HW04_Age_Kulikov

May 17, 2016

1 Linear Regression + k-means – Age prediction

1.1 Kulikov Alex, gr. 397

In [2]:
```
import numpy as np
import pandas as pd
from sklearn.linear_model import LinearRegression
from sklearn.cross_validation import cross_val_score
from sklearn.feature_extraction.text import HashingVectorizer

%pylab inline
%load_ext autoreload
%autoreload 2

from IPython.core.display import HTML
HTML("<style>.container { width:90% !important; }</style>")
```

```
Populating the interactive namespace from numpy and matplotlib
The autoreload extension is already loaded. To reload it, use:
  %reload_ext autoreload
```

Out[2]: <IPython.core.display.HTML object>

1.2 Data analysis

1.2.1 age

In [3]:
```
age_train_df = pd.read_csv('kaggle_data/age_profile_train', header=None, delimiter='\t')
age_train_df.columns = ['id', 'age']
```

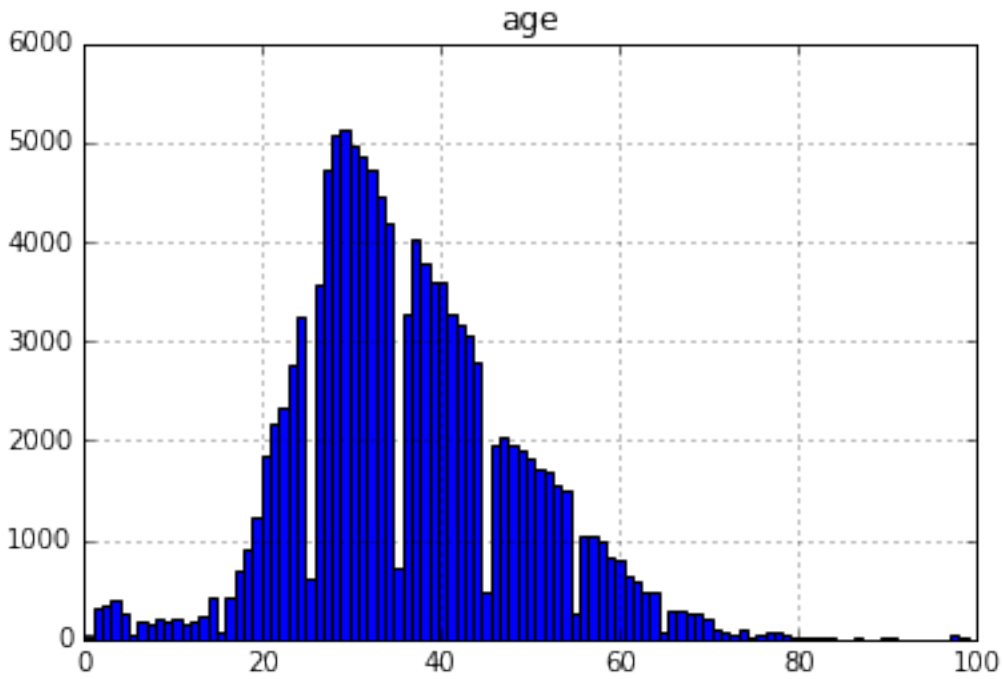
In [4]:
```
age_train_df.head()
```

Out[4]:

	id	age
0	000000013CB5719C0000A2C90002C101	53
1	00000001442BE24000001B7D00F50801	48
2	00000001448580F800003F1B31FB0901	28
3	0000000145BDB2FF000157971645E901	44
4	000000014602771F0000DB9359714C01	48

In [5]:
```
age_train_df.hist(column="age", bins=100)
```

Out[5]: array([[<matplotlib.axes._subplots.AxesSubplot object at 0x7fa8ca595390>]], dtype=object)



1.2.2 urls

In [6]:
```
urls_train_df = pd.read_csv('kaggle_data/url_domain_train', header=None, delimiter='\t')
urls_train_df.columns = ['id', 'url', 'count']
```

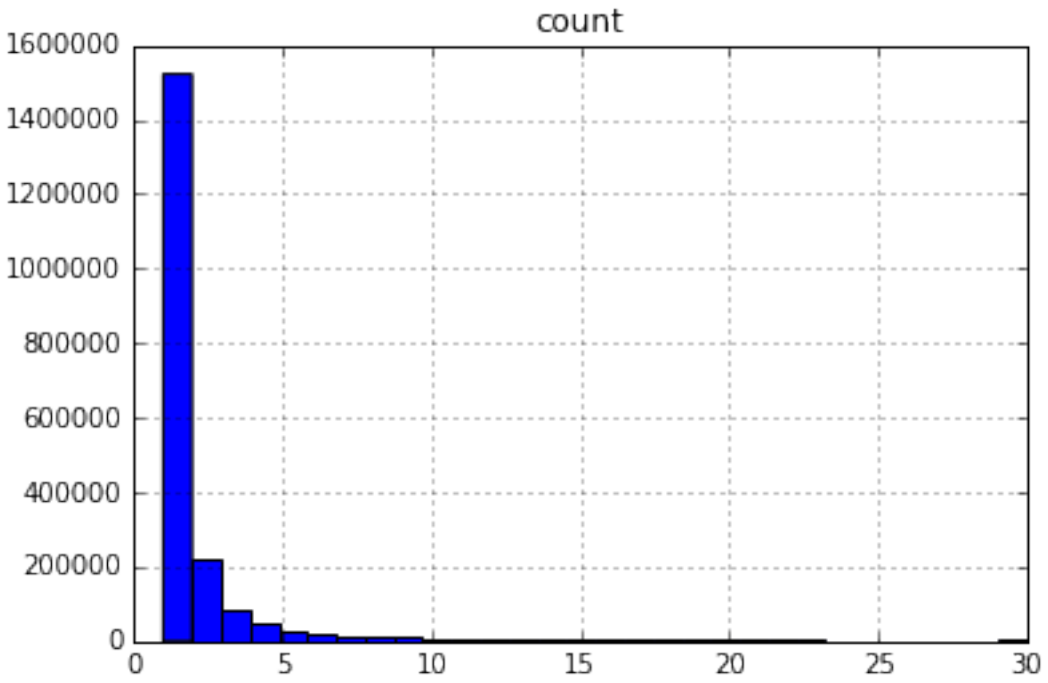
In [7]:
```
urls_train_df.head()
```

Out[7]:

	id	url	count
0	000000014B60815F65B38258011B6C01	login.rutracker.org	1
1	000000014B60815F65B38258011B6C01	rutracker.org	4
2	000000014C03DA2A47AC433A0C755201	admin.tour-spb.net	1
3	000000014C03DA2A47AC433A0C755201	czinfo.ru	1
4	000000014C03DA2A47AC433A0C755201	forumsostav.ru	1

In [8]:
```
urls_train_df.hist(column="count", bins=30)
```

Out[8]:
```
array([[<matplotlib.axes._subplots.AxesSubplot object at 0x7fa8c5bb3690>]], dtype=object)
```



