hw_3

October 20, 2024

[1]:
```
import pandas as pd
from tqdm import tqdm
import numpy as np
import matplotlib.pyplot as plt
```

0.0.1

[2]:
```
dtypes = {
    'row_id': 'int32',
    'timestamp': 'int64',
    'user_id': 'int32',
    'content_id': 'int16',
    'content_type_id': 'int8',
    'task_container_id': 'int16',
    'user_answer': 'int8',
    'answered_correctly': 'int8',
    'prior_question_elapsed_time': 'float32',
    'prior_question_had_explanation': 'boolean'
}
```

[3]:
```
train = pd.read_csv('data/train.csv', dtype=dtypes)
lectures = pd.read_csv('data/lectures.csv')
questions = pd.read_csv('data/questions.csv')
```

0.0.2

[4]:
```
train = train.drop(['row_id'], axis=1)
```

0.0.3

[5]:
```
train
```

[5]:

	timestamp	user_id	content_id	content_type_id	\
0	0	115	5692	0	
1	56943	115	5716	0	
2	118363	115	128	0	
3	131167	115	7860	0	

```
4              137965         115        7922                0
…                …            …          …                 …
101230327   428564420  2147482888        3586                0
101230328   428585000  2147482888        6341                0
101230329   428613475  2147482888        4212                0
101230330   428649406  2147482888        6343                0
101230331   428692118  2147482888        7995                0

            task_container_id  user_answer  answered_correctly  \
0                           1            3                   1
1                           2            2                   1
2                           0            0                   1
3                           3            0                   1
4                           4            1                   1
…                         …            …                   …
101230327                  22            0                   1
101230328                  23            3                   1
101230329                  24            3                   1
101230330                  25            1                   0
101230331                  26            3                   1

            prior_question_elapsed_time  prior_question_had_explanation
0                                   NaN                           <NA>
1                               37000.0                           False
2                               55000.0                           False
3                               19000.0                           False
4                               11000.0                           False
…                                 …                             …
101230327                       18000.0                            True
101230328                       14000.0                            True
101230329                       14000.0                            True
101230330                       22000.0                            True
101230331                       29000.0                            True

[101230332 rows x 9 columns]
```

[6]:
```
questions
```

[6]:
```
        question_id  bundle_id  correct_answer  part           tags
0                 0          0               0     1   51 131 162 38
1                 1          1               1     1       131 36 81
2                 2          2               0     1  131 101 162 92
3                 3          3               0     1  131 149 162 29
4                 4          4               3     1    131 5 162 38
…               …          …               …   …              …
13518         13518      13518               3     5             14
13519         13519      13519               3     5              8
```

```
13520        13520       13520                2     5              73
13521        13521       13521                0     5             125
13522        13522       13522                3     5              55

[13523 rows x 5 columns]
```

[7]:
```
lectures
```

[7]:
```
     lecture_id  tag  part          type_of
0            89  159     5          concept
1           100   70     1          concept
2           185   45     6          concept
3           192   79     5  solving question
4           317  156     5  solving question
..          …  …   …              …
413       32535    8     5  solving question
414       32570  113     3  solving question
415       32604   24     6          concept
416       32625  142     2          concept
417       32736   82     3          concept

[418 rows x 4 columns]
```

0.0.4 , train : 101 . ,

[8]:
```
d_parts = {questions.iloc[i, 0]: questions.iloc[i, 3] for i in␣
↪range(len(questions))}
```

[9]:
```
content_type_questions = set(train[train.content_type_id == 0].index)
```

0.0.5 ,

[10]:
```
d_content_id = dict(zip(list(range(len(train))), train.content_id.values.
↪tolist()))
```

[11]:
```
parts = []
for i in tqdm(range(len(train))):
    if i in content_type_questions:
        parts.append(d_parts[d_content_id[i]])
    else:
        parts.append(-1)
```

```
100%|      | 101230332/101230332 [00:14<00:00, 6964338.51it/s]
```

