DMC2016

April 22, 2016

```
In [1]: import graphlab
```

```
A newer version of GraphLab Create (v1.8.5) is available! Your current version is v1.8.4.

You can use pip to upgrade the graphlab-create package. For more information see https://dato.com/produc
```

```
In [2]: url = 'C:/Users/yehplus.bl0076-xx/Dropbox/DMC2016/featureMatrix_v01/train.csv'
```

```
In [2]: train_url_mac = '/Users/yaosuhan/Desktop/DMC_2016_local/featureMatrix_v03/train.csv'
        test_url_mac = '/Users/yaosuhan/Desktop/DMC_2016_local/featureMatrix_v03/test.csv'
```

```
In [3]: raw = graphlab.SFrame.read_csv(train_url_mac)
```

```
2016-04-22 16:21:11,138 [INFO] graphlab.cython.cy_server, 176: GraphLab Create v1.8.4 started. Logging:

Finished parsing file /Users/yaosuhan/Desktop/DMC_2016_local/featureMatrix_v03/train.csv


Parsing completed. Parsed 100 lines in 0.940106 secs.


This non-commercial license of GraphLab Create is assigned to yeh@iastate.edu and will expire on March 
------------------------------------------------------

Read 42821 lines. Lines per second: 23639


Finished parsing file /Users/yaosuhan/Desktop/DMC_2016_local/featureMatrix_v03/train.csv


Parsing completed. Parsed 315445 lines in 6.87474 secs.


Inferred types from first line of file as
column_type_hints=[int,int,float,float,float,float,float,float,float,float,float,float,float,float,float
If parsing fails due to incorrect types, you can correct
the inferred type list above and pass it to read_csv in
the column_type_hints argument
------------------------------------------------------
```

```
In [4]: test = graphlab.SFrame.read_csv(test_url_mac)
```

```
Finished parsing file /Users/yaosuhan/Desktop/DMC_2016_local/featureMatrix_v03/test.csv


Parsing completed. Parsed 100 lines in 0.920094 secs.
```

```
Read 42272 lines. Lines per second: 23821.7


Finished parsing file /Users/yaosuhan/Desktop/DMC_2016_local/featureMatrix_v03/test.csv


Parsing completed. Parsed 117779 lines in 2.83133 secs.


------------------------------------------------------
Inferred types from first line of file as
column_type_hints=[int,int,float,float,float,float,float,float,float,float,float,float,float,float,float
If parsing fails due to incorrect types, you can correct
the inferred type list above and pass it to read_csv in
the column_type_hints argument
------------------------------------------------------
```

```
In [5]: train,valid = raw.random_split(.8)
```

```
In [6]: train_new = train.dropna('rrp')
```

```
In [7]: model_boosted_tree = graphlab.boosted_trees_classifier.create(raw,target='returnQuantity',
                                                                       max_iterations=100)
```

```
Boosted trees classifier:


--------------------------------------------------------


Number of examples          : 299782


Number of classes           : 6


Number of feature columns   : 98


Number of unpacked features : 98


+-----------+--------------+-------------------+-------------------+---------------------+-------------


| Iteration | Elapsed Time | Training-accuracy | Training-log_loss | Validation-accuracy | Validation-l


+-----------+--------------+-------------------+-------------------+---------------------+-------------


| 1         | 6.143165     | 0.677282          | 1.384584          | 0.674137            | 1.385116


| 2         | 12.300334    | 0.685151          | 1.165891          | 0.680585            | 1.167275


| 3         | 17.550624    | 0.687730          | 1.022610          | 0.683139            | 1.024590
| 4          | 23.124646   | 0.688781          | 0.922301          | 0.684352            | 0.924375

| 5          | 28.547202   | 0.689835          | 0.849165          | 0.685373            | 0.851561

| 6          | 34.649265   | 0.690942          | 0.794231          | 0.685692            | 0.796904

| 7          | 39.902395   | 0.691563          | 0.752350          | 0.686522            | 0.755142

| 8          | 45.074196   | 0.692076          | 0.719843          | 0.687161            | 0.722942

| 9          | 50.376700   | 0.692810          | 0.694169          | 0.687672            | 0.697372

| 10         | 56.105041   | 0.693150          | 0.673672          | 0.688757            | 0.677155

