How does Software break?
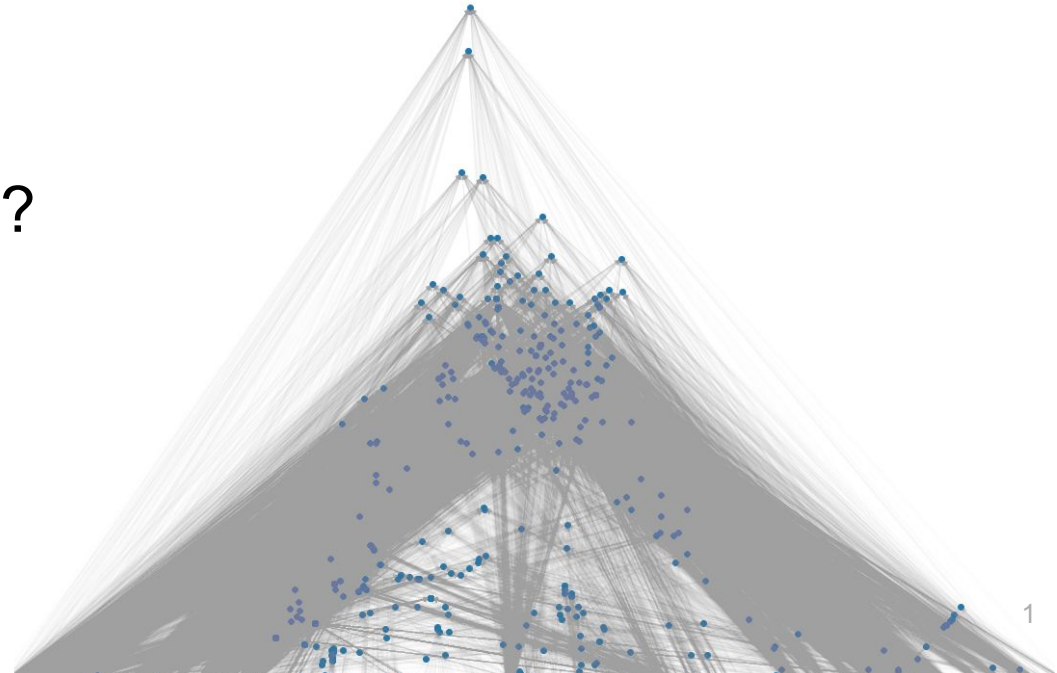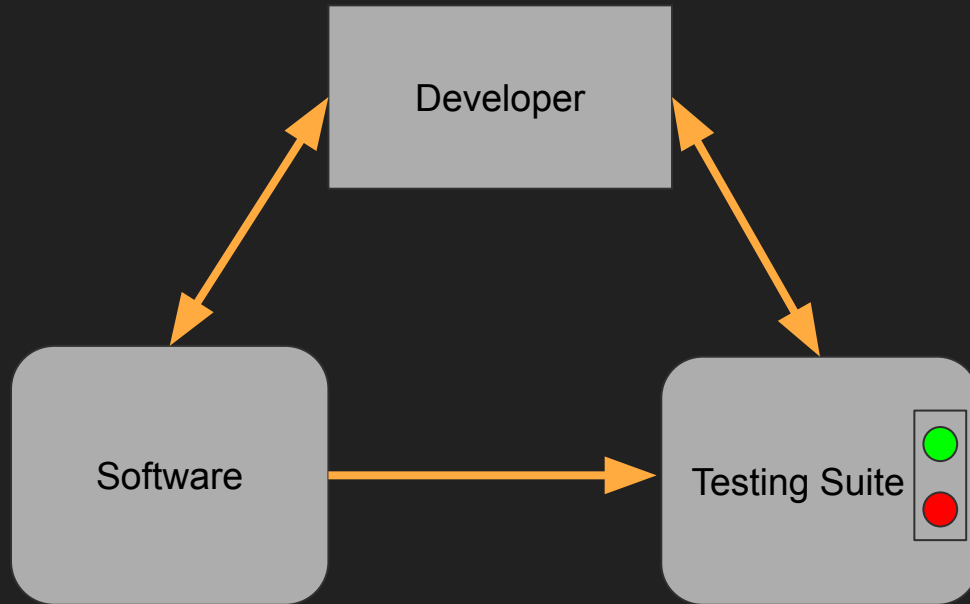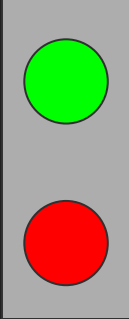
Code Repository Mining Seminar SS20
Software Architecture Chair

Dominik Meier
May 26th, 2020

Why do we test?

Only ?

