HW1

2015 年 10 月 22 日

1 応用統計 HW1

詳細: http://www.stat.t.u-tokyo.ac.jp/~takemura/ouyoutoukei/

In [4]:
```
#-*- encoding: utf-8 -*-
'''
Ouyoutoukei HW1
'''
%matplotlib inline
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
import statsmodels.api as sm
np.set_printoptions(precision=3)
pd.set_option('display.precision', 4)
```

1.1 下準備

データのインポート, 基礎統計量の表示

In [5]:
```
# csv をインポート
df = pd.read_csv( 'odakyu-mansion.csv' )
# 基礎統計量を表示
print(df.describe())
```

```
time       bus      walk      price       area       bal     kosuu  \
count  185.000  185.000  185.000    185.000  185.000  185.000   178.000
mean    27.292    2.465    8.470   2929.730   72.682    9.620    89.449
std     14.076    5.277    5.426   2596.096   27.722    6.479   203.317
min      3.000    0.000    1.000    630.000   19.120    0.000     1.000
25%     18.000    0.000    4.000   1490.000   56.850    6.000    21.000
50%     26.000    0.000    8.000   2180.000   69.020    8.800    35.000
75%     33.000    0.000   13.000   3580.000   80.990   11.670    73.750
max     65.000   26.000   19.000  24800.000  230.720   39.670  2080.000
```

	floor	tf	year
count	185.000	185.000	185.000
mean	3.681	6.454	80.924
std	2.703	3.420	18.423
min	1.000	2.000	0.000
25%	2.000	4.000	74.000
50%	3.000	5.000	85.000
75%	5.000	8.000	92.000
max	14.000	20.000	99.000

家の向きは東, 西, 南, 北のダミー変数（0 または 1。南東の場合、南と東の両方に 1）に分解し変換して、

In [6]:
```
# サンプルサイズ
data_len = df.shape[0]

# 家の向きは dummy に
df['d_N'] = np.zeros(data_len, dtype=float)
df['d_E'] = np.zeros(data_len, dtype=float)
df['d_W'] = np.zeros(data_len, dtype=float)
df['d_S'] = np.zeros(data_len, dtype=float)
for i, row in df.iterrows():
    for direction in ["N", "W", "S", "E"]:
        if direction in str(row.muki):
            df.loc[i, 'd_{0}'.format(direction)] = 1

# 先頭 10 件を表示
print(df.head(10))
```

	time	bus	walk	price	area	bal	kosuu	floor	tf	muki	year	d_N \
0	3	0	6	1680	44.60	3.50	19	4	5	S	68	0
1	3	0	4	2280	48.87	4.05	12	2	4	S	74	0
2	3	0	7	2880	57.00	7.22	26	4	7	S	70	0
3	3	0	2	4340	55.25	7.35	44	3	6	SW	92	0
4	3	0	6	4980	88.02	8.70	30	4	8	SE	74	0
5	3	0	6	9800	121.56	6.71	30	2	3	S	83	0
6	5	0	3	1150	19.12	0.00	35	8	8	NE	70	1
7	5	0	1	3850	52.08	5.67	21	5	9	S	98	0
8	5	0	9	7580	78.60	14.10	68	3	4	W	99	0
9	5	0	6	11870	123.29	14.14	26	2	6	E	98	0

	d_E	d_W	d_S
0	0	0	1
1	0	0	1
2	0	0	1

```
3    0    1    1
4    1    0    1
5    0    0    1
6    1    0    0
7    0    0    1
8    0    1    0
9    1    0    0
```