For this time, we omit the "count" column

```
In [9]: urls_train_df = pd.DataFrame(urls_train_df.groupby('id')['url'].apply(lambda x: x.tolist()))
        urls_train_df['id'] = urls_train_df.index
        urls_train_df.index = range(len(urls_train_df))
        urls_train_df.columns = ['urls', 'id']

In [10]: urls_train_df.head()

Out[10]:                                                urls  \
         0            [id.rambler.ru, mail.rambler.ru, r0.ru]
         1  [1prime.ru, autorambler.ru, chellak.ru, docs.c...
         2                                  [bosch-korolev.ru]
         3  [aptekanizkihcen.ua, colady.ru, gorod.dp.ua, i...
         4  [astrorok.ru, diets.ru, edaplus.info, eshzdoro...

                                       id
         0  000000013CB5719C0000A2C90002C101
         1  00000001442BE24000001B7D00F50801
         2  00000001448580F800003F1B31FB0901
         3  0000000145BDB2FF000157971645E901
         4  000000014602771F0000DB9359714C01
```

1.2.3 titles

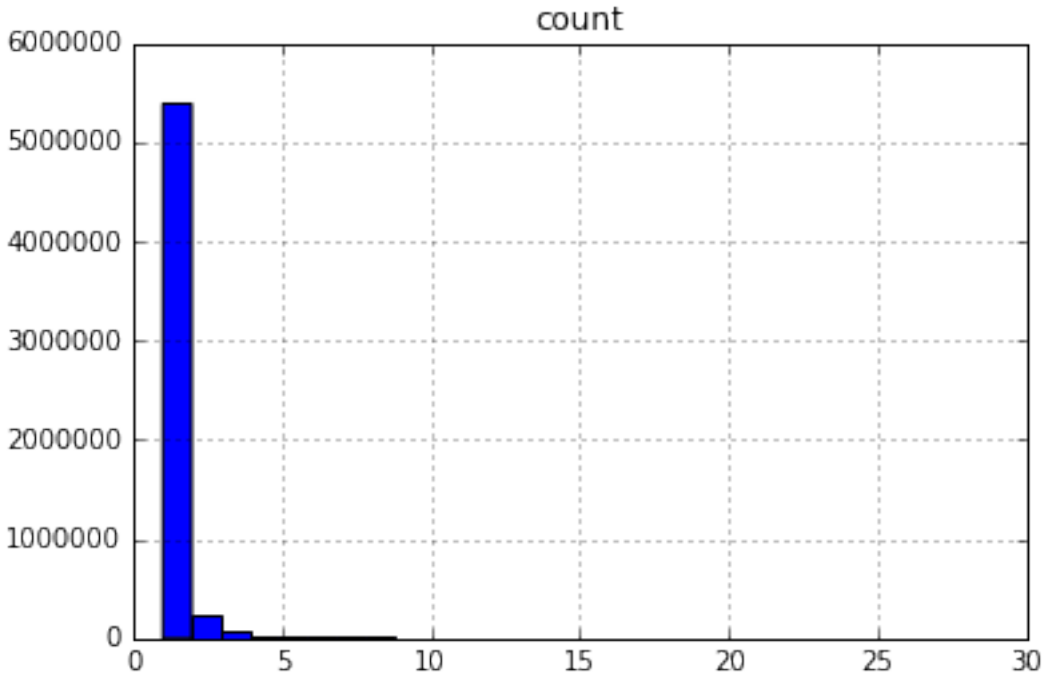
```
In [12]: titles_train_df = pd.read_csv('kaggle_data/title_unify_train', header=None, delimiter='\t')
         titles_train_df.columns = ['id', 'title', 'count']

In [13]: titles_train_df.head()
```

Out[13]:

	id	title	count
0	000000014B6D41C13D777E8314725401	коляна лента прикол	1
1	0000000150707ACB8A82451C0307BC01	candi410 rambler ru входящая рамблер-почта	1
2	0000000150707ACB8A82451C0307BC01	cosmopolitan витамин волос для женщина журнал ...	1
3	0000000150707ACB8A82451C0307BC01	realbox бокс для интернет-магазин страница тов...	1
4	0000000150707ACB8A82451C0307BC01	realbox бокс для интернет-магазин товар экипир...	2

In [14]:
```
titles_train_df.hist(column="count", bins=30)
```

Out[14]:
```
array([[<matplotlib.axes._subplots.AxesSubplot object at 0x7fa89ef113d0>]], dtype=object)
```



Here too

In [15]:
```
titles_train_df = pd.DataFrame(titles_train_df.groupby('id')['title'].apply(lambda x: x.tolist(
titles_train_df['id'] = titles_train_df.index
titles_train_df.index = range(len(titles_train_df))
titles_train_df.columns = ['titles', 'id']
```

In [16]:
```
titles_train_df.head()
```

Out[16]:

	titles	\
0	[бесплатный надёжный почта рамблер электронный...	

```
1  [24-х 34-х до договор неделя новость предложит...
2  [авто бош контакт королёв сервис, авто бош кор...
3  [ua втрать війни донбасі за на новини озвучить...
4  [black walnut грецкий орех чёрный, inmoment ru...

                                  id
0  000000013CB5719C0000A2C90002C101
1  00000001442BE24000001B7D00F50801
2  00000001448580F800003F1B31FB0901
3  0000000145BDB2FF000157971645E901
4  000000014602771F0000DB9359714C01
```

1.2.4 merge it all!

```
In [17]: # train_df = urls_train_df.merge(age_train_df, on='id', how='left') # to start, we'll consider
         # train_df = titles_train_df.merge(age_train_df, on='id', how='left')
         train_df = urls_train_df.merge(titles_train_df.merge(age_train_df, on='id', how='inner'), on='i
```

```
In [18]: train_df.head()
```

```
Out[18]:                                                urls  \
         0            [id.rambler.ru, mail.rambler.ru, r0.ru]
         1  [1prime.ru, autorambler.ru, chellak.ru, docs.c...
         2                               [bosch-korolev.ru]
         3  [aptekanizkihcen.ua, colady.ru, gorod.dp.ua, i...
         4  [astrorok.ru, diets.ru, edaplus.info, eshzdoro...

                                  id  \
         0  000000013CB5719C0000A2C90002C101
         1  00000001442BE24000001B7D00F50801
         2  00000001448580F800003F1B31FB0901
         3  0000000145BDB2FF000157971645E901
         4  000000014602771F0000DB9359714C01

                                               titles  age
         0  [бесплатный надёжный почта рамблер электронный...   53
         1  [24-х 34-х до договор неделя новость предложит...   48
         2  [авто бош контакт королёв сервис, авто бош кор...   28
         3  [ua втрать війни донбасі за на новини озвучить...   44
         4  [black walnut грецкий орех чёрный, inmoment ru...   48
```

```
In [19]: print(train_df.shape)
         print(type(train_df))
```

```
(114090, 4)
<class 'pandas.core.frame.DataFrame'>
```

