livelessons
video instruction from technology experts

Lesson 3: Your First Spark Application

3.5 Making Sense of Data: Summary Statistics and Distributions

Frequently Occurring Values

- `dataframe.freqItems(columns, support)`

Note: this is an approximate algorithm that **always** returns all the frequent items, but may contain false positives.

required minimum proportion of rows

```
freq_items = df.freqItems(['school_city', 'primary_focus_area', \
                           'grade_level', 'poverty_level','resource'], 0.7).collect()
```

```
freq_items[0]
```

```
Row(school_city_freqItems=[u'Los Angeles'], primary_focus_area_freqItems=[u'Literacy & Language'], grade_level_freqIt
ems=[u'Grades PreK-2'], poverty_level_freqItems=[u'highest poverty'], resource_freqItems=[u'Supplies'])
```

http://dl.acm.org/citation.cfm?doid=762471.762473

Summary Statistics

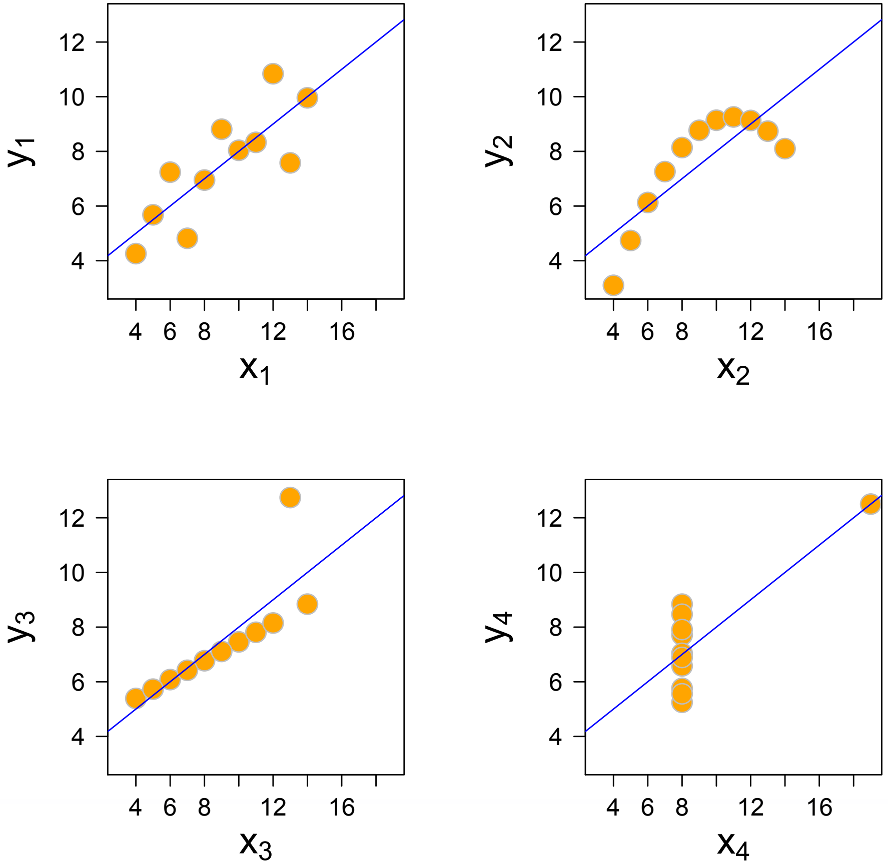
- dataframe.describe(column_name)

```
df.select('total_donations', 'num_donors', 'students_reached', \
                 df_dates['total_price_excluding_optional_support'].alias('p_exclude'), \
                 df_dates['total_price_including_optional_support'].alias('p_include')) \
  .describe().show()
```

```
+-------+------------------+-----------------+-----------------+------------------+------------------+
|summary|   total_donations|       num_donors| students_reached|         p_exclude|         p_include|
+-------+------------------+-----------------+-----------------+------------------+------------------+
|  count|            771929|           771929|           771779|            771929|            771929|
|   mean|370.85023398481707|4.264279486843997|96.71620114048193| 569.6223687446723|  676.180708551764|
| stddev| 733.4647726421459|6.132976060232441|2118.592960253374|11763.955807309705|14344.347534777195|
|    min|               0.0|              0.0|              0.0|               0.0|               0.0|
|    max|          244778.0|            521.0|         999999.0|        1.0250017E7|        1.2500021E7|
+-------+------------------+-----------------+-----------------+------------------+------------------+
```

Interlude: Sometimes numbers aren’t enough!

