In [1]:

```
%matplotlib inline
```

In [2]:

```
import numpy as np
import re
import matplotlib.pyplot as plt
import pandas as pd
import statsmodels.api as sm
```

Helper functions

In [3]:

```
def read_data(file_name, has_header):
    f = open(file_name, 'r')
    if has_header:
        header = re.sub('["\n]', '', f.readline()).split(' ')
    matrix = []
    data = []
    for line in f:
        items = line.split(' ')
        if len(items) > 1:
            matrix.append(items[0])
            del items[0]
            data.append(list(map(float, items)))
    if has_header:
        return (matrix, np.asmatrix(data), header)
    else:
        return matix, np.asmatrix(data)
```

In [4]:

```
def calc_avg_data(prefix, suffix ,files):
    avg_data = None
    for f in files:
        matrix, data, header = read_data(prefix + f + suffix, True)
        if avg_data is None:
            avg_data = data
        else:
            avg_data = avg_data + data
    avg_data = avg_data / len(files)
    return matrix, avg_data, header
```

In [5]:

```
def plot_relative_values(matrix, data, header):
    relative_data = data / data.max(1)
    fig = plt.figure(figsize = (400,data.shape[1]))
    ax = fig.add_axes([0.0, 0.0, 1.0, 1.0])
    ax.set_yticks(range(data.shape[1]))
    ax.set_yticklabels(header[1:(data.shape[1]+1)])
    ax.set_xticks([i for i in range(len(matrix))])
    ax.set_xticklabels(matrix, rotation=45, ha="right")
    plt.imshow(relative_data.T, interpolation='nearest', cmap="OrRd_r")
    plt.colorbar()
    plt.show()
```

In [6]:

```
def plot_statistics(matrix, data, header):
    fig = plt.figure(figsize = (444,24))
    for i in range(0, 24, 1):
        ax = plt.subplot(24, 1, i+1)
        ax.set_yticks([0])
        ax.set_yticklabels([header[i+1]])
        if i == 23:
            ax.set_xticks(range(len(matrix)))
            ax.set_xticklabels(matrix, rotation=45, ha="right")
        else:
            ax.get_xaxis().set_visible(False)
        plt.imshow(data.T[i-1,:], interpolation='nearest', cmap="OrRd_r")
        plt.colorbar()

    plt.show()
```

Visualization of matrix vector multiplication runtime, format conversion runtime and matrix statistics

In [7]:

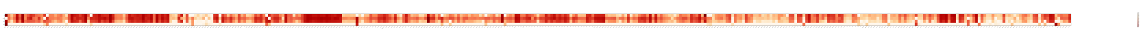
```
files = ["25615", "391750", "494030", "497461", "643522", "674742", "68520
3", "810763", "867607", "872897"]
```

In [8]:

```
matrix_matvec, avg_data_matvec, header_matvec = calc_avg_data("../stats/Mat
VecMultStats_", ".txt", files)
```

In [9]:

```
plot_relative_values(matrix_matvec, avg_data_matvec, header_matvec)
```



In [10]:

```
matrix_convert, avg_data_convert, header_convert = calc_avg_data("../stat
s/MatVecConvertStats_", ".txt", files)
```

In [11]:

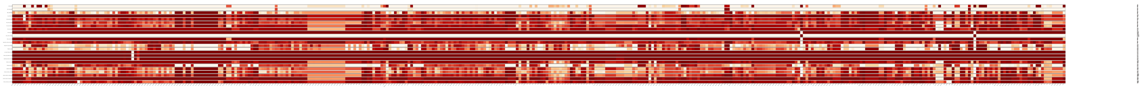
```
plot_relative_values(matrix_convert, avg_data_convert, header_convert)
```


In [12]:

```
matrix_stats, avg_data_stats, header_stats = calc_avg_data("../stats/MtxSta
ts_", ".txt", files)
```

In [13]:

```
plot_statistics(matrix_stats, avg_data_stats, header_stats)
```



Investigating performance improvements on using different formats

In [14]:

```
print "Total time in nanoseconds for executing the matrix vector multiplica
tion (all test matrices) in CSR format only"
total_csr = avg_data_matvec[:,0].sum()
total_csr
```

```
Total time in nanoseconds for executing the matrix vector multi
plication (all test matrices) in CSR format only
```