欠損値は平均値で置き換える。

```
In [7]: df = df.fillna(df.mean())
```

1.2 最小二乗法

被説明変数に与える影響の小さい説明変数を順に取り除いていく。具体的には $p > 0.05$ であるような説明変数を除いていく。
同時に、外れ値の考慮もする。

1.2.1 最小二乗法その 1

説明変数 13 個で最小二乗法を実行すると、

```
In [8]: # 定数項も加える
        X = sm.add_constant(df[['time', 'bus', 'walk', 'area',
                                'bal', 'kosuu', 'floor', 'tf', 'd_N', 'd_E', 'd_S', 'd_W', 'year']])

        # 普通の最小二乗法
        model = sm.OLS(df.price, X)
        results = model.fit()

        # 結果を表示
        print(results.summary())
```

```
OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.805
Model:                            OLS   Adj. R-squared:                  0.790
Method:                 Least Squares   F-statistic:                     54.26
Date:                Thu, 22 Oct 2015   Prob (F-statistic):           1.16e-53
Time:                        07:21:42   Log-Likelihood:                -1565.3
No. Observations:                 185   AIC:                             3159.
Df Residuals:                     171   BIC:                             3204.
Df Model:                          13
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
```

```
const        659.0401     570.899      1.154      0.250      -467.877    1785.957
time         -61.1605       7.044     -8.682      0.000       -75.065     -47.256
bus          -88.3823      21.727     -4.068      0.000      -131.269     -45.495
walk         -55.4500      20.468     -2.709      0.007       -95.852     -15.048
area          70.0731       3.379     20.737      0.000        63.403      76.743
bal          -17.0300      14.871     -1.145      0.254       -46.385      12.325
kosuu          0.0837       0.477      0.176      0.861        -0.858       1.025
floor         -2.9003      43.868     -0.066      0.947       -89.493      83.692
tf           -52.3960      37.057     -1.414      0.159      -125.545      20.753
d_N         -867.1676     653.815     -1.326      0.187     -2157.756     423.420
d_E         -341.6601     225.624     -1.514      0.132      -787.027     103.707
d_S         -684.7974     280.782     -2.439      0.016     -1239.043    -130.552
d_W         -247.0280     232.685     -1.062      0.290      -706.333     212.277
year           9.7516       5.187      1.880      0.062        -0.487      19.990
==============================================================================
Omnibus:                      126.693   Durbin-Watson:                   1.586
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             2453.138
Skew:                           2.165   Prob(JB):                         0.00
Kurtosis:                      20.306   Cond. No.                     1.80e+03
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.8e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
```

となる。

p 値を見ると、kosuu, floor がほとんど無関係であるように見える。

kosuu には外れ値が 1 つある（kosuu=2080）ので、それを除いてみる。

1.2.2 最小二乗法その 2

外れ値を除き、

```
In [9]: print(df.loc[161])
        df = df.drop(161)
```

```
time       57
bus         0
walk       15
price     800
area     57.2
bal         0
kosuu    2080
floor       1
tf          4
```

```
muki          S
year         67
d_N           0
d_E           0
d_W           0
d_S           1
Name: 161, dtype: object
```

再び最小二乗法を実行すると、

```
In [10]: X = sm.add_constant(df[['time', 'bus', 'walk', 'area', 'bal',
                                 'kosuu', 'floor', 'tf', 'd_N', 'd_E', 'd_S', 'd_W', 'year']])
         model = sm.OLS(df.price, X)
         results = model.fit()
         print(results.summary())
```

```
OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.805
Model:                            OLS   Adj. R-squared:                  0.790
Method:                 Least Squares   F-statistic:                     53.92
Date:                Thu, 22 Oct 2015   Prob (F-statistic):           2.63e-53
Time:                        07:21:42   Log-Likelihood:                -1557.0
No. Observations:                 184   AIC:                             3142.
Df Residuals:                     170   BIC:                             3187.
Df Model:                          13
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
const        648.1009    571.820      1.133      0.259      -480.681  1776.883
time         -61.6799      7.087     -8.703      0.000       -75.670   -47.689
bus          -87.3626     21.797     -4.008      0.000      -130.391   -44.334
walk         -56.4869     20.541     -2.750      0.007       -97.035   -15.939
area          70.1205      3.384     20.720      0.000        63.440    76.801
bal          -16.8531     14.892     -1.132      0.259       -46.250    12.544
kosuu         -0.3897      0.792     -0.492      0.623        -1.954     1.174
floor         -2.9617     43.925     -0.067      0.946       -89.670    83.746
tf           -42.4715     39.401     -1.078      0.283      -120.249    35.306
d_N         -890.6252    655.405     -1.359      0.176     -2184.406   403.156
d_E         -336.1515    226.034     -1.487      0.139      -782.346   110.043
d_S         -688.7604    281.193     -2.449      0.015     -1243.841  -133.680
d_W         -222.7084    235.237     -0.947      0.345      -687.070   241.653
year           9.6658      5.195      1.861      0.065        -0.589    19.920
==============================================================================
```

```
Omnibus:                      125.689   Durbin-Watson:                   1.592
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             2436.830
Skew:                           2.153   Prob(JB):                         0.00
Kurtosis:                      20.301   Cond. No.                     1.37e+03
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.37e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
```