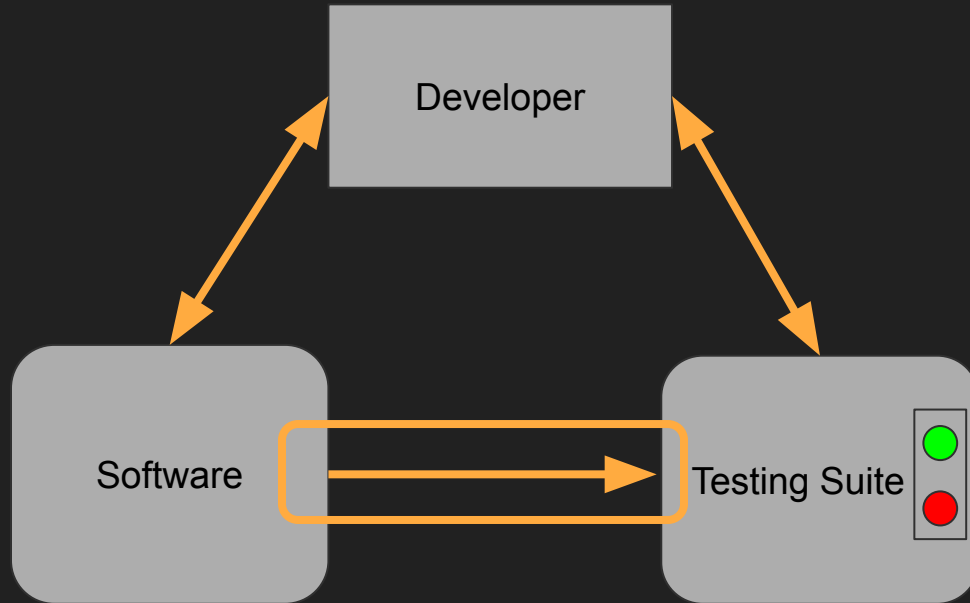Azure Pipelines `succeeded` | build `failing` | codecov `98%` | circleci `passing` | python `3.6 | 3.7 | 3.8` | pypi package `0.23.0`

DOI `10.5281/zenodo.3822537`

scikit-learn

Point of interest

What might we find?

- Find lines that are not tested
  - Improve Test-Quality
- Test Correlation
  - Find hidden dependencies -> use additional information while Coding
- Test-Case Reordering -> Especially evaluation of it
  - Interesting with the rise of Monorepos

Monorepo

From Wikipedia, the free encyclopedia

In revision control systems, a **monorepo** ("mono" from Greek μόνος, mónos, 'single, alone' and "repo" short for repository) is a software development strategy where code for many projects is stored in the same repository. As of 2017 some forms of this software engineering practice were over a decade old, but the general concept had only recently been named.[1] Many attempts have been made to differentiate between monolithic applications and other, newer forms of monorepos. [2][3][4]

Google,[5] Facebook,[6] Microsoft,[7] Uber,[8] Airbnb and Twitter[9] all employ very large monorepos with varying strategies to scale build systems and version control software with a large volume of code and daily changes.

How to gather data?

Mutation Testing

```python
@property
def max_cookie_size(self):
    """Read-only view of the :data:`MAX_COOKIE_SIZE` config key.

    See :attr:`~werkzeug.wrappers.BaseResponse.max_cookie_size` in
    Werkzeug's docs.
    """
    if current_app:
        return current_app.config["MAX_COOKIE_SIZE"]

    # return Werkzeug's default when not in an app context
    return super().max_cookie_size
```

```
======================================= test session starts ========================================
platform linux -- Python 3.8.1, pytest-5.4.2, py-1.8.1, pluggy-0.13.1
rootdir: /mnt/brick/home/dmeier/CRM2020/repos/flask, inifile: setup.cfg, testpaths: tests
plugins: timeout-1.3.4, json-report-1.2.1, json-0.4.0, metadata-1.9.0
collected 468 items

tests/test_appctx.py ..............                                                       [  2%]
tests/test_basic.py ..................................................................... [ 22%]
..............................                                                             [ 29%]
tests/test_blueprints.py ................................................                  [ 39%]
tests/test_cli.py .................................ss.s......                              [ 50%]
tests/test_config.py ....................                                                  [ 53%]
tests/test_converters.py ..                                                                [ 54%]
tests/test_helpers.py ................................................................     [ 71%]
tests/test_instance_config.py .XXX.......                                                  [ 73%]
tests/test_json_tag.py ..............                                                      [ 76%]
tests/test_logging.py ......                                                               [ 77%]
tests/test_regression.py ...                                                               [ 78%]
tests/test_reqctx.py ........ss.....                                                       [ 81%]
tests/test_signals.py sssssss                                                              [ 83%]
tests/test_subclassing.py .                                                               [ 83%]
tests/test_templating.py .............................                                     [ 89%]
tests/test_testing.py ....................s......                                          [ 95%]
tests/test_user_error_handler.py .........                                                 [ 97%]
tests/test_views.py ............                                                           [100%]

=============================== 452 passed, 13 skipped, 3 xpassed in 2.81s ===============================
```
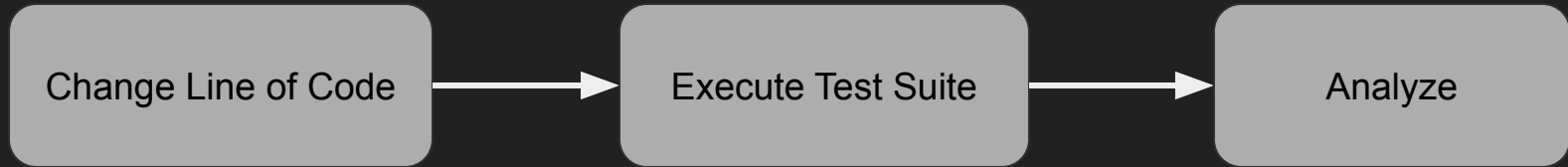
14

```python
@property
def max_cookie_size(self):
    """Read-only view of the :data:`MAX_COOKIE_SIZE` config key.

    See :attr:`~werkzeug.wrappers.BaseResponse.max_cookie_size` in
    Werkzeug's docs.
    """
    if current_app:
        return current_app.config["XXMAX_COOKIE_SIZEXX"]

    # return Werkzeug's default when not in an app context
    return super().max_cookie_size
```
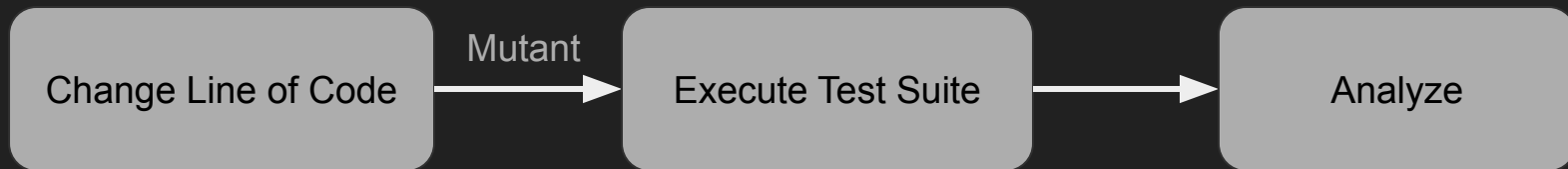
```diff
        if current_app:
-           return current_app.config["MAX_COOKIE_SIZE"]
+           return current_app.config["XXMAX_COOKIE_SIZEXX"]
```

