In [ ]:

```
!pip install -q kaggle
```

In [ ]:

```
from google.colab import files
uploaded = files.upload()
```

Выбрать файлы Файл не выбран

Upload widget is only available when the cell has been executed in the current browser session.
Please rerun this cell to enable.

In [ ]:

```
!mkdir /root/.kaggle
!mv kaggle.json /root/.kaggle/kaggle.json
!kaggle competitions download -c nyc-taxi-trip-duration
```

```
Warning: Your Kaggle API key is readable by other users on this system! To
fix this, you can run 'chmod 600 /root/.kaggle/kaggle.json'
Warning: Looks like you're using an outdated API Version, please consider
updating (server 1.5.6 / client 1.5.4)
Downloading sample_submission.zip to /content
  0% 0.00/2.49M [00:00<?, ?B/s]
100% 2.49M/2.49M [00:00<00:00, 83.0MB/s]
Downloading test.zip to /content
 25% 5.00M/20.3M [00:00<00:02, 5.43MB/s]
100% 20.3M/20.3M [00:01<00:00, 20.0MB/s]
Downloading train.zip to /content
 91% 57.0M/62.9M [00:01<00:00, 27.6MB/s]
100% 62.9M/62.9M [00:01<00:00, 59.1MB/s]
```

In [ ]:

```
import pandas as pd
import seaborn as sns
from matplotlib import pyplot as plt

df = pd.read_csv('train.zip', compression='zip', header=0, sep=',', quotechar='"')
```

In [ ]:

```
df.head()
```

Out[ ]:

	id	vendor_id	pickup_datetime	dropoff_datetime	passenger_count	pickup_longitu
0	id2875421	2	2016-03-14 17:24:55	2016-03-14 17:32:30	1	-73.9821
1	id2377394	1	2016-06-12 00:43:35	2016-06-12 00:54:38	1	-73.9804
2	id3858529	2	2016-01-19 11:35:24	2016-01-19 12:10:48	1	-73.9790
3	id3504673	2	2016-04-06 19:32:31	2016-04-06 19:39:40	1	-74.0100
4	id2181028	2	2016-03-26 13:30:55	2016-03-26 13:38:10	1	-73.9730

In [ ]:

```
df = df.drop('dropoff_datetime', axis=1)
```

In [ ]:

```
df = df.sort_values(by='pickup_datetime')
```

In [ ]:

```
train_df, test_df = df[:10 ** 6], df[10 ** 6:]
```

In [ ]:

```
len(test_df)
```

Out[ ]:

