DueCredit

automagically collect citations
for the software, methods, and data
used in analysis pipelines

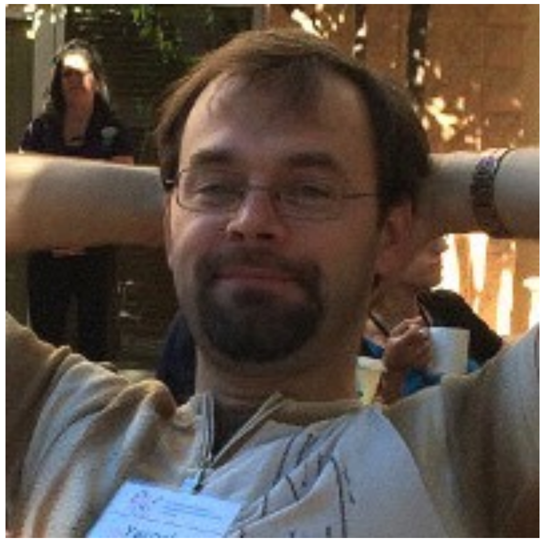
Matteo Visconti di Oleggio Castello, Dartmouth College

Neurohackweek 2016

Why?

- Methods, software, and datasets are not cited adequately
- If cited, version information is often omitted
- Tedious to collect/format references for publications

DueCredit's approach



Make it very easy to

- collect references for methods, software, and data used in the analysis pipeline
- accumulate reference information over time for the entire research project
- keep track of references for methods implemented in libraries
- output references in the desired format (LaTeX + BibTeX, different styles, etc.)

A tiny example

```
# A tiny analysis script to demonstrate duecredit
#
# Import of duecredit is not necessary if you just run this script with
# python -m duecredit
# import duecredit # Just to enable duecredit
from scipy.cluster.hierarchy import linkage
from scipy.spatial.distance import pdist
from sklearn.datasets import make_blobs

print("I: Simulating 4 blobs")
data, true_label = make_blobs(centers=4)

dist = pdist(data, metric=’euclidean’)

Z = linkage(dist, method=’single’)
print("I: Done clustering 4 blobs")
```

A tiny example

```
$> python -m duecredit examples/example_scipy.py
I: Simulating 4 blobs
I: Done clustering 4 blobs

DueCredit Report:
- Scientific tools library / numpy (v 1.10.4) [1]
- Scientific tools library / scipy (v 0.14) [2]
  - Single linkage hierarchical clustering /
    scipy.cluster.hierarchy:linkage (v 0.14) [3]

2 packages cited
0 modules cited
1 function cited

References
----------

[1] Van Der Walt, S., Colbert, S.C. & Varoquaux, G., 2011. The NumPy array: ...
[2] Jones, E. et al., 2001. SciPy: Open source scientific tools for Python.
[3] Sibson, R., 1973. SLINK: an optimally efficient algorithm ...
```

A bigger example

```
$> python -m duecredit /usr/bin/nosetests mvpa2.tests.test_transerror
...

DueCredit Report:
- LIBSVM: A library for support vector machines / libsvm (v None) [1]
- Multivariate pattern analysis of neural data / mvpa2 (v 2.6.0.dev1) [2]
  - Support Vector Machines (SVM) / mvpa2.clfs.SVM (v 2.6.0.dev1) [3]
  - Sparse multinomial-logistic regression classifier / mvpa2.clfs.smlr:SMLR (v 2.6.0.dev1) [4]
  - Bayesian hypothesis testing / mvpa2.clfs.transerror:_call (v 2.6.0.dev1) [5]
  - Recursive feature elimination procedure / mvpa2.featsel.rfe:_train (v 2.6.0.dev1) [6]
  - Searchlight analysis approach / mvpa2.measures.searchlight:_call (v 2.6.0.dev1) [7]
- Scientific tools library / numpy (v 1.10.4) [8]
- Machine Learning library / sklearn (v 0.17.1) [9]
  - Random forest classifiers / sklearn.ensemble.forest:RandomForestClassifier.predict_proba (v 0.17.1) [10]
  - Classification and regression trees / sklearn.tree.tree:DecisionTreeClassifier.predict_proba (v 0.17.1) [11]

4 packages cited
1 module cited
6 functions cited

References
----------

[1] Chang, C.-C. & Lin, C.-J., 2011. LIBSVM. TIST, 2(3), pp.1–27.
[2] Hanke, M. et al., 2009. PyMVPA: a Python Toolbox for Multivariate Pattern Analysis of fMRI Data ...
[3] Vapnik, V., 1995. The Nature of Statistical Learning Theory, New York: Springer.
...
```

A bigger example

```
$> duecredit summary –format=bibtex

@article{Chang_2011,
     doi = {10.1145/1961189.1961199},
     url = {http://dx.doi.org/10.1145/1961189.1961199},
     year = 2011,
     month = {apr},
     publisher = {Association for Computing Machinery ({ACM})},
     volume = {2},
     number = {3},
     pages = {1--27},
     author = {Chih-Chung Chang and Chih-Jen Lin},
     title = {{LIBSVM}},
     journal = {{TIST}}
}
@article{Hanke_2009,
     doi = {10.1007/s12021-008-9041-y},
     url = {http://dx.doi.org/10.1007/s12021-008-9041-y},
     year = 2009,
     month = {jan},
     publisher = {Springer Science $\mathplus$ Business Media},
     volume = {7},
     number = {1},
     pages = {37--53},
     author = {Michael Hanke and Yaroslav O. Halchenko and Per B. Sederberg and Stephen Jos{\'{e}} Hanson
              and James V. Haxby and Stefan Pollmann},
     title = {{PyMVPA}: a Python Toolbox for Multivariate Pattern Analysis of {fMRI} Data},
     journal = {Neuroinform}
}
@Book{Vapnik95:SVM, ...
```

HOWTO 1: In your software

1. Copy `duecredit/stub.py` in your codebase, e.g.,

   ```
   wget -q -O /path/tomodule/yourmodule/due.py \
   https://raw.githubusercontent.com/duecredit/duecredit/master/
   duecredit/stub.py
   ```

2. Then import necessary pieces, e.g.,

   ```
   from .due import due, Doi
   ```

   to provide a reference for the entire module just use

   ```
   due.cite(Doi("1.2.3/x.y.z"), description="Solves all your problems",
            path="magicpy")
   ```

   To provide a reference for a function or method, use the dcite decorator

   ```
   @due.dcite(Doi("1.2.3/x.y.z"), description="Solves some ...")
   def help_me():
        …
   ```

HOWTO 2: Injection

Example: `duecredit/injections/mod_scipy.py`

```
from ..entries import Doi, BibTeX, Url
def inject(injector):
    injector.add(’scipy’, None, BibTeX("""
                 @Misc{JOP+01,
                 ...
                 }"""),
                 description="Scientific tools library",
                 tags=[’implementation’])
    ...
    injector.add(’scipy.cluster.hierarchy’, ’linkage’, BibTeX("""
                 @article{ward1963hierarchical,
                 ...
                 }"""),
                 conditions={(1, ’method’): {’ward’}},
                 description="Ward hierarchical clustering”,
                 min_version=’0.4.3’,
                 tags=[’reference’])
    ...
```

Get involved!

- Use it! :-)
- Report bugs, send pull requests/patches
- Provide support for other languages
    - MATLAB/Octave (https://github.com/duecreditduecredit/issues/20)
    - Java, R, C/C++, … (help wanted!)
- Spread the word : www.duecredit.org