15

```
============================== short test summary info ==============================
FAILED tests/test_basic.py::test_session - KeyError: 'XXMAX_COOKIE_SIZEXX'
FAILED tests/test_basic.py::test_session_using_server_name - KeyError: 'XXMAX_COOKIE_SIZEXX'
FAILED tests/test_basic.py::test_session_using_server_name_and_port - KeyError: 'XXMAX_COOKIE_SIZEXX'
FAILED tests/test_basic.py::test_session_using_server_name_port_and_path - KeyError: 'XXMAX_COOKIE_SIZEXX'
FAILED tests/test_basic.py::test_session_using_application_root - KeyError: 'XXMAX_COOKIE_SIZEXX'
FAILED tests/test_basic.py::test_session_using_session_settings - KeyError: 'XXMAX_COOKIE_SIZEXX'
FAILED tests/test_basic.py::test_session_using_samesite_attribute - KeyError: 'XXMAX_COOKIE_SIZEXX'
FAILED tests/test_basic.py::test_session_localhost_warning - KeyError: 'XXMAX_COOKIE_SIZEXX'
FAILED tests/test_basic.py::test_session_ip_warning - KeyError: 'XXMAX_COOKIE_SIZEXX'
FAILED tests/test_basic.py::test_session_expiration - KeyError: 'XXMAX_COOKIE_SIZEXX'
FAILED tests/test_basic.py::test_session_stored_last - KeyError: 'XXMAX_COOKIE_SIZEXX'
FAILED tests/test_basic.py::test_session_special_types - KeyError: 'XXMAX_COOKIE_SIZEXX'
FAILED tests/test_basic.py::test_session_cookie_setting - KeyError: 'XXMAX_COOKIE_SIZEXX'
FAILED tests/test_basic.py::test_session_vary_cookie - KeyError: 'XXMAX_COOKIE_SIZEXX'
FAILED tests/test_basic.py::test_extended_flashing - KeyError: 'XXMAX_COOKIE_SIZEXX'
FAILED tests/test_basic.py::test_max_cookie_size - KeyError: 'XXMAX_COOKIE_SIZEXX'
FAILED tests/test_helpers.py::TestStreaming::test_stream_keeps_session - KeyError: 'XXMAX_COOKIE_SIZEXX'
FAILED tests/test_reqctx.py::test_session_dynamic_cookie_name - AssertionError: assert b'<!DOCTYPE H...cation.</p>\n' ...
FAILED tests/test_templating.py::test_standard_context - KeyError: 'XXMAX_COOKIE_SIZEXX'
FAILED tests/test_testing.py::test_redirect_keep_session - KeyError: 'XXMAX_COOKIE_SIZEXX'
FAILED tests/test_testing.py::test_session_transactions - KeyError: 'XXMAX_COOKIE_SIZEXX'
ERROR tests/test_helpers.py::TestStreaming::test_stream_keeps_session - AssertionError: assert [<RequestCont...f flask...
========================== 21 failed, 431 passed, 13 skipped, 3 xpassed, 1 error in 5.55s ==========================
```