Out[14]:

```
3850490.8999999999
```

In [15]:

```
print "Total time in nanoseconds for executing the matrix vector multiplica
tion (all test matrices) in the fastest format always"
total_best = avg_data_matvec.min(1).sum()
total_best
```

```
Total time in nanoseconds for executing the matrix vector multi
plication (all test matrices) in the fastest format always
```

Out[15]:

```
2538347.3000000003
```

In [16]:

```
print "Performance increase (speedup)"
total_csr / total_best
```

```
Performance increase (speedup)
```

Out[16]:

```
1.5169283178862087
```

In [17]:

```
#indices of the best format
indices = avg_data_matvec.argmin(1).A1
#conversion time from CSR to the best format (0 when CSR is best Format)
convert_time = avg_data_convert[range(avg_data_convert.shape[0]), indices-
1].A1
convert_time[indices == 0] = 0
print "Total conversion time in nanoseconds when using best multiplication
format"
convert_time.sum()
```

```
Total conversion time in nanoseconds when using best multiplica
tion format
```

Out[17]:

```
6069641.0
```

In [18]:

```
print "Performace decrease when adding conversion time to multiplication ti
me (speedup)"
total_csr / (total_best + convert_time.sum())
```

```
Performace decrease when adding conversion time to multiplicati
on time (speedup)
```

Out[18]:

```
0.44731600065023319
```

Investigating influence of matrix statistics on multiplication runtime of different formats (using linear regression)

In [19]:

```
Y = avg_data_matvec / np.linalg.norm(avg_data_matvec)
```

In [20]:

```
Y_header = header_matvec
del Y_header[0]
```

In [21]:

```
df_Y = pd.DataFrame(Y, columns = Y_header)
```

In [22]:

```
norm_x = np.ones_like(avg_data_stats)

for i in range(avg_data_stats.shape[1]):
    norm_x[:, i] = avg_data_stats[:, i] / np.linalg.norm(avg_data_stats[:,
i])

X = sm.add_constant(norm_x)
```

In [23]:

```
X_header = header_stats
X_header[0] = "const"
df_X = pd.DataFrame(X, columns = X_header)
```

In [24]:

```
#removing statistics that cause linear regression to fail (they have NaN el
ements)
del df_X[X_header[20]]
del df_X[X_header[19]]
del df_X[X_header[18]]
X = np.delete(X, 20, axis=1)
X = np.delete(X, 19, axis=1)
X = np.delete(X, 18, axis=1)
```