Anscombe's Quartet



Mean (x)	9
Sample Variance (x)	11
Mean (y)	7.50
Sample Variance (y)	4.127
Correlation	0.816
Linear Regression	y = 3.00 + 0.500x

livelessons

Distributions

RDD

- `rdd.histogram()`
- `rdd.stats()`

DataFrame

- `dataframe.groupby('column_name').count()`
- `dataframe.describe('column_name')`

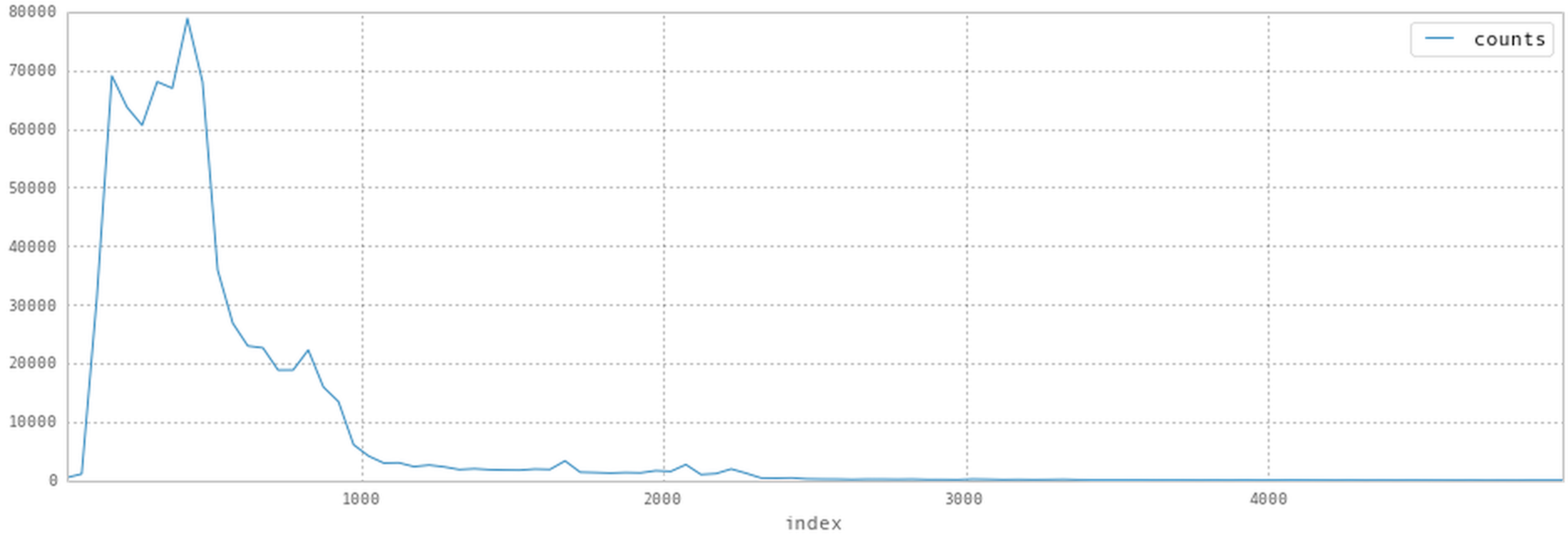
```
price_rdd = df_no_null.select('total_price_excluding_optional_support').rdd.map(lambda r: r.asDict().values()[0])
```

```
def plot_rdd_hist(hist):
    idx = []

    for i in range(len(hist[0]) - 1):
        idx.append((hist[0][i] + hist[0][i+1])/ 2)

    pd.DataFrame({'counts': hist[1], 'index': idx}).set_index('index').plot(figsize=(16,5))
```