1.3 Relabeling data into groups

```
In [20]: train_df['label'] = pd.Series(np.zeros(train_df.shape[0]), index=train_df.index)
```

```
In [21]: X_train, y_train = train_df.loc[:, ["urls", "titles"]], train_df.loc[:, ["age"]]
```

```
In [22]: print(y_train)
         print(type(y_train))
```

```
age
0         53
1         48
2         28
3         44
4         48
5         36
6         33
7         41
8         51
9         32
10        29
11        36
12        35
13        37
14        37
15        31
16        34
17        24
18        34
19        19
20        40
21        39
22        43
23        27
24        26
25        33
26        51
27        48
28        23
29        27
...      ...
114060    44
114061    44
114062    39
114063    42
114064    39
114065     6
114066    50
114067    30
114068    40
114069    29
114070    29
114071    31
114072    33
114073    26
114074    20
114075    24
114076    51
114077    28
114078    41
114079    46
114080    58
114081    31
```

```
114082   32
114083   44
114084   34
114085   34
114086   26
114087   58
114088   56
114089   31

[114090 rows x 1 columns]
<class 'pandas.core.frame.DataFrame'>
```

In [23]:
```
train_df.set_value(y_train[y_train["age"].isin(range(0, 27))].index, "label", 1)
train_df.set_value(y_train[y_train["age"].isin(range(27, 31))].index, "label", 2)
train_df.set_value(y_train[y_train["age"].isin(range(31, 35))].index, "label", 3)
train_df.set_value(y_train[y_train["age"].isin(range(35, 41))].index, "label", 4)
train_df.set_value(y_train[y_train["age"].isin(range(41, 49))].index, "label", 5)
train_df.set_value(y_train[y_train["age"] > 49].index, "label", 6)
```

Out[23]:

	urls
0	[id.rambler.ru, mail.rambler.ru, r0.ru]
1	[1prime.ru, autorambler.ru, chellak.ru, docs.c...
2	[bosch-korolev.ru]
3	[aptekanizkihcen.ua, colady.ru, gorod.dp.ua, i...
4	[astrorok.ru, diets.ru, edaplus.info, eshzdoro...
5	[agroru.com, donupak.ru, meteonova.ru, rus-map...
6	[9tv.co.il, katalogturbaz.ru, nanotara.ru, ofi...
7	[arbitr.ru, consultant.ru, utexco.ru]
8	[1tv.ru, arhinovosti.ru, bilettorg.ru, dev.diz...
9	[bibliotekar.ru, casinoaltair.com, expertiza34...
10	[bettingexpert.com, carexpo.ru, chelsea-fc.ru,...
11	[aif.ru, business-lady.com, calorizator.ru, ce...
12	[1prime.ru, dr-krupnik.ru, mirtesen.ru, ria.ru...
13	[championat.com]
14	[7fon.ru, ru.motorsport.com]
15	[voronezh.hh.ru]
16	[afisha.ru, altagro22.ru, asfera.info, barnaul...
17	[agbz.ru, atmagro.ru, moyareklama.by]
18	[1245.ru, aeroflot.ru, afy.ru, akwarium.su, aq...
19	[picantecooking.com, povarenok.ru, rabbit.ru, ...
20	[inf.reshuoge.ru, reshuoge.ru]
21	[kommersant.ru, m.aeroflot.ru, m.gq.ru, miss-t...
22	[aptekamos.ru, assessor.ru, buhonline.ru, drom...
23	[yota.ru]
24	[altstu.ru]
25	[tvrain.ru]
26	[takabo.ru, vladkomfort.com]
27	[proshkolu.ru, studresearsh.ru, vashaspina.ru]
28	[bytovaya-tekhnika.taganrog.mnogonado.net, tag...
29	[inf.reshuege.ru, vm.ru, whoswhos.org]
...	...
114060	[kirby-russia.tiu.ru, mail.rambler.ru, pevp.mo...
114061	[mail.rambler.ru, news.rambler.ru, online777sl...
114062	[1gb.ru, airsteps.ru, eda.ru, gotovim.ru, job-...
114063	[i.ovkuse.ru, mail.rambler.ru, osd.ru, pevp.mo...

```
114064  [aif.ru, finance.rambler.ru, mail.rambler.ru, ...
114065  [gotovim-doma.ru, id.rambler.ru, mail.rambler...
114066  [dachadecor.ru, humangarden.ru, l2announce.ru,...
114067  [chadorado.ru, family.rambler.ru, krasnodar.ir...
114068    [baby-52.ru, fundamenti-v-spb.ru, pesnifilm.ru]
114069  [formulalubvi.com, kino-teatr.ru, mail.rambler...
114070  [ac-mzsa.ru, autodoc.ru, avtotsentry.smolensk...
114071                       [allnum.ru, mail.rambler.ru]
114072  [drivers.mydiv.net, drp.su, hotdownloads.ru, m...
114073  [almaxavto.com.ua, avtorynok.com.ua, exist.ru,...
114074  [babylonvape.ru, gdzometr.ru, id.rambler.ru, m...
114075          [mail.rambler.ru, noffery.ru, rambler.ru]
114076  [forum.ribca.net, rubo.ru, top.mail.ru, top100...
114077  [consultant.ru, estatelatvia.com, god-2016.com...
114078  [bagforlife.ru, bellore.ru, blondie.ru, brjune...
114079  [autoplus.su, euroauto.ru, xn--2111-43da1a8c.x...
114080            [kino-teatr.ru, knijky.ru, zak.depo.ua]
114081  [m.rambler.ru, m.rns.online, m.weekend.rambler...
114082                  [mail-pda.rambler.ru, pozdrav.ru]
114083                                        [lady74.ru]
114084  [art-med.ru, khablyalya.ru, pregnant-club.ru, ...
114085  [afisha.ru, exist.ru, inmoment.ru, kiev.vgorod...
114086  [cosmo.ru, pda.ura.ru, podoprigora74.livejourn...
114087  [aif.ru, autochel.ru, chelyabinsk.ru, id.rambl...
114088              [blog.partisani.ge, li.ru, tvrain.ru]
114089  [doctorkirov.ru, drive.ru, extrim-park43.ru, m...

                                      id  \
0       000000013CB5719C0000A2C90002C101
1       00000001442BE24000001B7D00F50801
2       00000001448580F800003F1B31FB0901
3       0000000145BDB2FF000157971645E901
4       000000014602771F0000DB9359714C01
5       0000000147B2D6F311DB5C4201B7FB01
6       0000000147C68954150168D701A8B801
7       0000000147EB76D738CD80750C879701
8       00000001482AAFB69FA5228008AC2A01
9       0000000148390BB56A6B22BB178D3901
10      00000001487DAF8D69CD43E416D6AD01
11      0000000148AC192341E3BDAD0B95DE01
12      0000000148C2B61B70F8651309287201
13      0000000148DB999352D9CAF309511101
14      0000000148EB77A2435E76A711E38B01
15      0000000148F786727740B36700E82101
16      0000000148FD5BA0B553811202AB4601
17      0000000148FF6AFB2A2183F50311A901
18      00000001490B9AF7989CCE8E026C2901
19      00000001490C03CD81C3F77D03591501
20      000000014917C91E88E90DC90A3B9D01
21      0000001491955518 8E90DC90FF8E401
22      00000001491A89428561 0DC813C50E01
23      000000014929641F66F6C0110E62C401
24      000000014947912B3F44BA2C032EB701
25      00000001494E5DE0241913C9184EBF01
```