16

Flask

Change Line of Code → Mutant → Execute Test Suite → Analyze
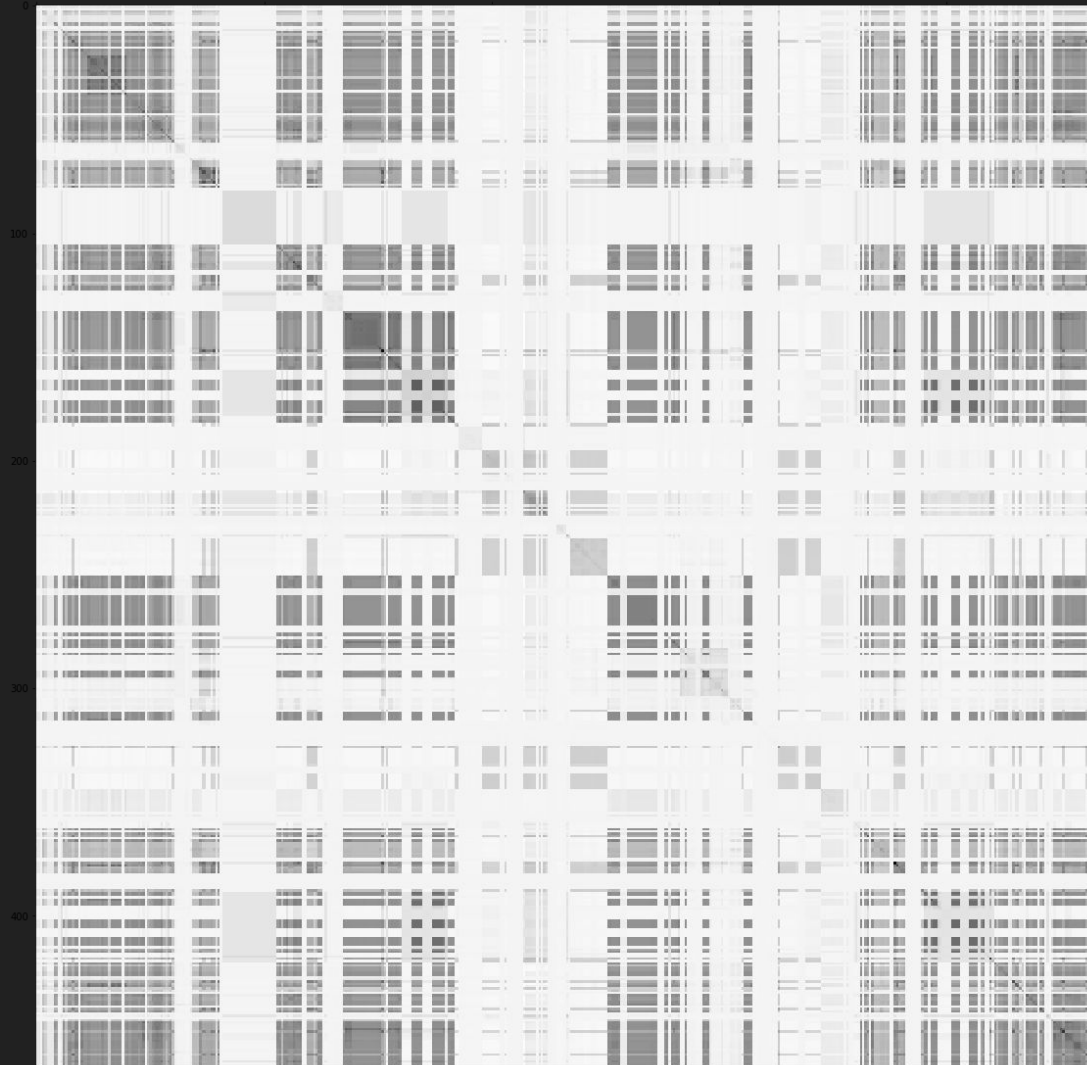
```