In [25]:

```
model = sm.OLS(df_Y['CSR'], df_X)
results = model.fit()
print results.summary()
```

```
                            OLS Regression Results
==============================================================
===============
Dep. Variable:                    CSR   R-squared:
0.104
Model:                            OLS   Adj. R-squared:
0.053
Method:                 Least Squares   F-statistic:
2.037
Date:                Sun, 13 Sep 2015   Prob (F-statistic):
0.00486
Time:                        14:50:23   Log-Likelihood:
1113.6
No. Observations:                 389   AIC:
-2183.
Df Residuals:                     367   BIC:
-2096.
Df Model:                          21
Covariance Type:            nonrobust
```

```
================================================================
=================================
                                    coef    std err          t
P>|t|      [95.0% Conf. Int.]
----------------------------------------------------------------
---------------------------------
const                             0.0102      0.009      1.160
0.247        -0.007     0.027
rc_dimension                     -0.1725      0.205     -0.842
0.400        -0.576     0.231
lower_bandwidth                   0.0720      0.061      1.182
0.238        -0.048     0.192
upper_bandwidth                   0.0268      0.034      0.797
0.426        -0.039     0.093
max_bandwidth                    -0.0709      0.071     -1.005
0.316        -0.210     0.068
average_bandwidth                 0.0897      0.077      1.169
0.243        -0.061     0.241
max_column_length                -0.1089      0.069     -1.573
0.117        -0.245     0.027
min_column_length                -0.1515      0.035     -4.387
0.000        -0.219    -0.084
max_row_length                -7674.1039   1.23e+04     -0.623
0.534     -3.19e+04  1.66e+04
min_row_length                -7674.1050   1.23e+04     -0.623
0.534     -3.19e+04  1.66e+04
zero_column_number                0.0010      0.029      0.034
0.973        -0.055     0.057
zero_row_number                   0.0139      0.044      0.320
0.749        -0.072     0.100
diag_domi_column_percentage       0.0098      0.075      0.130
0.896        -0.137     0.157
diag_domi_row_percentage          0.0121      0.073      0.165
0.869        -0.132     0.156
Frobenius_norm                  -13.0053     20.152     -0.645
0.519       -52.633    26.623
sym_part_Frobenius_norm          12.8071     19.825      0.646
0.519       -26.177    51.791
nonsym_part_Frobenius_norm        0.2023      0.342      0.591
0.555        -0.470     0.875
matching_elements_number         -0.0003      0.065     -0.005
0.996        -0.127     0.127
nonzero_number_in_skyline        -0.0826      0.076     -1.093
0.275        -0.231     0.066
nonzero_number_of_lower_part      0.1695      0.072      2.352
0.019         0.028     0.311
nonzero_number_of_upper_part      0.0943      0.051      1.838
0.067        -0.007     0.195
nonzero_number_of_maindiag        0.0256      0.083      0.309
0.758        -0.137     0.189
================================================================
===============
Omnibus:                        852.290   Durbin-Watson:
2.121
Prob(Omnibus):                    0.000   Jarque-Bera (JB):
1480668.112
Skew:                            16.257   Prob(JB):
```

```
0.00
Kurtosis:                     303.491   Cond. No.
2.44e+07
==============================================================
===============

Warnings:
[1] Standard Errors assume that the covariance matrix of the er
rors is correctly specified.
[2] The smallest eigenvalue is 6.66e-13. This might indicate th
at there are
strong multicollinearity problems or that the design matrix is
singular.
```

In [26]:

```python
model = sm.OLS(df_Y['ModCSR'], df_X)
results = model.fit()
print results.summary()
```

```
                            OLS Regression Results
==============================================================
===============
Dep. Variable:                 ModCSR   R-squared:
0.019
Model:                            OLS   Adj. R-squared:
-0.037
Method:                 Least Squares   F-statistic:
0.3382
Date:                Sun, 13 Sep 2015   Prob (F-statistic):
0.998
Time:                        14:50:23   Log-Likelihood:
749.41
No. Observations:                 389   AIC:
-1455.
Df Residuals:                     367   BIC:
-1368.
Df Model:                          21
Covariance Type:            nonrobust
==============================================================
=================================
                                  coef    std err          t
P>|t|      [95.0% Conf. Int.]
--------------------------------------------------------------
---------------------------------
const                           0.0088      0.022      0.393
0.695       -0.035      0.053
rc_dimension                   -0.1733      0.523     -0.331
0.740       -1.201      0.855
lower_bandwidth                 0.1002      0.155      0.645
0.519       -0.205      0.406
upper_bandwidth                -0.0194      0.086     -0.226
0.821       -0.188      0.149
max_bandwidth                   0.0044      0.180      0.024
0.981       -0.350      0.358
average_bandwidth              -0.0043      0.196     -0.022
0.983       -0.389      0.381
```

```
max_column_length                    -0.0700      0.177     -0.397
0.692        -0.417     0.277
min_column_length                    -0.1305      0.088     -1.481
0.139        -0.304     0.043
max_row_length                     4962.0579   3.14e+04      0.158
0.875     -5.68e+04  6.68e+04
min_row_length                     4962.0639   3.14e+04      0.158
0.875     -5.68e+04  6.68e+04
zero_column_number                   -0.0160      0.073     -0.220
0.826        -0.159     0.127
zero_row_number                       0.0502      0.111      0.452
0.652        -0.168     0.269
diag_domi_column_percentage           0.0549      0.191      0.288
0.774        -0.321     0.430
diag_domi_row_percentage              0.0705      0.187      0.377
0.706        -0.297     0.438
Frobenius_norm                       -7.7421     51.393     -0.151
0.880      -108.803    93.319
sym_part_Frobenius_norm               7.6438     50.557      0.151
0.880       -91.775   107.062
nonsym_part_Frobenius_norm            0.0970      0.872      0.111
0.912        -1.619     1.813
matching_elements_number             -0.0062      0.165     -0.038
0.970        -0.330     0.318
nonzero_number_in_skyline            -0.1010      0.193     -0.524
0.600        -0.480     0.278
nonzero_number_of_lower_part          0.1960      0.184      1.066
0.287        -0.165     0.557
nonzero_number_of_upper_part          0.0745      0.131      0.570
0.569        -0.183     0.332
nonzero_number_of_maindiag           -0.0303      0.212     -0.143
0.886        -0.446     0.386
===============================================================
===============
Omnibus:                         854.441   Durbin-Watson:
1.170
Prob(Omnibus):                     0.000   Jarque-Bera (JB):
1376601.008
Skew:                             16.408   Prob(JB):
0.00
Kurtosis:                        292.577   Cond. No.
2.44e+07
===============================================================
===============

Warnings:
[1] Standard Errors assume that the covariance matrix of the er
rors is correctly specified.
[2] The smallest eigenvalue is 6.66e-13. This might indicate th
at there are
strong multicollinearity problems or that the design matrix is
singular.
```