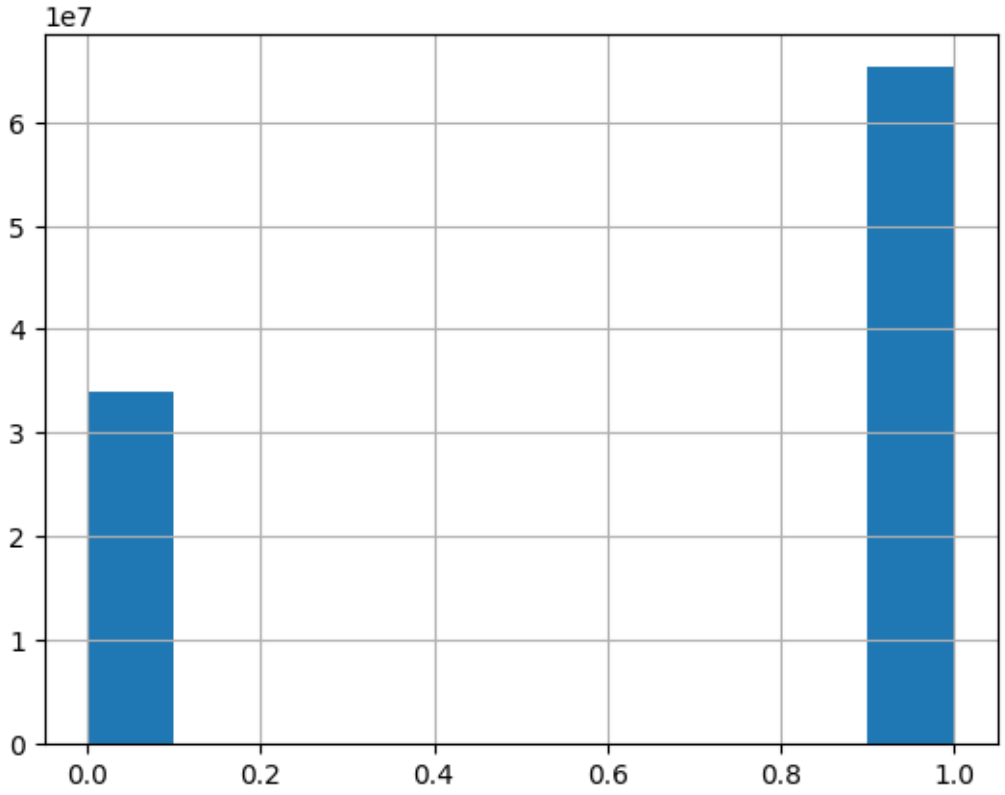
[12]:
```
parts = np.array(parts)
assert (parts != -1).sum() == len(train[train.content_type_id == 0])
```