やはり kosuu, floor の p 値が大きいので、説明変数から除くと、

1.2.3 最小二乗法その 3

```
In [11]: X = sm.add_constant(df[['time', 'bus', 'walk', 'area',
                                 'bal', 'tf', 'd_N', 'd_E', 'd_S', 'd_W', 'year']])
         model = sm.OLS(df.price, X)
         results = model.fit()
         print(results.summary())
```

```
OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.805
Model:                            OLS   Adj. R-squared:                  0.792
Method:                 Least Squares   F-statistic:                     64.35
Date:                Thu, 22 Oct 2015   Prob (F-statistic):           4.59e-55
Time:                        07:21:42   Log-Likelihood:                -1557.1
No. Observations:                 184   AIC:                             3138.
Df Residuals:                     172   BIC:                             3177.
Df Model:                          11
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
const        647.4256    568.107      1.140      0.256      -473.933  1768.784
time         -61.7145      7.051     -8.753      0.000       -75.631   -47.797
bus          -88.0394     21.589     -4.078      0.000      -130.653   -45.426
walk         -55.9212     20.403     -2.741      0.007       -96.194   -15.649
area          70.0917      3.349     20.932      0.000        63.482    76.701
bal          -16.5189     14.746     -1.120      0.264       -45.626    12.588
tf           -52.1050     27.938     -1.865      0.064      -107.251     3.041
d_N         -869.5508    649.087     -1.340      0.182     -2150.753   411.652
d_E         -336.5608    222.059     -1.516      0.131      -774.872   101.751
d_S         -682.6687    278.482     -2.451      0.015     -1232.352  -132.985
d_W         -238.6622    231.247     -1.032      0.303      -695.109   217.784
```

```
year            9.8681      5.142      1.919      0.057        -0.281     20.018
==============================================================================
Omnibus:                      125.761   Durbin-Watson:                   1.586
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             2414.208
Skew:                           2.159   Prob(JB):                         0.00
Kurtosis:                      20.212   Cond. No.                         924.
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

となる。さらに、bal と南向き以外の方角のダミー変数を説明変数から除く。

1.2.4 最小二乗法その 4

```
In [12]: X = sm.add_constant(df[['time', 'bus', 'walk', 'area', 'tf', 'year', 'd_S']])
         model = sm.OLS(df.price, X)
         results = model.fit()
         print(results.summary())
```

```
OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.799
Model:                            OLS   Adj. R-squared:                  0.791
Method:                 Least Squares   F-statistic:                     99.83
Date:                Thu, 22 Oct 2015   Prob (F-statistic):           6.87e-58
Time:                        07:21:42   Log-Likelihood:                -1559.8
No. Observations:                 184   AIC:                             3136.
Df Residuals:                     176   BIC:                             3161.
Df Model:                           7
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
const        351.2443    548.343      0.641      0.523      -730.929  1433.418
time         -63.7175      7.012     -9.087      0.000       -77.556   -49.879
bus          -84.9009     21.357     -3.975      0.000      -127.050   -42.752
walk         -54.6035     20.293     -2.691      0.008       -94.653   -14.554
area          69.5406      3.241     21.459      0.000        63.145    75.936
tf           -58.8849     27.183     -2.166      0.032      -112.531    -5.239
year           8.3053      5.081      1.634      0.104        -1.723    18.334
d_S         -433.0197    250.589     -1.728      0.086      -927.566    61.527
==============================================================================
Omnibus:                      134.268   Durbin-Watson:                   1.605
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             2841.632
Skew:                           2.343   Prob(JB):                         0.00
```