In [27]:

```
model = sm.OLS(df_Y['Jagged'], df_X)
results = model.fit()
print results.summary()
```

```
                            OLS Regression Results
==============================================================================
===============
Dep. Variable:                 Jagged   R-squared:
0.097
Model:                            OLS   Adj. R-squared:
0.046
Method:                 Least Squares   F-statistic:
1.881
Date:                Sun, 13 Sep 2015   Prob (F-statistic):
0.0113
Time:                        14:50:23   Log-Likelihood:
946.16
No. Observations:                 389   AIC:
-1848.
Df Residuals:                     367   BIC:
-1761.
Df Model:                          21
Covariance Type:            nonrobust
==============================================================================
=================================
                                   coef    std err          t
P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
---------------------------------
const                            0.0163      0.013      1.210
0.227        -0.010     0.043
rc_dimension                    -0.3041      0.315     -0.965
0.335        -0.924     0.316
lower_bandwidth                  0.1069      0.094      1.141
0.254        -0.077     0.291
upper_bandwidth                  0.0375      0.052      0.725
0.469        -0.064     0.139
max_bandwidth                   -0.0943      0.109     -0.869
0.386        -0.308     0.119
average_bandwidth                0.0978      0.118      0.828
0.408        -0.134     0.330
max_column_length               -0.1707      0.106     -1.603
0.110        -0.380     0.039
min_column_length               -0.2228      0.053     -4.194
0.000        -0.327    -0.118
max_row_length               -1.229e+04    1.9e+04     -0.649
0.517     -4.96e+04    2.5e+04
min_row_length               -1.229e+04    1.9e+04     -0.649
0.517     -4.96e+04    2.5e+04
zero_column_number              -0.0015      0.044     -0.035
0.972        -0.088     0.085
zero_row_number                  0.0314      0.067      0.469
0.639        -0.100     0.163
diag_domi_column_percentage      0.0567      0.115      0.492
0.623        -0.170     0.283
diag_domi_row_percentage         0.0129      0.113      0.115
```

```
diag_domi_row_percentage         -0.0129      0.113     -0.115
0.909       -0.234      0.209
Frobenius_norm                   -14.7531     30.992     -0.476
0.634      -75.697     46.191
sym_part_Frobenius_norm           14.5314     30.488      0.477
0.634      -45.422     74.485
nonsym_part_Frobenius_norm         0.2257      0.526      0.429
0.668       -0.809      1.260
matching_elements_number           0.0011      0.099      0.011
0.991       -0.194      0.197
nonzero_number_in_skyline         -0.1102      0.116     -0.949
0.343       -0.339      0.118
nonzero_number_of_lower_part       0.2595      0.111      2.341
0.020        0.042      0.477
nonzero_number_of_upper_part       0.1526      0.079      1.934
0.054       -0.003      0.308
nonzero_number_of_maindiag         0.0242      0.128      0.190
0.849       -0.227      0.275
===============================================================
===============
Omnibus:                      833.494   Durbin-Watson:
2.078
Prob(Omnibus):                  0.000   Jarque-Bera (JB):
1268355.570
Skew:                          15.494   Prob(JB):
0.00
Kurtosis:                     281.016   Cond. No.
2.44e+07
===============================================================
===============

Warnings:
[1] Standard Errors assume that the covariance matrix of the er
rors is correctly specified.
[2] The smallest eigenvalue is 6.66e-13. This might indicate th
at there are
strong multicollinearity problems or that the design matrix is
singular.
```