26	00000001494FA563241913C91D6FA201
27	00000001495B2BC55C0079021B37AD01
28	00000001496318EB3F650D200EB00101
29	00000001496C4195AD4C674803C9B501
...	...
114060	06393B87570260BC00000543F588B501
114061	06393B87570270DC00006DAC93F56F01
114062	06393B8757027C0C00006DE991DD9101
114063	06393B8757029A140000053DF5A87A01
114064	06393B8757034DB20000518C41942C01
114065	06393B875705208E0000054E2A852C01
114066	06393BC156F59CF100000533F051B801
114067	06393C8956F8F1B600006DBECC09A701
114068	06393D0556F3D2510000053CCC1E9501
114069	06393D555700FE8A00006E0D72071401
114070	06393DC056F6C1CC000004E57E05A001
114071	06393E1656FF91D4000051CCF0D13C01
114072	06393E7D56F67A6F000005489A814C01
114073	06393FF656FB75860000054780A8EE01
114074	0639434C56FE7EBA000005403C284E01
114075	0639452A5705042000000055527E56F01
114076	063945575700ED290000054ECF865F01
114077	6B9BD7D94D7F3767F8CD695D77FA6E77
114078	77F8BDE14D77F99E9E6773F7EB9EE7E1
114079	795E97E747DB38B7B9C67F9A771F8E77
114080	7DF577774777BDFDBA7EEF9F77F6C6D7
114081	D374CD7D56E7359F5B7E6F9F7B3D6DDF
114082	D79FBED94DD35ED75AADEDBC6D1CAFE7
114083	DB399FE347F77B6DB566DB1FD136DF73
114084	E736D7D34EF71D8FB46D6BDD6F9CADD1
114085	E7DB77EB4EEFF6C73DD7EB5CD37CB7E3
114086	E7DFB7DB4DDF3C7E785EDF7DF17AA76D
114087	E9B9D7D347EB1ACE5AAEEBFCE3FBCE7B
114088	EBFAC66B4EE3FB96BA5DD7DDE3787FE7
114089	F537AD6B46D31ABFF597EFDFE1BDDE71

	titles	age	label
0	[бесплатный надёжный почта рамблер электронный...	53	6
1	[24-х 34-х до договор неделя новость предложит...	48	5
2	[авто бош контакт королёв сервис, авто бош кор...	28	2
3	[ua втрать вiйни донбасi за на новини озвучить...	44	5
4	[black walnut грецкий орех чёрный, inmoment ru...	48	5
5	[апрель год день март месяц на неделя от погод...	36	4
6	[rankw ru tovar xoz xoz-tovar анализ доход ком...	33	3
7	[возможность госзакупка консультант малое плюс...	41	5
8	[13d билет заказать имя купить на спектакль те...	51	6
9	[1-метр velol александр альберто арно велоспор...	32	3
10	[1vitali1 ii мафра от порту санинга токио фк ц...	29	2
11	[caloriz ru www диета курица модный на похуден...	36	4
12	[plc powerlin адаптер, агентство азербайджан и...	35	4
13	[матч новость премьер-лига расписание россия р...	37	4
14	[window бесплатно для живой на обои рабочий ск...	37	4
15	[оплата услуга]	31	3
16	[png psd бесплатно для клипарта мультяшки на п...	34	3

```
17       [by moyareklama гомель купить на овца, агропро...   24      1
18       [06м 10x1 art daphn винить каталог коллекция к...   34      3
19       [chow mein picantecook китайский курица лапша ...   19      1
20       [гущина дмитрия задание информатика обучать ог...   40      4
21       [gq администратор блог ваш журнал лори любимый...   39      4
22       [2мл 50мг для инъекция купить мл раствор трама...   43      5
23       [4g yota безлимитный весь интернет мобильный м...   27      2
24       [алтгту, алтгту занятие расписание, алтгту отд...   26      1
25       [авторизация дождь телеканал, восстановление д...   33      3
26       [dr mosreg ru takabo uslugi zdrav врач запись ...   51      6
27       [almag алмаг аппарат инструкция отзыв по приме...   48      5
28       [вред её капуста ламинария морской она польза ...   23      1
29       [вечерний жизнь зощенко из интересный михаил м...   27      2
...                                                    ...   ...    ...
114060   [footer, ridzip ru аксессуар бытовой запчасть ...   44      5
114061   [columbu автомат игровой онлайн отправляться п...   44      5
114062   [air airstep id nike yeezi каталог кроссовок ч...   39      4
114063   [footer, prodport ru консервы масло молоко мяс...   42      5
114064   [footer, аргумент биатлонист бьорндален вид до...   39      4
114065   [co rambler авторизация, footer, бесплатный на...    6      1
114066   [adrenalin bot l2walker lineag top, battlestar...   50      6
114067   [a7398h plai-doh принцесса софия церемония чай...   30      2
114068   [беременность весь для заботливый мама она пор...   40      4
114069   [hello russia встретить метро можно москва сно...   29      2
114070   [2746agamvw1b guardian автодок интернет-магази...   29      2
114071   [lenta ru sivkova-elena входящая рамблер-почта...   31      3
114072   [crystal player pro window без бесплатно для к...   33      3
114073   [4a91 lancer mitsubishi для запчасть седан, al...   26      1
114074   [co rambler авторизация, nation ohm rda turbo ...   20      1
114075   [nofferi автомобиль автомобильный жидкий новый...   24      1
114076   [весь добавить каталог рейтинг сайт скриншот с...   51      6
114077   [glamour ru базовый весна-лето гардероб как но...   28      2
114078   [698308-fondent airstep cotto kultpokupki ru s...   41      5
114079   [autoplu frontera opel su автомобиль автомобил...   46      5
114080   [depo буковель красуня купальник на новини при...   58      6
114081   [new rambler servic бензин биржевой более выра...   31      3
114082                                       [март стих]   32      3
114083   [апрель год гороскоп для женский на телец, вес...   44      5
114084   [арт-мёд беременность ветрянка вопрос на ответ...   34      3
114085   [chanel афиша киев магазин, cl merced для зажи...   34      3
114086   [cosmopolitan ведущий гламурный гузеев журнал ...   26      1
114087   [4z7616051a 4z7616051d 4z7616052a allroad arno...   58      6
114088   [ru онлайн попка порно секс, азербайджан военн...   56      6
114089   [5d merced-benz видео драйв комплектация отзыв...   31      3

[114090 rows x 5 columns]
```