| 11         | 61.378295   | 0.693431          | 0.657115          | 0.688885            | 0.660906

| 12         | 66.792222   | 0.693794          | 0.643464          | 0.688246            | 0.647353

| 13         | 72.170216   | 0.694078          | 0.632287          | 0.688821            | 0.636365

| 14         | 78.533600   | 0.694731          | 0.622610          | 0.689651            | 0.626923

| 15         | 83.760975   | 0.695375          | 0.614479          | 0.689842            | 0.619016

| 16         | 88.962506   | 0.696216          | 0.607440          | 0.690417            | 0.612586

| 17         | 94.826288   | 0.697043          | 0.601300          | 0.691183            | 0.606903

| 18         | 99.933049   | 0.697554          | 0.596267          | 0.691119            | 0.602256

| 19         | 105.055165  | 0.698124          | 0.592009          | 0.691502            | 0.598307

| 20         | 110.149930  | 0.698748          | 0.588104          | 0.691758            | 0.594956

| 21         | 115.314138  | 0.699412          | 0.584838          | 0.692013            | 0.592194

| 22         | 120.700775  | 0.699762          | 0.582053          | 0.691119            | 0.589695

| 23         | 125.932835  | 0.700352          | 0.579625          | 0.691375            | 0.587644
| 24      | 130.981896  | 0.701169      | 0.577279      | 0.691630        | 0.585890

| 25      | 136.029217  | 0.701546      | 0.575090      | 0.692013        | 0.584350

| 26      | 142.437439  | 0.702010      | 0.573405      | 0.692268        | 0.582991

| 27      | 149.884867  | 0.702594      | 0.571730      | 0.692013        | 0.581810

| 28      | 154.870633  | 0.702774      | 0.570557      | 0.691694        | 0.580860

| 29      | 159.988958  | 0.703084      | 0.569491      | 0.692077        | 0.580004

| 30      | 165.074487  | 0.703378      | 0.568426      | 0.691821        | 0.579440

| 31      | 170.093345  | 0.703781      | 0.567249      | 0.691949        | 0.578595

| 32      | 175.069587  | 0.704092      | 0.566467      | 0.691949        | 0.578025

| 33      | 180.185950  | 0.704639      | 0.565464      | 0.691566        | 0.577816

| 34      | 185.447219  | 0.704762      | 0.564938      | 0.691949        | 0.577424

| 35      | 190.470163  | 0.704979      | 0.564351      | 0.691438        | 0.577200

| 36      | 195.855027  | 0.705249      | 0.563711      | 0.692141        | 0.576851

| 37      | 202.877823  | 0.705466      | 0.562991      | 0.691949        | 0.576691

| 38      | 208.173630  | 0.705980      | 0.562320      | 0.691438        | 0.576421

| 39      | 213.812139  | 0.706280      | 0.561738      | 0.691502        | 0.576268

| 40      | 218.980700  | 0.707054      | 0.560921      | 0.691885        | 0.575813

| 41      | 224.117475  | 0.707471      | 0.560173      | 0.691949        | 0.575667

| 42      | 229.222973  | 0.707788      | 0.559714      | 0.692332        | 0.575576

| 43      | 234.045439  | 0.707904      | 0.559497      | 0.692268        | 0.575544
| 44      | 239.587690  | 0.708091      | 0.559095      | 0.692460        | 0.575410

| 45      | 245.194276  | 0.708762      | 0.558444      | 0.692588        | 0.575341

| 46      | 250.040814  | 0.708862      | 0.558201      | 0.692460        | 0.575266

| 47      | 255.099921  | 0.709275      | 0.557560      | 0.692141        | 0.575034

| 48      | 260.092780  | 0.709766      | 0.556994      | 0.692396        | 0.574868

| 49      | 265.858663  | 0.709946      | 0.556520      | 0.692588        | 0.574680

| 50      | 270.869963  | 0.710153      | 0.556127      | 0.692843        | 0.574599

| 51      | 275.941008  | 0.710500      | 0.555625      | 0.692715        | 0.574445

| 52      | 280.868849  | 0.710873      | 0.555153      | 0.691949        | 0.574229

| 53      | 285.871816  | 0.711017      | 0.554923      | 0.692652        | 0.574188

| 54      | 290.835253  | 0.711153      | 0.554589      | 0.692779        | 0.574247

| 55      | 296.425253  | 0.711207      | 0.554390      | 0.692588        | 0.574250

| 56      | 301.962023  | 0.711607      | 0.553888      | 0.692779        | 0.574194