-    self.instance_path = instance_path
+    self.instance_path = None

-        if index is None:
+        if index is not None:

if current_app:
-    return current_app.config["MAX_COOKIE_SIZE"]
+    return current_app.config["XXMAX_COOKIE_SIZEXX"]
```
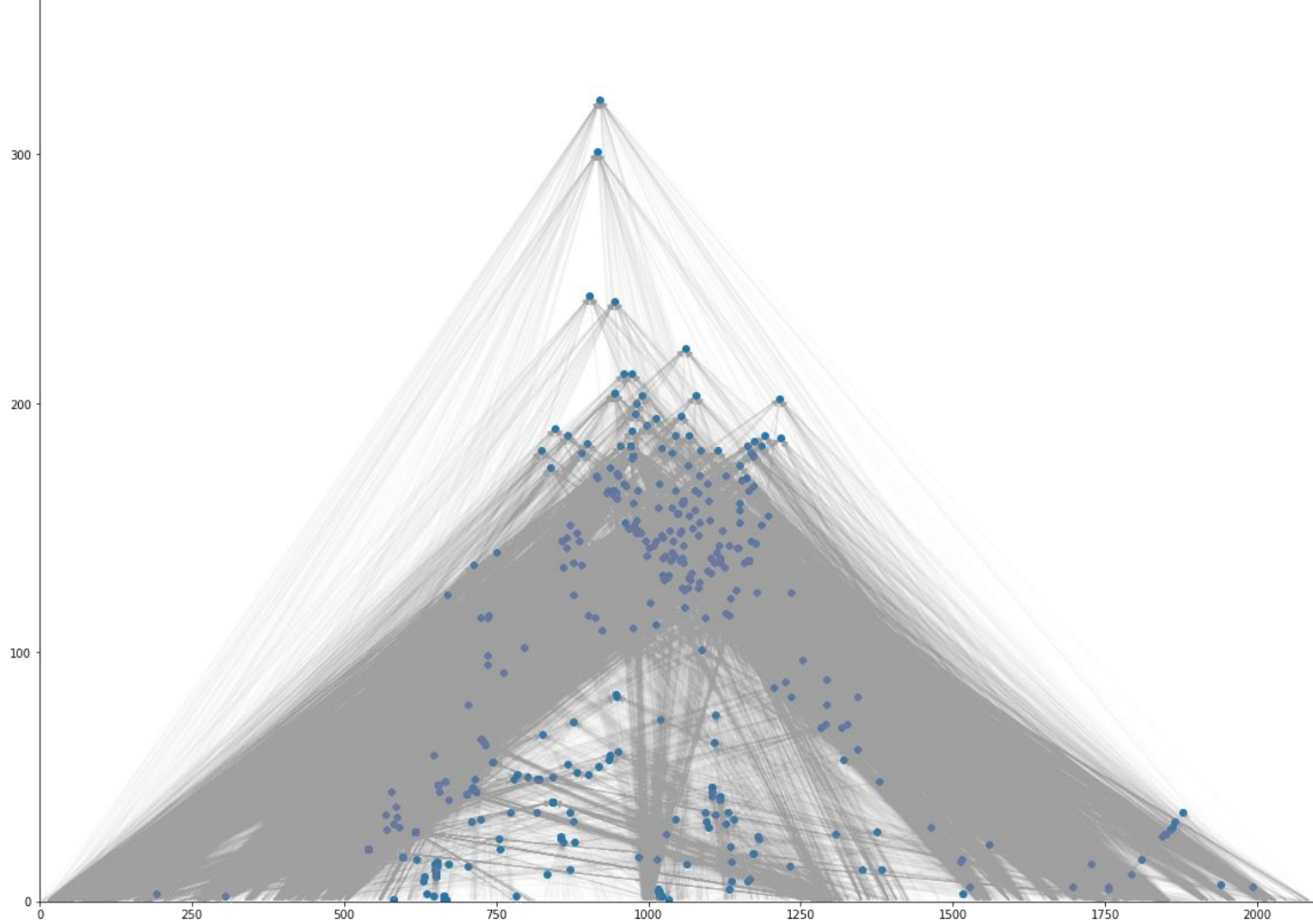