In [ ]:
```
train_df.set_value(y_train[y_train["age"].isin(range(0, 10))].index, "label", 1)
train_df.set_value(y_train[y_train["age"].isin(range(10, 20))].index, "label", 2)
train_df.set_value(y_train[y_train["age"].isin(range(20, 30))].index, "label", 3)
train_df.set_value(y_train[y_train["age"].isin(range(30, 40))].index, "label", 4)
train_df.set_value(y_train[y_train["age"].isin(range(40, 50))].index, "label", 5)
train_df.set_value(y_train[y_train["age"].isin(range(50, 60))].index, "label", 6)
train_df.set_value(y_train[y_train["age"].isin(range(60, 70))].index, "label", 7)
```

```
train_df.set_value(y_train[y_train["age"].isin(range(70, 80))].index, "label", 8)
train_df.set_value(y_train[y_train["age"].isin(range(80, 90))].index, "label", 9)
train_df.set_value(y_train[y_train["age"].isin(range(90, 100))].index, "label", 10)
```

In [24]:
```
print(train_df[train_df["label"] == 0].shape)
```

```
(1863, 5)
```

1.4 Getting test data

1.4.1 urls

In [25]:
```
urls_test_df = pd.read_csv('kaggle_data/url_domain_test', header=None, delimiter='\t')
urls_test_df.columns = ['id', 'url', 'count']
urls_test_df = urls_test_df[['id', 'url']]
```

In [26]:
```
urls_test_df = pd.DataFrame(urls_test_df.groupby('id')['url'].apply(lambda x: x.tolist()))
urls_test_df['id'] = urls_test_df.index
urls_test_df.index = range(len(urls_test_df))
urls_test_df.columns = ['urls', 'id']
```

In [27]:
```
urls_test_df.head()
```

Out[27]:

	urls	id
0	[1000bankov.ru, 1tv.ru, 4put.ru, argumenti.ru,...	000000014A02348E701552980349FF01
1	[autorambler.ru, bilettorg.ru, dsol-druzhba.ru...	000000014A10EA183BF8594A0B2AB201
2	[photosight.ru, rambler.ru]	000000014A4FE5C33A929D4C26943601
3	[base.consultant.ru, dogovor-obrazets.ru, fd.r...	000000014B7BB9957784A9BC0AC9F401
4	[assessor.ru, audit-it.ru, base.garant.ru, com...	000000014C7749F896D82C2B01E8B801

1.4.2 titles

In [28]:
```
titles_test_df = pd.read_csv('kaggle_data/title_unify_train', header=None, delimiter='\t')
titles_test_df.columns = ['id', 'title', 'count']
titles_test_df = titles_test_df[['id', 'title']]
```

In [29]:
```
titles_test_df = pd.DataFrame(titles_test_df.groupby('id')['title'].apply(lambda x: x.tolist())
titles_test_df['id'] = titles_test_df.index
titles_test_df.index = range(len(titles_test_df))
titles_test_df.columns = ['titles', 'id']
```

In [30]:
```
titles_test_df.head()
```

Out[30]:

	titles	id
0	[бесплатный надёжный почта рамблер электронный...	000000014A02348E701552980349FF01
1	[24-х 34-х до договор неделя новость предложит...	000000014A10EA183BF8594A0B2AB201
2	[авто бош контакт королёв сервис, авто бош кор...	000000014A4FE5C33A929D4C26943601
3	[ua втрать війни донбасі за на новини озвучить...	000000014B7BB9957784A9BC0AC9F401
4	[black walnut грецкий орех чёрный, inmoment ru...	000000014C7749F896D82C2B01E8B801

```
                                      id
0  000000013CB5719C0000A2C90002C101
1  00000001442BE24000001B7D00F50801
2  00000001448580F800003F1B31FB0901
3  0000000145BDB2FF000157971645E901
4  000000014602771F0000DB9359714C01
```

1.4.3 merge

```
In [31]: X_urls_test = urls_test_df.urls.values
         X_urls_test = map(lambda x: ' '.join(x), X_urls_test)
         %time hw = HashingVectorizer(n_features=1000).fit(X_urls_test)
         X_urls_test = hw.transform(X_urls_test).todense()
```

```
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 41 μs
```

```
In [32]: print(X_urls_test.shape)
```

```
(19974, 1000)
```