| 57      | 306.911252  | 0.711817      | 0.553618      | 0.692652        | 0.574194

| 58      | 311.734121  | 0.712121      | 0.553229      | 0.692779        | 0.574068

| 59      | 316.525557  | 0.712484      | 0.552803      | 0.692396        | 0.573865

| 60      | 321.565492  | 0.712605      | 0.552493      | 0.692652        | 0.573926

| 61      | 326.397340  | 0.712701      | 0.552296      | 0.692843        | 0.573855

| 62      | 331.896544  | 0.712735      | 0.552188      | 0.692779        | 0.573867

| 63      | 337.672233  | 0.712938      | 0.551965      | 0.692843        | 0.573877
| 64      | 344.054399  | 0.713245      | 0.551494      | 0.693290        | 0.573865

| 65      | 350.544139  | 0.713335      | 0.551367      | 0.693162        | 0.573879

| 66      | 355.470647  | 0.713575      | 0.550948      | 0.692971        | 0.573724

| 67      | 360.253388  | 0.713619      | 0.550777      | 0.693162        | 0.573739

| 68      | 365.092991  | 0.713999      | 0.550291      | 0.692588        | 0.573850

| 69      | 370.457278  | 0.714306      | 0.549817      | 0.692141        | 0.574001

| 70      | 376.104018  | 0.714549      | 0.549359      | 0.692971        | 0.574129

| 71      | 380.907227  | 0.714809      | 0.549039      | 0.692907        | 0.574196

| 72      | 385.968860  | 0.715106      | 0.548775      | 0.692779        | 0.574342

| 73      | 391.108487  | 0.715523      | 0.548208      | 0.693162        | 0.574182

| 74      | 396.664278  | 0.715920      | 0.547652      | 0.693609        | 0.573976

| 75      | 405.421027  | 0.716501      | 0.547125      | 0.694120        | 0.573807

| 76      | 410.157882  | 0.716714      | 0.546826      | 0.694439        | 0.573727

| 77      | 415.260332  | 0.717114      | 0.546393      | 0.693609        | 0.573691

| 78      | 420.387915  | 0.717955      | 0.545794      | 0.694120        | 0.573640

| 79      | 425.337819  | 0.718112      | 0.545469      | 0.693673        | 0.573725

| 80      | 430.251949  | 0.718122      | 0.545393      | 0.693865        | 0.573754

| 81      | 435.414487  | 0.718375      | 0.544974      | 0.693801        | 0.573795

| 82      | 440.471961  | 0.718782      | 0.544535      | 0.693481        | 0.573799

| 83      | 445.623856  | 0.719039      | 0.544100      | 0.693226        | 0.573696
| 84        | 450.701118   | 0.719396          | 0.543548          | 0.692907             | 0.573571

| 85        | 456.314214   | 0.719716          | 0.543196          | 0.693545             | 0.573674

| 86        | 461.257251   | 0.719850          | 0.542991          | 0.693737             | 0.573671

| 87        | 466.218663   | 0.719923          | 0.542844          | 0.693801             | 0.573685

| 88        | 470.929789   | 0.720270          | 0.542394          | 0.694375             | 0.573541

| 89        | 475.879544   | 0.720343          | 0.542196          | 0.694184             | 0.573580

| 90        | 480.925216   | 0.720614          | 0.541870          | 0.695141             | 0.573482

| 91        | 487.447120   | 0.720881          | 0.541561          | 0.695014             | 0.573524

| 92        | 494.186639   | 0.720904          | 0.541435          | 0.695205             | 0.573553

| 93        | 499.130905   | 0.721241          | 0.540946          | 0.695014             | 0.573498

| 94        | 504.665177   | 0.721534          | 0.540485          | 0.694631             | 0.573511

| 95        | 509.164459   | 0.721721          | 0.540150          | 0.695014             | 0.573668

| 96        | 514.038849   | 0.721915          | 0.539905          | 0.694886             | 0.573783

| 97        | 519.635206   | 0.722335          | 0.539512          | 0.694503             | 0.573916

| 98        | 526.408947   | 0.722635          | 0.539059          | 0.694248             | 0.574023

| 99        | 531.175425   | 0.722992          | 0.538681          | 0.694631             | 0.574068

| 100       | 536.971533   | 0.723296          | 0.538177          | 0.695014             | 0.573845

+-----------+--------------+-------------------+-------------------+---------------------+--------------

PROGRESS: Creating a validation set from 5 percent of training data. This may take a while.