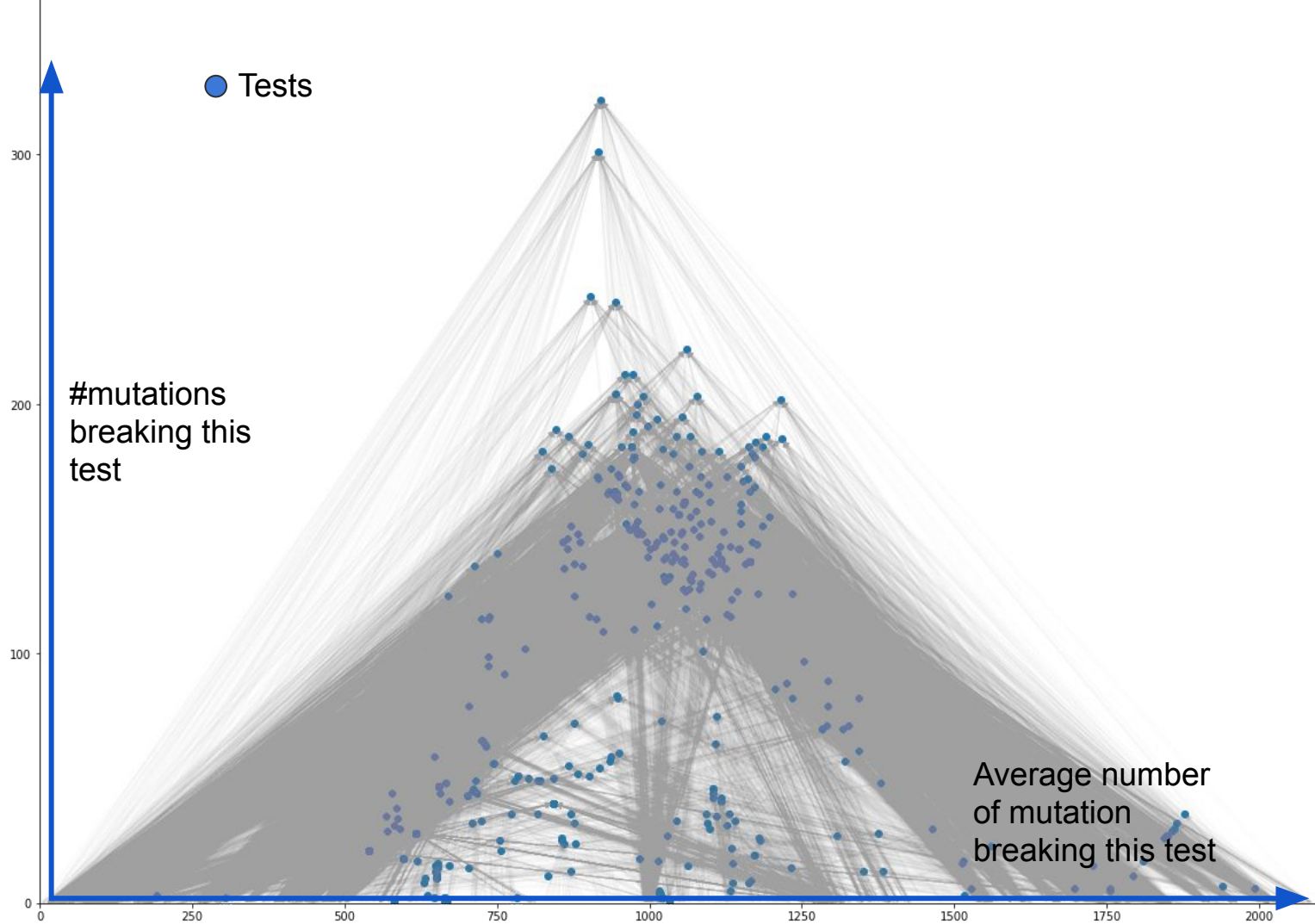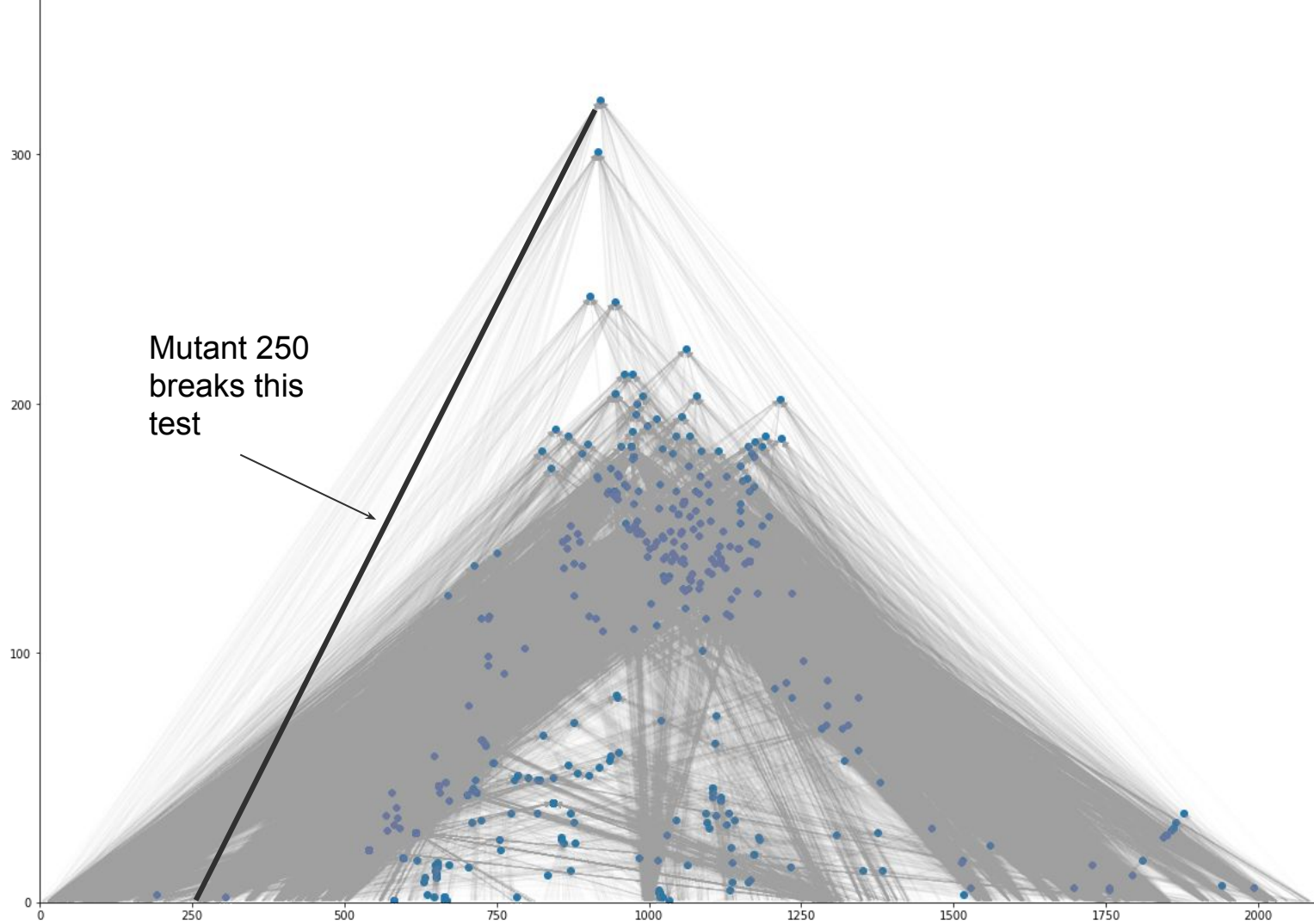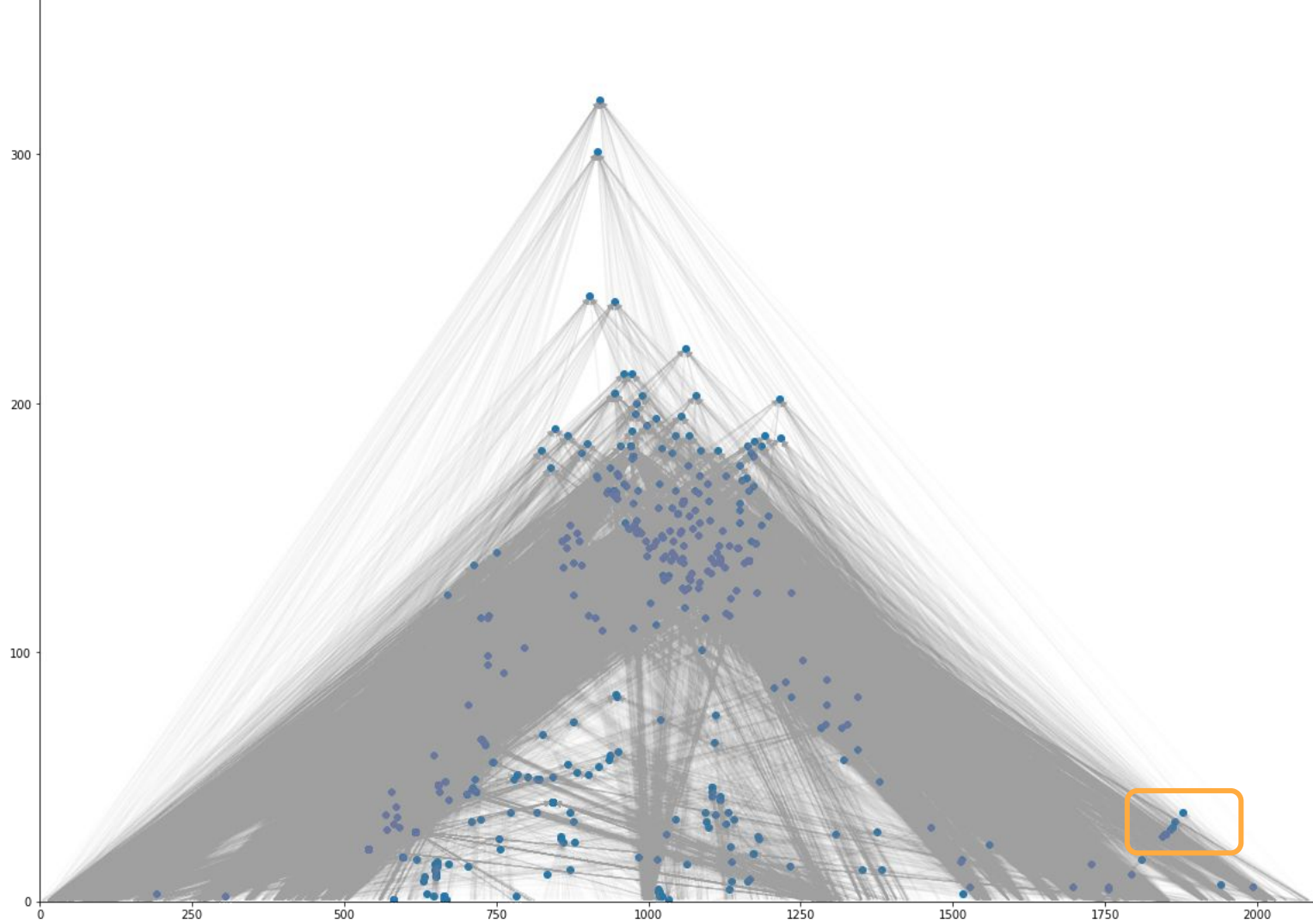
pytest

