2
2943759	10464977	23034003	4/11/2140 15:39:00	4/11/2140 17:26:00	51265	95	95.0
2943761	10464977	23034003	6/11/2140 11:28:00	6/11/2140 12:06:00	51265	84	84.0

825 rows × 7 columns

In [60]:
```
data_pharmacy[data_pharmacy['hadm_id'] == 23034003]
```

Out[60]:

	subject_id	hadm_id	starttime	stoptime	medication	frequency
207407	10464977	23034003	3/11/2140 23:00:00	4/11/2140 22:00:00	Furosemide	ONCE
207410	10464977	23034003	3/11/2140 23:00:00	6/11/2140 17:00:00	Insulin	ASDIR
207412	10464977	23034003	3/11/2140 00:00:00	6/11/2140 17:00:00	Influenza Virus Vaccine	NOW X1
207413	10464977	23034003	3/11/2140 00:00:00	6/11/2140 17:00:00	Pneumococcal Vac Polyvalent	NOW X1
207414	10464977	23034003	3/11/2140 08:00:00	6/11/2140 17:00:00	Docusate Sodium (Liquid)	BID
...	...	...	...	...	...	...
207556	10464977	23034003	11/11/2140 14:00:00	12/11/2140 13:00:00	Warfarin	ONCE
207557	10464977	23034003	12/11/2140 13:00:00	13/11/2140 12:00:00	Warfarin	ONCE
207558	10464977	23034003	12/11/2140 10:00:00	13/11/2140 09:00:00	Furosemide	DAILY
207559	10464977	23034003	13/11/2140 10:00:00	13/11/2140 19:00:00	Warfarin	ONCE
207560	10464977	23034003	13/11/2140 10:00:00	13/11/2140 19:00:00	Furosemide	DAILY

137 rows × 6 columns

In [38]:
```
hadm_chart = data_chart['hadm_id'].unique().tolist()
```

In [63]:
```
# data_chart['hadm_id'] data_lab['']
print('lab表中的hadm_id号包含多少chart表中的hadm号: {}'.format(len(data_lab.hadm_id.isin(data_chart.hadm_id))))
data_lab.shape[0]
##结论：lab表中的实验号在 chart中能够全部找到
```

```
lab表中的hadm_id号包含多少chart表中的hadm号: 3197442
```

Out[63]:
```
3197442
```