          You can set ``validation_set=None`` to disable validation tracking.
```

```
In [8]: model_boosted_tree.show()
```

```
Canvas is accessible via web browser at the URL: http://localhost:52029/index.html
Opening Canvas in default web browser.
```

```
In [9]: predictions = model_boosted_tree.predict(test)
```

```
In [13]: model_boosted_tree.evaluate(test)
```

```
Out[13]: {’accuracy’: 0.6952937280839538,
          ’auc’: 0.9208015979757028,
          ’confusion_matrix’: Columns:
                 target_label        int
                 predicted_label        int
                 count        int

          Rows: 17

          Data:
          +--------------+-----------------+-------+
          | target_label | predicted_label | count |
          +--------------+-----------------+-------+
          |      3       |        2        |   2   |
          |      5       |        1        |   3   |
          |      3       |        4        |   1   |
          |      2       |        1        |   86  |
          |      0       |        2        |   12  |
          |      2       |        2        |   32  |
          |      4       |        0        |   1   |
          |      1       |        0        | 15618 |
          |      3       |        0        |   6   |
          |      3       |        1        |   3   |
          +--------------+-----------------+-------+
          [17 rows x 3 columns]
          Note: Only the head of the SFrame is printed.
          You can use print_rows(num_rows=m, num_columns=n) to print more rows and columns.,
          ’f1_score’: 0.27553175013437753,
          ’log_loss’: 0.5787037434119917,
          ’precision’: 0.35746711339564347,
          ’recall’: 0.2646634589302001,
          ’roc_curve’: Columns:
                 threshold        float
                 fpr        float
                 tpr        float
                 p        int
                 n        int
                 class        int

          Rows: 600006

          Data:
          +-----------+-----+-----+-------+-------+-------+
          | threshold | fpr | tpr |   p   |   n   | class |
          +-----------+-----+-----+-------+-------+-------+
          |    0.0    | 1.0 | 1.0 | 56978 | 60801 |   0   |
          |   1e-05   | 1.0 | 1.0 | 56978 | 60801 |   0   |
          |   2e-05   | 1.0 | 1.0 | 56978 | 60801 |   0   |
          |   3e-05   | 1.0 | 1.0 | 56978 | 60801 |   0   |
          |   4e-05   | 1.0 | 1.0 | 56978 | 60801 |   0   |
          |   5e-05   | 1.0 | 1.0 | 56978 | 60801 |   0   |
          |   6e-05   | 1.0 | 1.0 | 56978 | 60801 |   0   |
          |   7e-05   | 1.0 | 1.0 | 56978 | 60801 |   0   |
          |   8e-05   | 1.0 | 1.0 | 56978 | 60801 |   0   |
          |   9e-05   | 1.0 | 1.0 | 56978 | 60801 |   0   |
          +-----------+-----+-----+-------+-------+-------+
          [600006 rows x 6 columns]
          Note: Only the head of the SFrame is printed.
          You can use print_rows(num_rows=m, num_columns=n) to print more rows and columns.}
```

```
In [15]: model_boosted_tree.summary()
```

```
Class                          : BoostedTreesClassifier

Schema
------
Number of examples             : 299782
Number of feature columns      : 98
Number of unpacked features    : 98
Number of classes              : 6

Settings
--------
Number of trees                : 600
Max tree depth                 : 6
Training time (sec)            : 536.9734
Training accuracy              : 0.7233
Validation accuracy            : 0.695
Training log_loss             : 0.5382
Validation log_loss           : 0.5738
```

```
In [17]: error = 1 - 0.6952937280839538
         print error
```

```
0.304706271916
```

```
In [ ]:
```