[13]:
```
train['parts'] = parts
```

0.0.6

[14]:
```
train[train.content_type_id == 0].answered_correctly.hist()
```

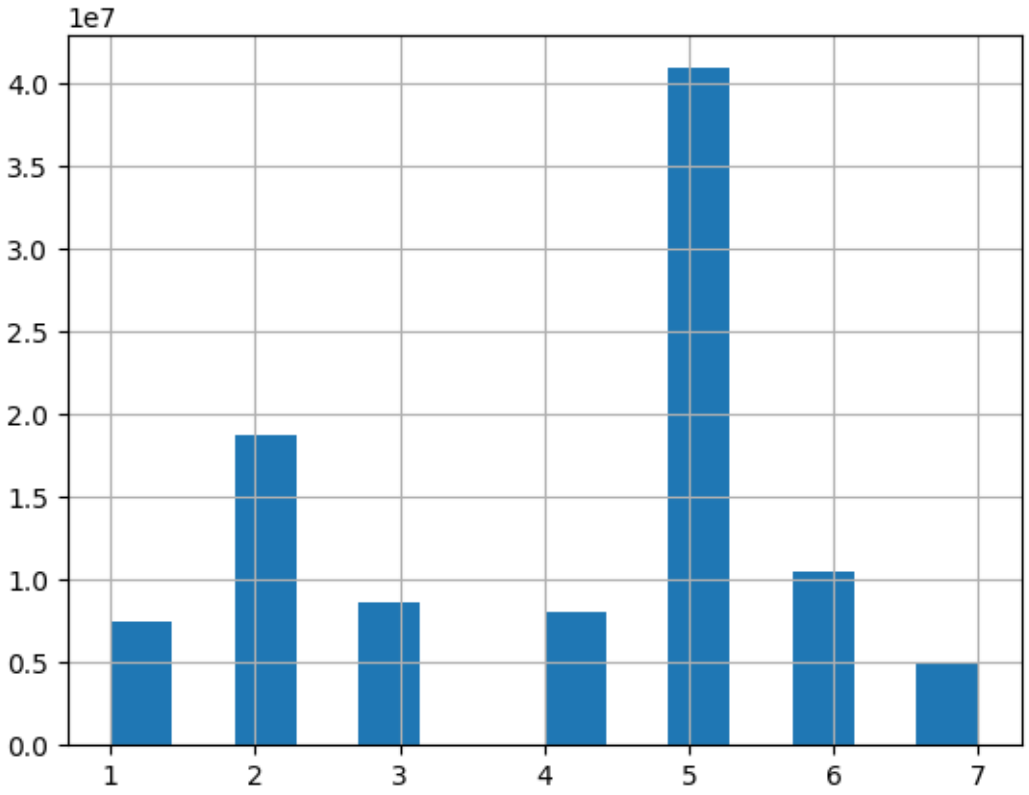
[14]: <Axes: >



0.0.7 : , 5

[15]:
```
train[train.parts != -1].parts.hist(bins=14)
```

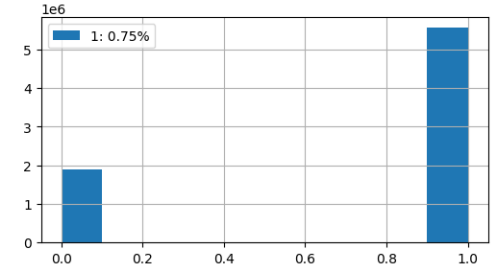
[15]: <Axes: >

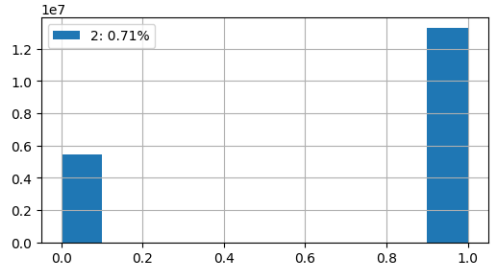


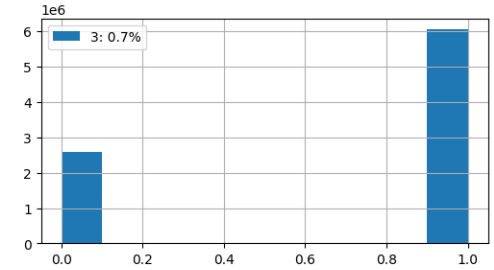
0.0.8 **: 5** ,

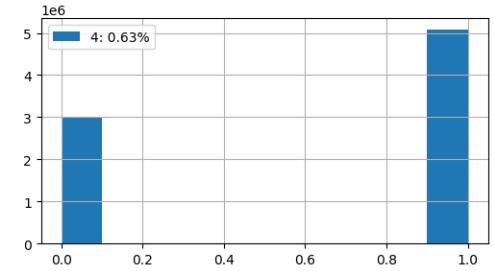
-

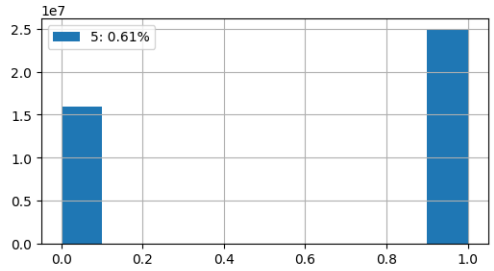
[16]:
```
fig, ax = plt.subplots(nrows=7, ncols=1, figsize=(6, 25))
for i in range(1, 8):
    train[train.parts == i].answered_correctly.hist(ax=ax[i-1])
    correct = (train[train.parts == i].answered_correctly == 1).sum()
    all = train[train.parts == i].shape[0]
    perc_correct = round(correct / all, 2)
    ax[i-1].legend([f'{i}: {perc_correct}%'])
```