```
Kurtosis:                      21.673   Cond. No.                         730.
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

p 値が大きい南向きのダミー変数 d_E、築年数 year も説明変数から除く。

1.2.5 最小二乗法その 5

```
In [13]: X = sm.add_constant(df[['time', 'bus', 'walk', 'area', 'tf']])
         model = sm.OLS(df.price, X)
         results = model.fit()
         print(results.summary())
```

```
OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.791
Model:                            OLS   Adj. R-squared:                  0.785
Method:                 Least Squares   F-statistic:                     135.0
Date:                Thu, 22 Oct 2015   Prob (F-statistic):           1.25e-58
Time:                        07:21:42   Log-Likelihood:                -1563.2
No. Observations:                 184   AIC:                             3138.
Df Residuals:                     178   BIC:                             3158.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
const        659.1196    411.570      1.601      0.111      -153.066  1471.305
time         -63.7977      6.742     -9.463      0.000       -77.102   -50.494
bus          -92.8873     21.396     -4.341      0.000      -135.109   -50.666
walk         -58.2817     20.499     -2.843      0.005       -98.734   -17.829
area          69.1817      3.222     21.470      0.000        62.823    75.541
tf           -46.2762     26.810     -1.726      0.086       -99.183     6.631
==============================================================================
Omnibus:                      131.166   Durbin-Watson:                   1.607
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             2595.213
Skew:                           2.288   Prob(JB):                         0.00
Kurtosis:                      20.820   Cond. No.                         383.
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

p 値が大きい tf を取り除く

1.2.6 最小二乗法その 6

```
In [14]: X = sm.add_constant(df[['time', 'bus', 'walk', 'area']])
         model = sm.OLS(df.price, X)
         results = model.fit()
         print(results.summary())
```

```
OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.788
Model:                            OLS   Adj. R-squared:                  0.783
Method:                 Least Squares   F-statistic:                     166.1
Date:                Thu, 22 Oct 2015   Prob (F-statistic):           3.96e-59
Time:                        07:21:42   Log-Likelihood:                -1564.7
No. Observations:                 184   AIC:                             3139.
Df Residuals:                     179   BIC:                             3155.
Df Model:                           4
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
const        326.0739    365.543      0.892      0.374      -395.254  1047.401
time         -64.2877      6.773     -9.492      0.000       -77.653   -50.923
bus          -95.0077     21.478     -4.423      0.000      -137.391   -52.625
walk         -52.6312     20.348     -2.587      0.010       -92.783   -12.479
area          69.2456      3.240     21.373      0.000        62.852    75.639
==============================================================================
Omnibus:                      133.686   Durbin-Watson:                   1.557
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             2729.285
Skew:                           2.342   Prob(JB):                         0.00
Kurtosis:                      21.277   Cond. No.                         338.
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

修正済み R^2 = 0.783, F 統計量の p 値 3.96e-59 を見ると、**「新宿駅からの乗車時間」，「バスの乗車時間」，「徒歩時間」，「部屋の広さ」の 4 つで十分に住宅価格を説明できている**と考えられる。
最小二乗法 1〜6 と比較しても、AIC・BIC は殆ど変わらないか、改善している。

あとは残差を検討して、誤差項に関する諸仮定が満たされているかをチェックする。

1.3 残差の分析

残差に関する仮定は:

- 誤差項の平均が 0

- 誤差項の分散が一定
- 誤差項は互いに独立
- 誤差項は（少なくとも近似的には）正規分布に従う
- 誤差項と各説明変数の相関係数は 0

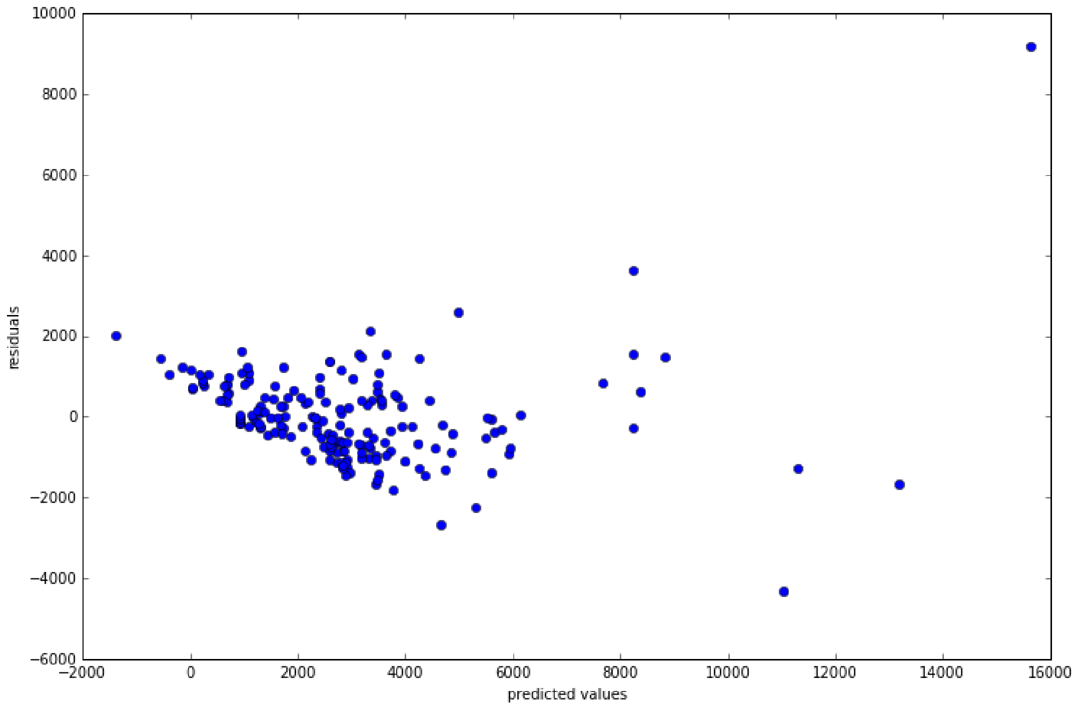
であった。
※全ての項目を厳密にチェックする方法を知らないので、出来る項目だけを確認します。

1.3.1 まずは、横軸に予測値（価格）を、縦軸に残差をとって点をプロットする。

In [15]:
```
# 回帰に使った変数だけを抜き出す
new_df = df.loc[:, ['price', 'time', 'bus', 'walk', 'area']]
# 説明変数行列
exp_matrix = new_df.loc[:, ['time', 'bus', 'walk', 'area']]
# 回帰係数ベクトル
coefs = results.params
# 理論価格ベクトル
predicted = exp_matrix.dot(coefs[1:]) + coefs[0]
#  残差ベクトル
residuals = new_df.price - predicted

# 残差を plot
fig, ax = plt.subplots(figsize=(12, 8))
plt.plot(predicted, residuals, 'o', color='b', linewidth=1, label="residuals distribution")
plt.xlabel("predicted values")
plt.ylabel("residuals")
plt.show()

# 残差平均
print("residuals mean:", residuals.mean())
```




```
residuals mean: -4.152041029832933e-12
```

平均はほぼ 0 であり、グラフでも 0 付近に点が集中していることがわかる: 仮定 1 は満たす しかしながら、右側にいくつか外れ値が見える。右上の 1 点を除いて、再度回帰分析を行う。

1.3.2 最小二乗法その 7

```
In [16]: print(new_df.loc[12] )
         new_df = new_df.drop(12)

         X = sm.add_constant(new_df[['time', 'bus', 'walk', 'area']])
         model = sm.OLS(new_df.price, X)
         results = model.fit()
         print(results.summary())
```

```
price    24800.00
time         4.00
bus          0.00
walk         8.00
area       230.72
Name: 12, dtype: float64
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.790
Model:                            OLS   Adj. R-squared:                  0.786
Method:                 Least Squares   F-statistic:                     167.7
```