1.5 1st level – Linear Regression with 6 equal-sized buckets

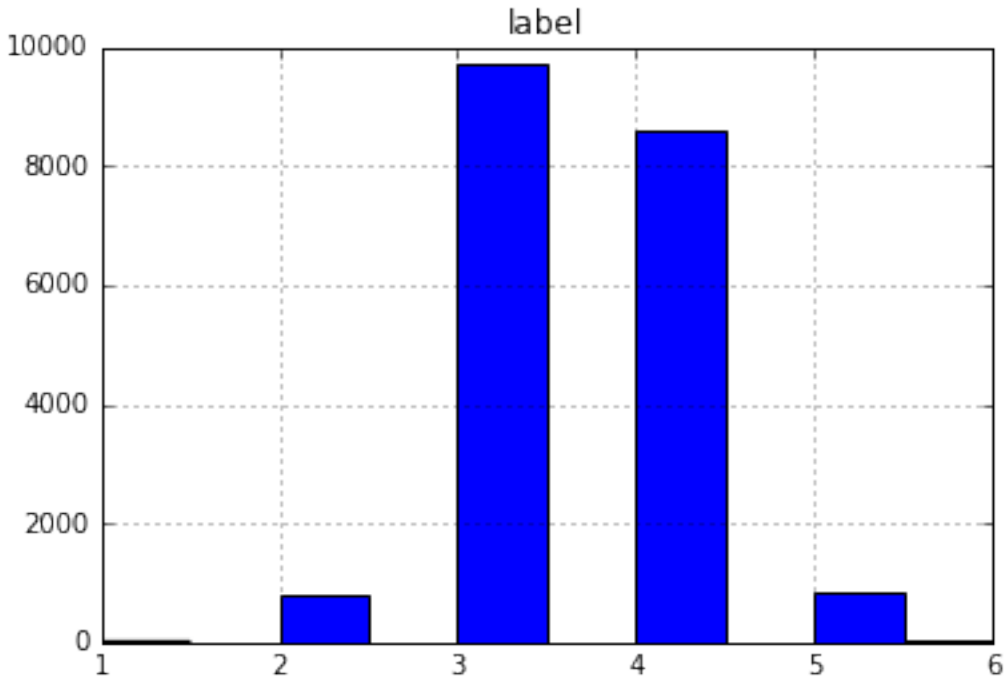
```
In [33]: X_train_matrix, y_train = train_df.loc[:, ["urls"]].as_matrix(), train_df.loc[:, ["label"]]
         # print(X_train)
         # X_train_matrix, y_train = train_df.loc[:, ["urls", "titles"]].as_matrix(), train_df.loc[:, [
         X_train = train_df.urls.values
         for index in range(X_train_matrix.shape[0]):
             X_train[index] = map(lambda x: ' '.join(x), X_train_matrix[index])[0]
         print(X_train)
         print(X_train.shape)
         print(type(X_train))
         %time hw = HashingVectorizer(n_features=1000)
         %time X_train = hw.transform(X_train)
         print(X_train.shape)
         print(y_train.shape)
```

```
['id.rambler.ru mail.rambler.ru r0.ru'
 '1prime.ru autorambler.ru chellak.ru docs.cntd.ru echo.msk.ru expert.ru finance.rambler.ru forbes.ru fo
 'bosch-korolev.ru' ...,
 'aif.ru autochel.ru chelyabinsk.ru id.rambler.ru khl.ru m.rambler.ru m.rsport.ru mail-pda.rambler.ru ne
 'blog.partisani.ge li.ru tvrain.ru'
 'doctorkirov.ru drive.ru extrim-park43.ru m.regions.pulset.ru mail-pda.rambler.ru reso.ru sberbank.ru']
(114090,)
<type 'numpy.ndarray'>
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 15 μs
CPU times: user 6.4 s, sys: 40 ms, total: 6.44 s
Wall time: 6.39 s
(114090, 1000)
(114090, 1)
```

```
In [34]: reg = LinearRegression()
         %time reg.fit(X_train, y_train)
```

```
CPU times: user 1.17 s, sys: 10 ms, total: 1.18 s
Wall time: 1.17 s
```

```
Out[34]: LinearRegression(copy_X=True, fit_intercept=True, n_jobs=1, normalize=False)

In [35]: y_pred_labels = reg.predict(X_urls_test).round()

In [36]: print(y_pred_labels)

[[ 4.]
 [ 4.]
 [ 4.]
 ...,
 [ 3.]
 [ 3.]
 [ 5.]]

In [37]: urls_test_df["label"] = y_pred_labels

In [38]: urls_test_df.hist(column="label")

Out[38]: array([[<matplotlib.axes._subplots.AxesSubplot object at 0x7fa88d1e7d10>]], dtype=object)
```




```
In [39]: urls_test_df['age'] = pd.Series(np.zeros(urls_test_df.shape[0]), index=urls_test_df.index)

In [40]: print(urls_test_df)

urls  \
0      [1000bankov.ru, 1tv.ru, 4put.ru, argumenti.ru,...
1      [autorambler.ru, bilettorg.ru, dsol-druzhba.ru...
2                              [photosight.ru, rambler.ru]
3      [base.consultant.ru, dogovor-obrazets.ru, fd.r...
4      [assessor.ru, audit-it.ru, base.garant.ru, com...
```

```
5      [mail.rambler.ru, mineral.ru, nalog.ru, nova.r...
6      [101.ru, 2do2go.ru, antikvar.su, antikvariat74...
7      [63.ru, deti-club.ru, k-agent.ru, mail.rambler...
8      [1tv.ru, aif.ru, allbanks.kz, auto.drom.ru, au...
9      [auto.drom.ru, autorambler.ru, drom.ru, eva.ru...
10     [6lib.ru, belogorck.ru, gazeta.ru, informio.ru...
11                         [mail.rambler.ru, rambler.ru]
12     [aeroflot.ru, id.rambler.ru, mail.rambler.ru, ...
13     [1000-tovarov.ru, a-lpha.su, aenergetika.ru, a...
14     [3-ndfl.com, citytraffic.ru, consultant.ru, co...
15     [dni.ru, games.kanobu.ru, goodhouse.ru, id.ram...
16     [98765432.www.nn.ru, aimfar.solution.weborama...
17     [autorambler.ru, blagoveschensk.tiu.ru, consul...
18     [1zagran.ru, 3-ndfl.info, 52.mobilfunk.ru, aif...
19     [agroserver.ru, aimfar.solution.weborama.fr, a...
20     [1prime.ru, afisha.ru, aif.ru, autorambler.ru,...
21     [buhonline.ru, hi-tech.mail.ru, id.rambler.ru,...
22     [a-russia.ru, adenium-doma.ru, autorambler.ru,...
23     [apteka-ifk.ru, by-vitebsk.vedun.ru, id.ramble...
24     [1tv.ru, 8634city.ru, aif.ru, allforchildren.r...
25     [3dyuriki.com, mail.rambler.ru, news.rambler.r...
26     [autorambler.ru, dni.ru, govoritmoskva.ru, inf...
27     [2r.ru, 30.joblib.ru, abc2home.ru, aif-nn.ru, ...
28     [17mkad.ru, 360.ru, artstudio-pro.ru, autoramb...
29     [eadaily.com, family.rambler.ru, inforeactor.r...
...                                                  ...
19944  [50.rjob.ru, 93job.ru, angarsk.careerist.ru, a...
19945  [100bretelek.ru, 1zagran.ru, 2r.ru, 2trade.ru,...
19946  [mail.rambler.ru, new.tv.rambler.ru, onlyf.ws,...
19947  [avtolitsey.jimdo.com, ax.do.am, bmwpark.ru, b...
19948  [amgum.ru, assets.adobedtm.com, avtovzglyad.ru...
19949  [1bm.ru, 1lentanovostei.ru, 1prime.ru, 1svegie...
19950  [220blog.ru, aimfar.solution.weborama.fr, cons...
19951                      [mail.rambler.ru, rambler.ru]
19952  [aliexpreses.ru, autorambler.ru, badm11.ru, ba...
19953  [id.rambler.ru, iland.tv, mail.rambler.ru, new...
19954  [job-mo.ru, lesorub44.ru, mail.rambler.ru, mo...
19955  [aimfar.solution.weborama.fr, id.rambler.ru, m...
19956  [football.ua, m.rambler.ru, mail-pda.rambler.r...
19957  [altshuller.ru, blog.liga.net, felbert.livejou...
19958  [babyplan.ru, kidbutik.com.ua, led-portal.ru, ...
19959  [advego.com, advego.ru, ax.do.am, clubmed.ru, ...
19960                     [futuland.ru, mail.rambler.ru]
19961  [aif.ru, baruk.ru, expert.ru, id.rambler.ru, l...
19962  [id.rambler.ru, ilizium.com, mail.rambler.ru, ...
19963  [mail.rambler.ru, n-novgorod.carent.ru, paulz...
19964  [football-shopping.ru, forum.sibnet.ru, id.ram...
19965  [career.ru, hh.ru, itkani.ru, mail.rambler.ru,...
19966  [autorambler.ru, gazeta.ru, iarex.ru, jobinmos...
19967  [mail.rambler.ru, ped-kopilka.ru, pevp.mos.ru,...
19968  [glavap.ru, irn.ru, mail.rambler.ru, medikforu...
19969  [akusherstvo.ru, cat.pet2me.com, domik3.ru, ge...
19970  [free.drweb.ru, mail.rambler.ru, pevp.mos.ru, ...
19971  [afisha.ru, formulalubvi.com, gazeta.ru, hh.ru...
```

```
19972  [kirrail.org, lenta.ru, mail.rambler.ru, rambl...
19973  [help.rambler.ru, mail.rambler.ru, realtymag.r...

                                      id  label  age
0      000000014A02348E701552980349FF01      4    0
1      000000014A10EA183BF8594A0B2AB201      4    0
2      000000014A4FE5C33A929D4C26943601      4    0
3      000000014B7BB9957784A9BC0AC9F401      3    0
4      000000014C7749F896D82C2B01E8B801      3    0
5      000000014CAD0728017D0850076EAA01      4    0
6      000000014CD445FDBD0B82AB03B7E601      4    0
7      000000014CE4E75679C7BAD4020FEA01      4    0
8      000000014D21FF3C0D847BEE0D309801      4    0
9      000000014D3ABB36375A07970C971401      4    0
10     000000014D98FA2EA7479FB404A03101      3    0
11     000000014DAE6C7C86332D11016BE501      3    0
12     000000014DD67EF27730CD2C0030D201      4    0
13     000000014DFED69C57CE072A04306601      4    0
14     000000014E083FE5A9DE149404617301      4    0
15     000000014E13029D28DD08A8000B8C01      4    0
16     000000014E130B625265210900274D01      4    0
17     000000014E1329B2352151A500573001      4    0
18     000000014E13344B1D9CD24D0015E201      3    0
19     000000014E136A81387C51A900B68B01      3    0
20     000000014E140852325D08B0008E5501      4    0
21     000000014E187ECE32BF08B102990B01      3    0
22     000000014E1C67067A1516A600059001      4    0
23     000000014E1C739F0CFD5FF60019F601      4    0
24     000000014E1C770911C947120010A201      4    0
25     000000014E1C79C5C12F05B300241B01      4    0
26     000000014E33F54E06E9DEA702372801      5    0
27     000000014E435E5B70122BC101D70901      3    0
28     000000014E4A5A8948B7064603A26401      4    0
29     000000014E4D2C4457D7F01E0246B301      5    0
...                                  ...    ...  ...
19944  0638F71656D7CEF5000004FDEEBC7A01      3    0
19945  0638FAFD56F0F2780000054F2C61F601      3    0
19946  0638FF59570537E90000054C1FA89601      4    0
19947  0639030F5702861A000004DE6E0AA401      3    0
19948  06390A7756CDFA780000054394E69501      3    0
19949  06390B8C56DB281E0000055EDF179E01      4    0
19950  0639141856F22C3B00000558462EC601      3    0
19951  0639159456FD44EB000004D600DD9301      3    0
19952  0639173F56B8D37900006DE98701AE01      3    0
19953  063921985703CA6E0000053D0D9D7901      4    0
19954  0639220256FFF8B40000054CACA14601      3    0
19955  063924CE5705583F000051CC70B91C01      3    0
19956  0639258857056E3700006DBED3C0AB01      3    0
19957  0639276956F990FE000051E07F3A3401      4    0
19958  063927B456FD3587000004DEFFA3E601      2    0
19959  06392B7156DD422C000004DE5EF5EB01      3    0
19960  06392EEA5703F65D0000054731AE6401      3    0
19961  0639303257001D0A00000533C3FBF601      4    0
19962  063936205703E1740000055FAEDB8001      3    0
```

```
19963  0639370457049F390000053324988801      3    0
19964  0639387E56F1821F0000053C9F63A701      3    0
19965  0639394D57041CC4000051B65788B001      2    0
19966  06393957570236E400000555EC4A5201      4    0
19967  06393B8756FBCE950000054B6CE7A501      4    0
19968  06393B8756FDFCB60000054B96091901      4    0
19969  06393B8756FE32ED000004DE11330B01      3    0
19970  06393B87570376CB000004D6801BDC01      4    0
19971  06393D555702D3B2000004DE774F1901      3    0
19972  06394267570511A900000540C88F9201      3    0
19973  0639457C57020EB700006DD1885A0501      5    0

[19974 rows x 4 columns]
```