In [28]:

```
model = sm.OLS(df_Y['Ellpack'], df_X)
results = model.fit()
print results.summary()
```

```
                            OLS Regression Results
===============================================================
===============
Dep. Variable:                Ellpack   R-squared:
0.085
Model:                            OLS   Adj. R-squared:
0.033
Method:                 Least Squares   F-statistic:
1.633
Date:                Sun, 13 Sep 2015   Prob (F-statistic):
0.0398
Time:                        14:50:23   Log-Likelihood:
```

```
967.08
No. Observations:                 389   AIC:
-1890.
Df Residuals:                     367   BIC:
-1803.
Df Model:                          21
Covariance Type:            nonrobust
===============================================================================
=================================
                                  coef    std err          t
P>|t|      [95.0% Conf. Int.]
-------------------------------------------------------------------------------
---------------------------------
const                          -0.0003      0.013     -0.024
0.981        -0.025     0.025
rc_dimension                    0.0119      0.299      0.040
0.968        -0.575     0.599
lower_bandwidth                 0.0355      0.089      0.400
0.689        -0.139     0.210
upper_bandwidth                -0.0187      0.049     -0.382
0.703        -0.115     0.078
max_bandwidth                   0.0451      0.103      0.438
0.661        -0.157     0.247
average_bandwidth              -0.0538      0.112     -0.481
0.631        -0.274     0.166
max_column_length               0.0512      0.101      0.508
0.612        -0.147     0.250
min_column_length               0.0218      0.050      0.433
0.666        -0.077     0.121
max_row_length              -6747.3447    1.8e+04     -0.376
0.707     -4.21e+04  2.86e+04
min_row_length              -6747.3462    1.8e+04     -0.376
0.707     -4.21e+04  2.86e+04
zero_column_number             -0.0129      0.042     -0.308
0.758        -0.095     0.069
zero_row_number                 0.0097      0.063      0.153
0.879        -0.115     0.135
diag_domi_column_percentage     0.0211      0.109      0.193
0.847        -0.193     0.236
diag_domi_row_percentage       -0.0588      0.107     -0.551
0.582        -0.269     0.151
Frobenius_norm                  0.2387     29.369      0.008
0.994       -57.514    57.991
sym_part_Frobenius_norm        -0.2522     28.892     -0.009
0.993       -57.066    56.562
nonsym_part_Frobenius_norm      0.0135      0.499      0.027
0.978        -0.967     0.994
matching_elements_number       -0.0330      0.094     -0.350
0.726        -0.218     0.152
nonzero_number_in_skyline       0.0288      0.110      0.262
0.794        -0.188     0.245
nonzero_number_of_lower_part    0.0624      0.105      0.595
0.553        -0.144     0.269
nonzero_number_of_upper_part    0.0596      0.075      0.796
0.426        -0.087     0.207
nonzero_number_of_maindiag      0.0649      0.121      0.537
0.592        -0.173     0.303
```

```
==============================================================================
Omnibus:                      889.195   Durbin-Watson:
2.011
Prob(Omnibus):                  0.000   Jarque-Bera (JB):
1849129.897
Skew:                          17.898   Prob(JB):
0.00
Kurtosis:                     338.863   Cond. No.
2.44e+07
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the er
rors is correctly specified.
[2] The smallest eigenvalue is 6.66e-13. This might indicate th
at there are
strong multicollinearity problems or that the design matrix is
singular.
```