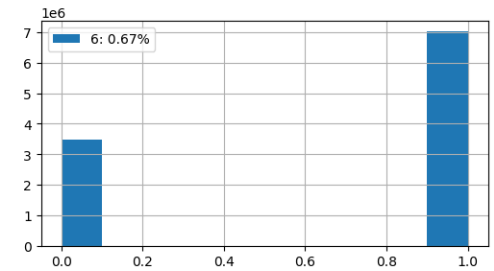

1e6
1: 0.75%
0.0 0.2 0.4 0.6 0.8 1.0

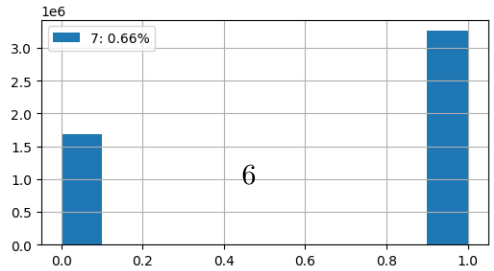

1e7
2: 0.71%
0.0 0.2 0.4 0.6 0.8 1.0


1e6
3: 0.7%
0.0 0.2 0.4 0.6 0.8 1.0


1e6
4: 0.63%
0.0 0.2 0.4 0.6 0.8 1.0


1e7
5: 0.61%
0.0 0.2 0.4 0.6 0.8 1.0


1e6
6: 0.67%
0.0 0.2 0.4 0.6 0.8 1.0


1e6
7: 0.66%
0.0 0.2 0.4 0.6 0.8 1.0

0.0.9 , , ,

[17]:
```
train_temp = train[train.content_type_id == 0]
```

[18]:
```
train_temp.parts.corr(train_temp.answered_correctly)
```

[18]:
```
-0.07505331429579741
```

[19]:
```
train_temp.prior_question_elapsed_time.corr(train_temp.answered_correctly)
```

[19]:
```
-0.007900239787032895
```

0.0.10 , . , , 30

[20]:
```
train_temp = train[train.content_type_id == 1]
```

[21]:
```
d_lecture_parts = {}
for i in tqdm(range(len(lectures))):
    d_lecture_parts[lectures.iloc[i, 0]] = lectures.iloc[i, 2]
```

```
100%|      | 418/418 [00:00<00:00, 58256.16it/s]
100%|      | 418/418 [00:00<00:00, 58256.16it/s]
```

[22]:
```
d_users_watched_lectures = {}
for x in tqdm(train.user_id.unique()):
    d_users_watched_lectures[x] = set()
for i in tqdm(range(len(train_temp))):
    d_users_watched_lectures[train_temp.iloc[i, 1]].
 ↪add(d_lecture_parts[train_temp.iloc[i, 2]])
```

```
100%|      | 393656/393656 [00:03<00:00, 113722.39it/s]
100%|      | 1959032/1959032 [00:32<00:00, 59980.50it/s]
```

[23]:
```
d_users = dict(zip(list(range(len(train))), train.user_id.values.tolist()))
```

[24]:
```
watched_lecture = []
for i in tqdm(range(len(train))):
    if i in content_type_questions:
        watched = int(parts[i] in d_users_watched_lectures[d_users[i]])
        watched_lecture.append(watched)
    else:
        watched_lecture.append(-1)
```

```
  0%|          | 0/101230332 [00:00<?, ?it/s]100%|      |
101230332/101230332 [01:45<00:00, 956885.29it/s]
```

[25]:
```
train['watched_lecture'] = watched_lecture
```

0.0.11 -

[26]:
```
train_temp = train[train.content_type_id == 0]
```

[27]:
```
train_temp.watched_lecture.corr(train_temp.answered_correctly)
```

[27]:
```
0.03628232049951815
```

[ ]:
```

```