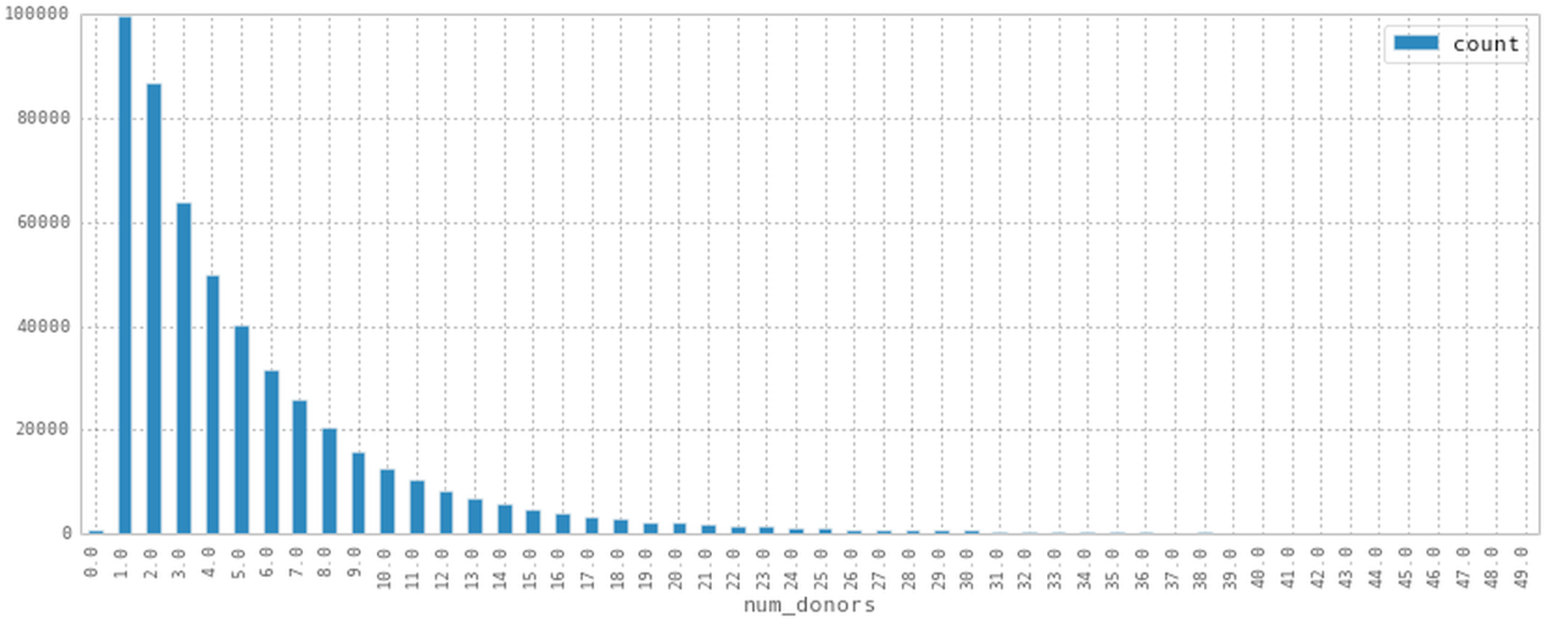
```
plot_rdd_hist(price_rdd.filter(lambda x: x < 5000).histogram(100))
```



```python
def spark_histogram(df, column):
    donor_counts = df.groupby(column).count()
    donor_df = donor_counts.toPandas()
    donor_df[column] = donor_df.num_donors.astype(float)
    return donor_df.sort(column).set_index(column).iloc[:50,:].plot(kind='bar', figsize=(14,5))
```

```python
spark_histogram(df_complete, 'num_donors')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x113b2be50>
```



Interactions

- `dataframe.crosstab()`
- `dataframe.corr()`

```
df_no_null.stat.corr('total_price_excluding_optional_support', 'num_donors')
```

```
0.007004254706419042
```

```
df_no_null.stat.corr('total_price_excluding_optional_support', 'students_reached')
```

```
0.0006159991686679948
```

```
df_no_null.stat.corr('total_price_excluding_optional_support', 'total_price_including_optional_support')
```

```
0.9999972199123168
```

```
df_dates.crosstab('resource_type', 'funding_status').show()
df_dates.crosstab('primary_focus_area', 'resource_type').show()
```

```
+--------------------------+-----+---------+-----------+-------+
|resource_type_funding_status| live|completed|reallocated|expired|
+--------------------------+-----+---------+-----------+-------+
|                      null|    2|       28|          0|     18|
|                     Other| 4542|    54610|        747|  22550|
|                     Books| 5982|   118810|       1527|  34554|
|                  Visitors|  102|      806|          6|    341|
|                  Supplies|11939|   185870|       2602|  63406|
|                     Trips|  347|     4381|         62|   1474|
|                Technology|18957|   150500|       2256|  85510|
+--------------------------+-----+---------+-----------+-------+

+------------------------------+-----+--------+-----+----------+------+--------+----+
|primary_focus_area_resource_type|Trips|Visitors|Other|Technology| Books|Supplies|null|
+------------------------------+-----+--------+-----+----------+------+--------+----+
|           Literacy & Language|  630|     228|32795|    109605|127282|   75924|   4|
|                          null|    0|       0|    0|         0|     1|       0|  41|
|              Applied Learning| 1197|     104| 9429|     17869|  4863|   22596|   0|
|                Math & Science| 1902|     323|16353|     75189| 11746|   89101|   3|
|              Music & The Arts|  947|     441| 8305|     19289|  2883|   37804|   0|
|               Health & Sports|  159|      54| 4633|      3054|   432|   12970|   0|
|                 Special Needs|  241|      32| 7636|     19359|  4112|   17151|   0|
|              History & Civics| 1188|      73| 3298|     12858|  9554|    8271|   0|
+------------------------------+-----+--------+-----+----------+------+--------+----+
```

Review

- Interactive REPL
- Rapid computation (especially aggregates) on large amounts of data
- High level abstractions for efficient querying of data
- "Condense" data for easier local exploration and visualization