jupyter

~ 500 Tests
~ 2000 Mutations

Plotted Covariance
Matrix of the tests

19

Tests

#mutations breaking this test

Average number of mutation breaking this test

21

Mutant 250 breaks this test

22

23

```
--- a/src/flask/json/tag.py
+++ b/src/flask/json/tag.py
@@ -264,7 +264,7 @@ class TaggedJSONSerializer:
                 not true.
            """

            tag = tag_class(self)
-           key = tag.key
+           key = None

            if key is not None:
                if not force and key in self.tags:
```

```
============================= short test summary info =============================
FAILED tests/test_basic.py::test_extended_flashing - AssertionError: assert [{' t': ['mes...ting</em>']}] == [('message', ...sting</em>'))]
FAILED tests/test_json_tag.py::test_dump_load_unchanged[data0] - AssertionError: assert {' t': {' t': [1, 2, 3]}} == {' t': (1, 2, 3)}
FAILED tests/test_json_tag.py::test_dump_load_unchanged[data1] - AssertionError: assert {' t__': {' b': 'YQ=='}} == {' t__': b'a'}
FAILED tests/test_json_tag.py::test_dump_load_unchanged[data3] - AssertionError: assert {'x': {' t': ..., 3]}, 'y': 4} == {'x': (1, 2, 3), 'y': 4}
FAILED tests/test_json_tag.py::test_dump_load_unchanged[data4] - AssertionError: assert {' t': [1, 2, 3]} == (1, 2, 3)
FAILED tests/test_json_tag.py::test_dump_load_unchanged[data5] - AssertionError: assert [{' t': [1, 2, 3]}] == [(1, 2, 3)]
FAILED tests/test_json_tag.py::test_dump_load_unchanged[\xff] - AssertionError: assert {' b': '/w=='} == b'\xff'
FAILED tests/test_json_tag.py::test_dump_load_unchanged[<html>] - assert {' m': '<html>'} == Markup('<html>')
FAILED tests/test_json_tag.py::test_dump_load_unchanged[data8] - assert {' u': '987947d1d7584fad8d3a57724afd75c2'} == UUID('987947d1-d758-4fad-8d3a-57724afd75c2')
FAILED tests/test_json_tag.py::test_dump_load_unchanged[data9] - assert {' d': 'Tue, 26 May 2020 11:14:24 GMT'} == datetime.datetime(2020, 5, 26, 11, 14, 24)
FAILED tests/test_json_tag.py::test_duplicate_tag - Failed: DID NOT RAISE <class 'KeyError'>
FAILED tests/test_json_tag.py::test_custom_tag - AttributeError: 'dict' object has no attribute 'data'
============================= 12 failed, 440 passed, 13 skipped, 3 xpassed in 3.17s =============================
```

```
--- a/src/flask/json/tag.py
+++ b/src/flask/json/tag.py
@@ -264,7 +264,7 @@ class TaggedJSONSerializer:
                not true.
            """

            tag = tag_class(self)
-           key = tag.key
+           key = None

            if key is not None:
                if not force and key in self.tags:
```

```
======================================= short test summary info ==========================================================
FAILED tests/test_basic.py::test_extended_flashing - AssertionError: assert [{' t': ['mes...ting</em>']}] == [('message', ...sting</em>'))]
FAILED tests/test_json_tag.py::test_dump_load_unchanged[data0] - AssertionError: assert {' t': {' t': [1, 2, 3]}} == {' t': (1, 2, 3)}
FAILED tests/test_json_tag.py::test_dump_load_unchanged[data1] - AssertionError: assert {' t__': {' b': 'YQ=='}} == {' t__': b'a'}
FAILED tests/test_json_tag.py::test_dump_load_unchanged[data3] - AssertionError: assert {'x': {' t': ..., 3]}, 'y': 4} == {'x': (1, 2, 3), 'y': 4}
FAILED tests/test_json_tag.py::test_dump_load_unchanged[data4] - AssertionError: assert {' t': [1, 2, 3]} == (1, 2, 3)
FAILED tests/test_json_tag.py::test_dump_load_unchanged[data5] - AssertionError: assert [{' t': [1, 2, 3]}] == [(1, 2, 3)]
FAILED tests/test_json_tag.py::test_dump_load_unchanged[\xff] - AssertionError: assert {' b': '/w=='} == b'\xff'
FAILED tests/test_json_tag.py::test_dump_load_unchanged[<html>] - AssertionError: assert {' m': '<html>'} == Markup('<html>')
FAILED tests/test_json_tag.py::test_dump_load_unchanged[data8] - assert {' u': '987947d1d7584fad8d3a57724afd75c2'} == UUID('987947d1-d758-4fad-8d3a-57724afd75c2')
FAILED tests/test_json_tag.py::test_dump_load_unchanged[data9] - assert {' d': 'Tue, 26 May 2020 11:14:24 GMT'} == datetime.datetime(2020, 5, 26, 11, 14, 24)
FAILED tests/test_json_tag.py::test_duplicate_tag - Failed: DID NOT RAISE <class 'KeyError'>
FAILED tests/test_json_tag.py::test_custom_tag - AttributeError: 'dict' object has no attribute 'data'
============================================== 12 failed, 440 passed, 13 skipped, 3 xpassed in 3.17s ======================
```

Examples of outcomes

- Visualization of Correlation
- Test Classification / Test Evaluation
- Model to predict failing tests based on gathered data
- Evaluation of different test reordering models (APFD / other Metric)

Challenges

- Mutation Testing not widely used