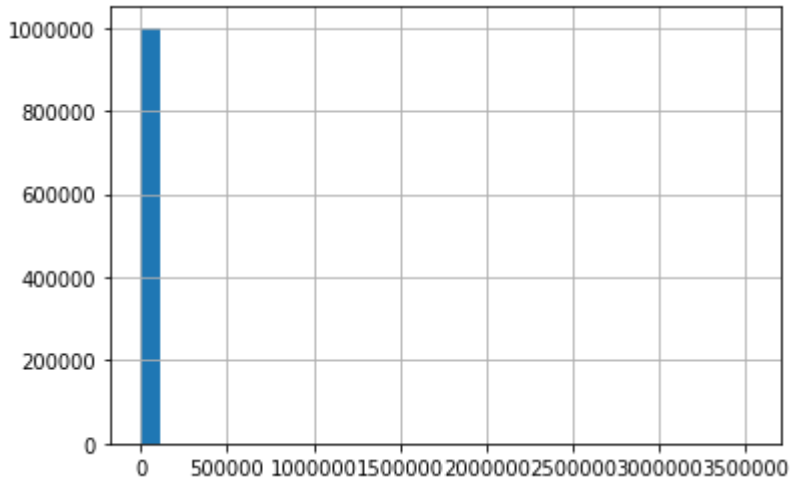
```
458644
```

In [ ]:

```
train_df.trip_duration.hist(bins=30)
```

Out[ ]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f8e3e59d9e8>
```



In [ ]:

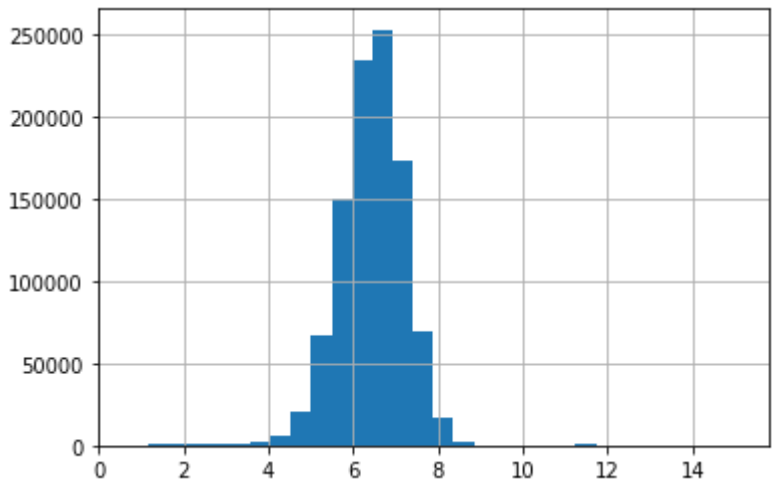
```
import numpy as np
```

In [ ]:

```
np.log1p(train_df.trip_duration).hist(bins=30)
```

Out[ ]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f8e3e4e46a0>
```



In [ ]:

```
train_df['log_trip_duration'] = np.log1p(train_df.trip_duration)
test_df['log_trip_duration'] = np.log1p(test_df.trip_duration)
```

```
/usr/local/lib/python3.6/dist-packages/ipykernel_launcher.py:1: SettingWit
hCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-doc
s/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  """Entry point for launching an IPython kernel.
/usr/local/lib/python3.6/dist-packages/ipykernel_launcher.py:2: SettingWit
hCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-doc
s/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

In [ ]:

```
train_df.pickup_datetime = pd.to_datetime(train_df.pickup_datetime)
test_df.pickup_datetime = pd.to_datetime(test_df.pickup_datetime)
```

```
/usr/local/lib/python3.6/dist-packages/pandas/core/generic.py:5208: Settin
gWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-doc
s/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  self[name] = value
```

In [ ]:

```
dates = train_df.pickup_datetime.apply(lambda x: x.date())
```

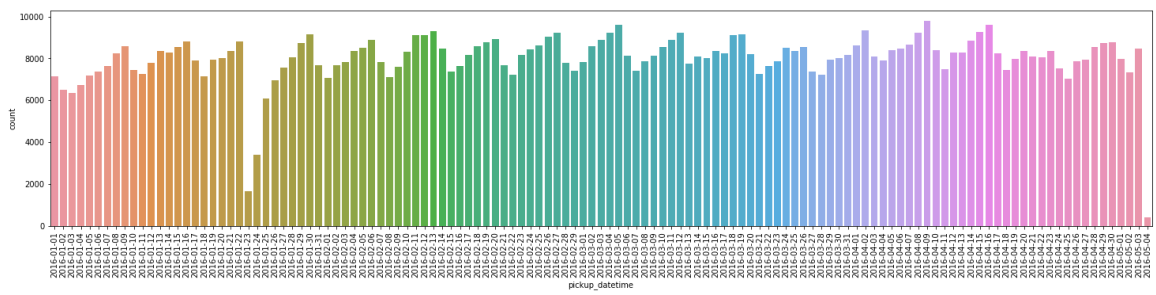
In [ ]:

```

```

In [ ]:

```
plt.figure(figsize=(25, 5))
data_count_plot = sns.countplot(x = dates)
data_count_plot.set_xticklabels(data_count_plot.get_xticklabels(), rotation=90);
```



In [ ]:

```
group_by_date = train_df.groupby(dates)
```

In [ ]:

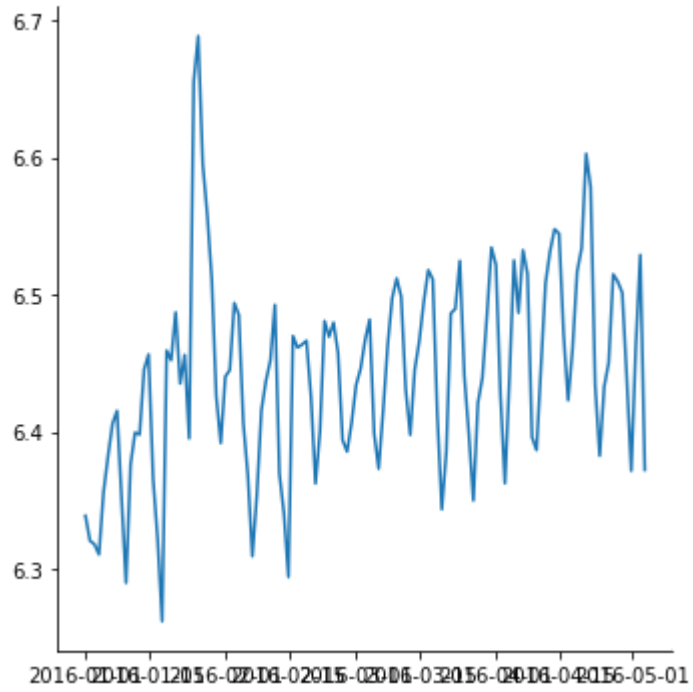
```
sns.relplot(data=group_by_date.log_trip_duration.aggregate('mean'), kind='line')
```

```
/usr/local/lib/python3.6/dist-packages/pandas/plotting/_matplotlib/convert
er.py:103: FutureWarning: Using an implicitly registered datetime converte
r for a matplotlib plotting method. The converter was registered by pandas
on import. Future versions of pandas will require you to explicitly regist
er matplotlib converters.

To register the converters:
        >>> from pandas.plotting import register_matplotlib_converters
        >>> register_matplotlib_converters()
  warnings.warn(msg, FutureWarning)
```

Out[ ]:

```
<seaborn.axisgrid.FacetGrid at 0x7f8e3b2d5470>
```



In [ ]:

```
def create_features(data_frame):
    X = pd.concat(
        [
         data_frame.pickup_datetime.apply(lambda x: x.timetuple().tm_yday),
         data_frame.pickup_datetime.apply(lambda x: x.hour)
        ], axis=1, keys=['day', 'hour']
    )
    return X, data_frame.log_trip_duration
```

In [ ]:

```
X_train, y_train = create_features(train_df)
X_test, y_test = create_features(test_df)
```

In [ ]:

```
X_train.head()
```

Out[ ]:

	day	hour
96469	1	0
223872	1	0
713067	1	0
652463	1	0
722901	1	0

In [ ]:

```
from sklearn.preprocessing import OneHotEncoder
from sklearn.compose import ColumnTransformer
```

In [ ]:

```
ohe = ColumnTransformer([("One Hot", OneHotEncoder(sparse=False), [1])], remainder='pas
sthrough')
```

In [ ]:

```
X_train = ohe.fit_transform(X_train)
```

In [ ]:

```
X_train.shape
```

Out[ ]:

```
(1000000, 25)
```

In [ ]:

```
X_test = ohe.transform(X_test)
```

In [ ]:

```
from sklearn.linear_model import Ridge
```

In [ ]:

```
from sklearn.metrics import mean_squared_error
```

In [ ]:

```
ridge = Ridge(alpha=1000).fit(X_train, y_train)
```

In [ ]:

```
mean_squared_error(y_test, ridge.predict(X_test))
```

Out[ ]:

```
0.6539687819760849
```

In [ ]:

```

```