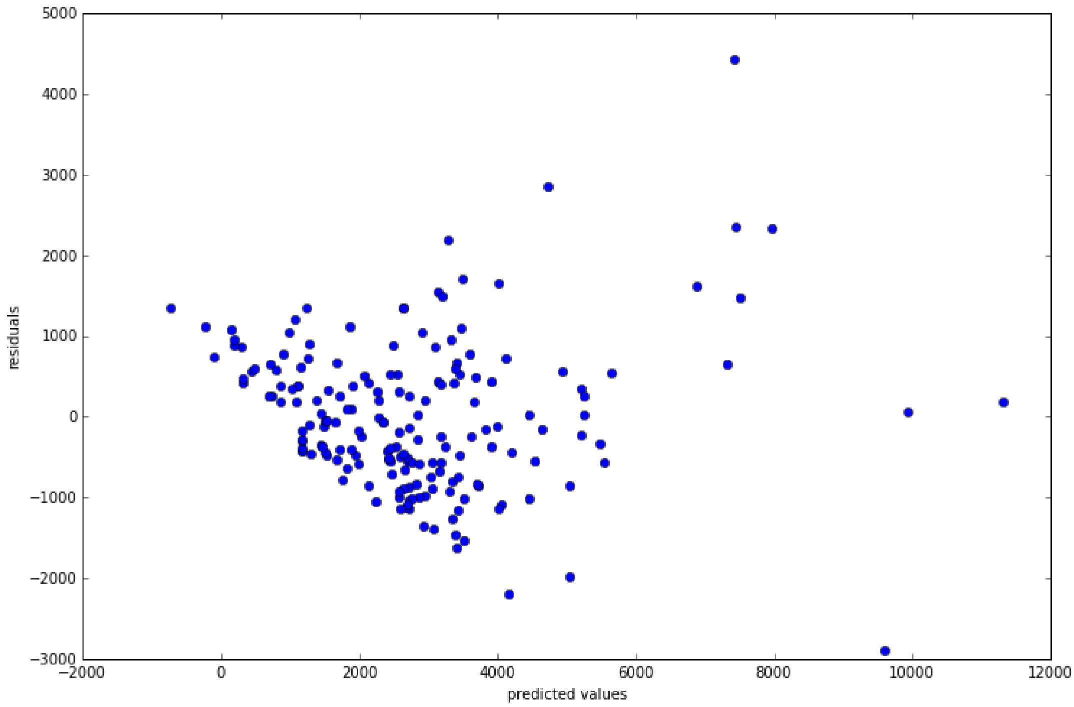
```
Date:                Thu, 22 Oct 2015   Prob (F-statistic):          3.01e-59
Time:                        07:21:42   Log-Likelihood:               -1510.6
No. Observations:                 183   AIC:                            3031.
Df Residuals:                     178   BIC:                            3047.
Df Model:                           4
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
const       1050.7368    292.682      3.590      0.000       473.164  1628.309
time         -59.3635      5.298    -11.205      0.000       -69.819   -48.908
bus          -94.7889     16.739     -5.663      0.000      -127.822   -61.756
walk         -54.4831     15.859     -3.435      0.001       -85.779   -23.187
area          56.8131      2.775     20.474      0.000        51.337    62.289
==============================================================================
Omnibus:                       30.767   Durbin-Watson:                   1.428
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               70.111
Skew:                           0.741   Prob(JB):                     5.97e-16
Kurtosis:                       5.645   Cond. No.                         340.
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

In [17]:
```python
# 説明変数行列
exp_matrix = new_df.loc[:, ['time', 'bus', 'walk', 'area']]
# 回帰係数ベクトル
coefs = results.params
# 理論価格ベクトル
predicted = exp_matrix.dot(coefs[1:]) + coefs[0]
#  残差ベクトル
residuals = new_df.price - predicted

# 残差を plot
fig, ax = plt.subplots(figsize=(12, 8))
plt.plot(predicted, residuals, 'o', color='b', linewidth=1, label="residuals distribution")
plt.xlabel("predicted values")
plt.ylabel("residuals")
plt.show()