1.6 2nd level – k-means with several centroids

```
In [44]: from sklearn.cluster import KMeans
         from pandas import Index

         n_of_clusters = [27, 4, 4, 6, 8, 50]
         start_age = [0, 27, 31, 35, 40, 49]

         for label in range(1, 7):
             df_subset = train_df[train_df["label"] == label]
             print(df_subset.shape)
             X_train_matrix, y_train = df_subset.loc[:, ["urls"]].as_matrix(), df_subset.loc[:, ["age"]]
         #     print(X_train_matrix)
         #     print(X_train_matrix.shape)
             # X_train_matrix, y_train = train_df.loc[:, ["urls", "titles"]].as_matrix(), train_df.loc[
             X_train = df_subset.urls.values.copy()
         #     print(X_train)
             for index in range(X_train_matrix.shape[0]):
                 X_train[index] = X_train_matrix[index][0]
         #     print(X_train)
         #     print(X_train.shape)
         #     print(type(X_train))
             test_subset = urls_test_df[urls_test_df["label"] == label].urls
             if test_subset.shape[0] == 0:
                 continue
         #     print(test_subset)
         #     print(test_subset.shape)
         #     print(test_subset.index)
             %time hw = HashingVectorizer(n_features=1000)
             %time X_train = hw.transform(X_train)
             %time km = KMeans(n_of_clusters[label - 1], n_jobs=-1)
             %time km.fit(X_train, y_train)
             test_subset_m = map(lambda x: ' '.join(x), test_subset)
             %time hw = HashingVectorizer(n_features=1000).fit(test_subset_m)
             test_subset_m = hw.transform(test_subset_m).todense()
             y_pred_subset = km.predict(test_subset_m)
         #     print(y_pred_subset)
         #     test_subset['age'] = y_pred_subset
             g = 0
             for index, data in test_subset.iteritems():
```

```
        #         print(index)
                  urls_test_df.set_value(index, "age", y_pred_subset[g] + start_age[label - 1])
                  g += 1
        #     print(urls_test_df.loc[3202, "age"])
        #     km.predict()
```

```
(22226, 5)
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 15 µs
CPU times: user 1.18 s, sys: 16.7 ms, total: 1.19 s
Wall time: 1.18 s
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 16 µs
CPU times: user 777 ms, sys: 157 ms, total: 933 ms
Wall time: 15.6 s
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 56 µs
(18969, 5)
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 13.1 µs
CPU times: user 1.07 s, sys: 6.67 ms, total: 1.08 s
Wall time: 1.05 s
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 16 µs
CPU times: user 733 ms, sys: 123 ms, total: 857 ms
Wall time: 4.77 s
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 57 µs
(17358, 5)
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 12.9 µs
CPU times: user 930 ms, sys: 10 ms, total: 940 ms
Wall time: 931 ms
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 29.1 µs
CPU times: user 723 ms, sys: 137 ms, total: 860 ms
Wall time: 3.91 s
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 179 µs
(18222, 5)
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 41 µs
CPU times: user 1.04 s, sys: 3.33 ms, total: 1.05 s
Wall time: 1.03 s
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 28.8 µs
CPU times: user 733 ms, sys: 133 ms, total: 867 ms
Wall time: 6.2 s
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 57.2 µs
(18124, 5)
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 14.1 µs
CPU times: user 1.09 s, sys: 6.67 ms, total: 1.1 s
```