# 残差平均
print("residuals mean:", residuals.mean())
```




```
residuals mean: 1.6127378728159302e-12
```

最小二乗法 6 の結果に比べ、ばらつきが均等になった。

1.3.3 次に、縦軸に残差、横軸に各説明変数の観測値をとって、残差のばらつきを見る。

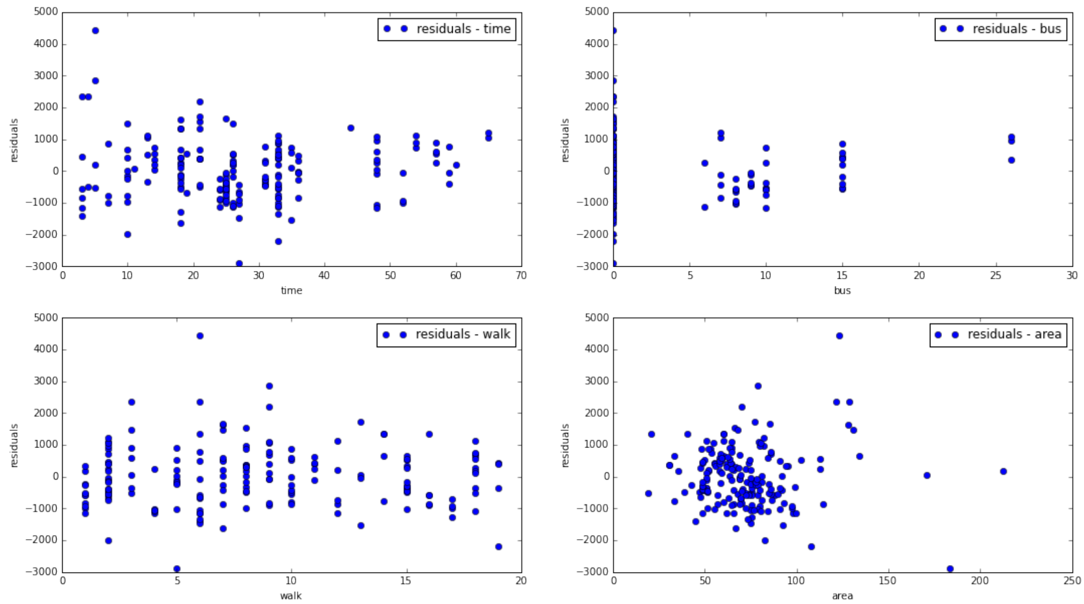
In [18]:
```
# 残差を plot
fig = plt.figure(figsize=(18, 10))
ax1 = plt.subplot(2, 2, 1)
plt.plot(exp_matrix['time'], residuals, 'o', color='b', linewidth=1, label="residuals - tim
plt.xlabel("time")
plt.ylabel("residuals")
plt.legend()

ax2 = plt.subplot(2, 2, 2, sharey=ax1)
plt.plot(exp_matrix['bus'], residuals, 'o', color='b', linewidth=1, label="residuals - bus"
plt.xlabel("bus")
plt.ylabel("residuals")
plt.legend()

ax3 = plt.subplot(2, 2, 3, sharey=ax1)
plt.plot(exp_matrix['walk'], residuals, 'o', color='b', linewidth=1, label="residuals - wal
plt.xlabel("walk")
plt.ylabel("residuals")
plt.legend()
```

```
ax4 = plt.subplot(2, 2, 4, sharey=ax1)
plt.plot(exp_matrix['area'], residuals, 'o', color='b', linewidth=1, label="residuals - are
plt.xlabel("area")
plt.ylabel("residuals")
plt.legend()

plt.show()
```



どの説明変数と残差の間にも特徴的な相関関係は見られない: 仮定 5 は満たす。

area 変数だけ残差のばらつき方が異なるので、何らかの対策をとったほうが良い可能性がある（すみません、わかりません)。

1.4 まとめ

当てはまりの良い回帰モデルを作ることが出来たが、残差の性質、特に等分散性の仮定を置いて良いのかについては問題が残った。等分散性の仮定に問題がある場合は、重みを付けて最小二乗法を使う必要があるので、慎重に考える必要がある。

```
In [ ]:
```