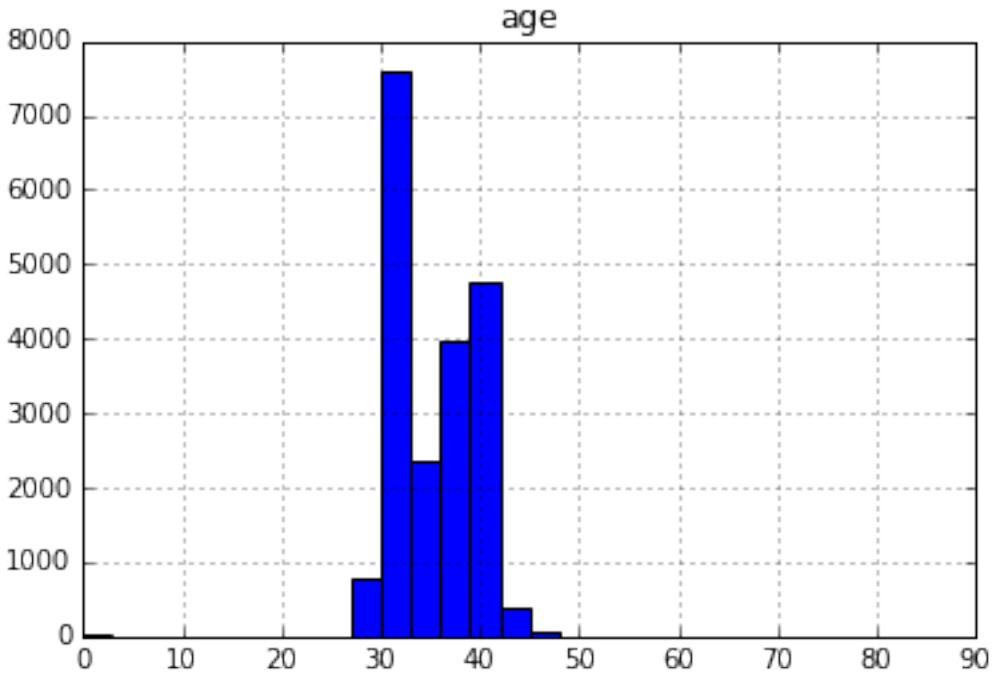
```
Wall time: 1.08 s
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 16.9 µs
CPU times: user 743 ms, sys: 137 ms, total: 880 ms
Wall time: 8.89 s
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 56 µs
(17328, 5)
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 12.9 µs
CPU times: user 1.19 s, sys: 6.67 ms, total: 1.19 s
Wall time: 1.17 s
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 27.9 µs
CPU times: user 750 ms, sys: 147 ms, total: 897 ms
Wall time: 26.8 s
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 64.8 µs
```

```
In [45]: print(urls_test_df.loc[3202, "age"])
```

```
11.0
```

```
In [50]: urls_test_df.hist(column="age", bins=30)
```

```
Out[50]: array([[<matplotlib.axes._subplots.AxesSubplot object at 0x7fa8648f9050>]], dtype=object)
```



1.7 Model testing

```
In [ ]: reg = LinearRegression(normalize=True, n_jobs=-1)
        %time cross_val_score(reg, X_train, y_train, scoring='mean_squared_error')
```

1.8 Solution Gathering

In [52]:
```
urls_test_df = urls_test_df[['id', 'age']]
urls_test_df.columns = ['Id', 'age']
```

In [53]:
```
urls_test_df.head()
```

Out[53]:

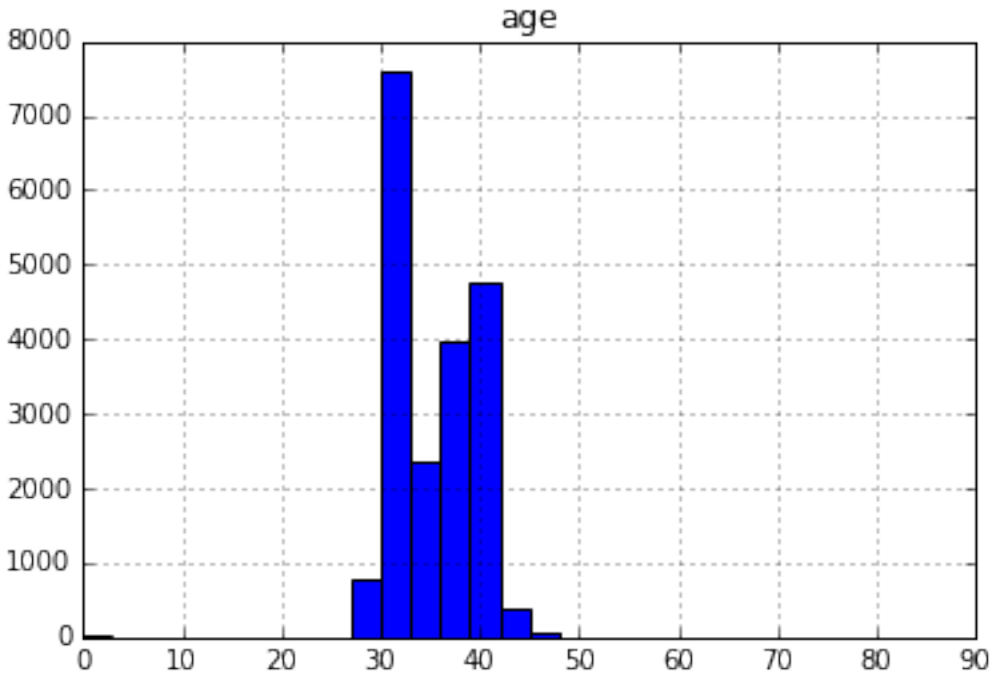
	Id	age
0	000000014A02348E701552980349FF01	39
1	000000014A10EA183BF8594A0B2AB201	36
2	000000014A4FE5C33A929D4C26943601	36
3	000000014B7BB9957784A9BC0AC9F401	34
4	000000014C7749F896D82C2B01E8B801	31

In [54]:
```
random_sol = pd.read_csv('random_solution.csv')
miss_idx = set(random_sol.Id.values) - set(urls_test_df.Id.values)
miss_df = pd.DataFrame(zip(list(miss_idx), np.ones(len(miss_idx))))
miss_df.columns = ['Id', 'age']
miss_df.age = 30
```

In [55]:
```
urls_test_df = urls_test_df.append(miss_df, ignore_index=True)
```

In [60]:
```
urls_test_df.hist(column="age", bins=30)
```

Out[60]:
```
array([[<matplotlib.axes._subplots.AxesSubplot object at 0x7fa8644a91d0>]], dtype=object)
```



In [57]:
```
urls_test_df.to_csv('solution.csv', index=False)
```

In [58]:
```
!wc -l solution.csv
```

```
19980 solution.csv
```

In [ ]:

In [59]:
```
def down_dimensions(cur_df):
    print(cur_df.shape)
    X_train_matrix, y_train = cur_df.loc[:, ["urls", "titles"]].as_matrix(), cur_df.loc[:, ["la
    X_train = cur_df.urls.values
    for index in range(X_train_matrix.shape[0]):
        tmp = map(lambda x: ' '.join(x), X_train_matrix[index])
        X_train[index] = tmp[0] + ' ' + tmp[1]
    %time hw = HashingVectorizer(n_features=1000).fit(X_train)
    %time X_train = hw.transform(X_train).todense()
    print(X_train.shape)
    print(y_train.shape)
    return X_train, y_train
```

In [ ]: