Painless machine learning in production

H. Chase Stevens
Principal Data Science Engineer, teikametrics
Boston, MA
chase@chasestevens.com
@hchasestevens

Europython 2020

“Painless machine learning in production”

“Painless machine learning in production”

~~“Painless **machine learning** in production”~~

“Painless machine learning in production”

~~“Painless **machine learning** in production”~~

“Painless machine learning in **production**”

“Painless machine learning in production”

~~“Painless **machine learning** in production”~~

“Painless machine learning in **production**”

“*Painless* machine learning in production”

“Painless machine learning in production”

~~“Painless **machine learning** in production”~~

“Painless machine learning in **production**”

“*Painless* machine learning in production”

“Painless machine learning in production”

~~“Painless **machine learning** in production”~~

“Painless machine learning in **production**”

“*Painless* machine learning in production”

“Painless machine learning in production”

~~“Painless **machine learning** in production”~~

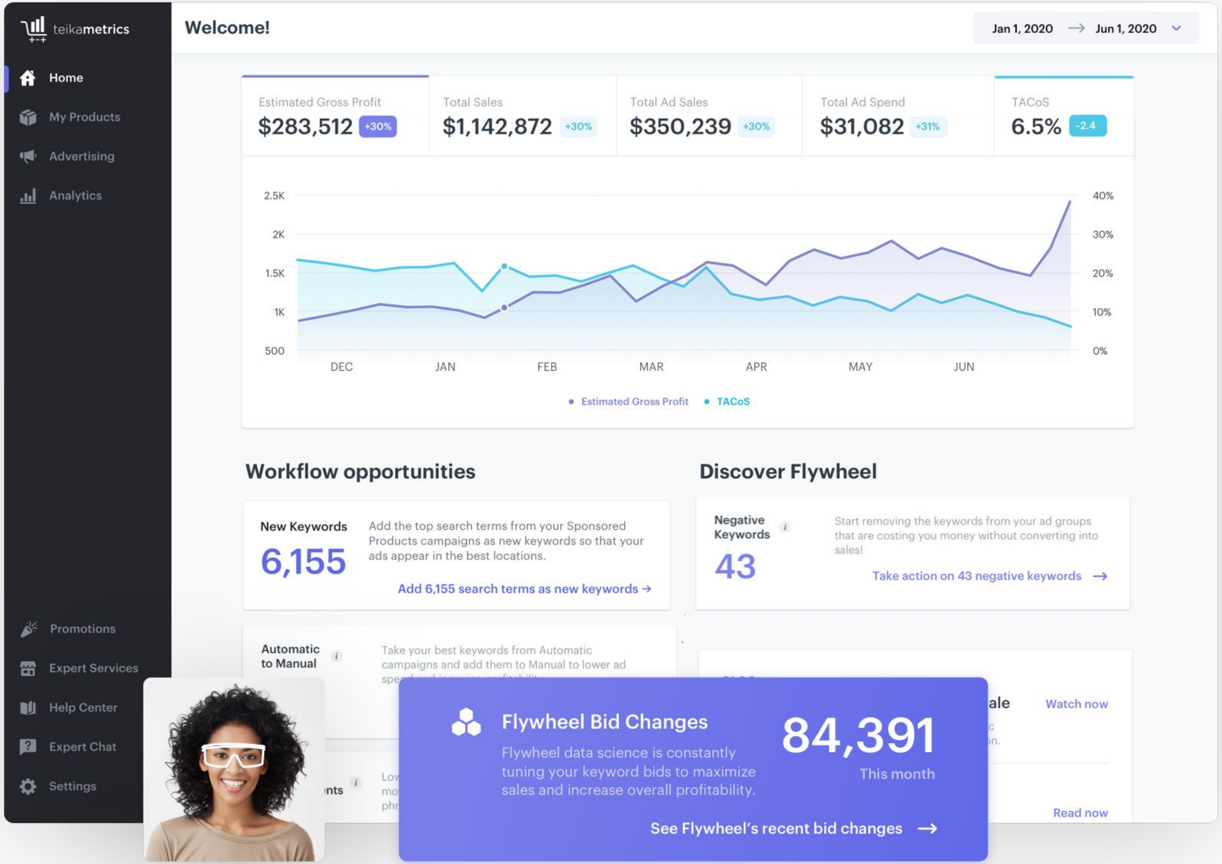
“Painless machine learning in **production**”

“*Painless* machine learning in production”

Lessons from industry regarding pain reduction and data scientist empowerment in the productionization of machine learning models

H. Chase Stevens
Principal Data Science Engineer, teikametrics
Boston, MA
chase@chasestevens.com
@hchasestevens

teikametrics



teikametrics

Home

My Products

Advertising

Analytics

Promotions

Expert Services

Help Center

Expert Chat

Settings

Welcome!

Jan 1, 2020 → Jun 1, 2020

Estimated Gross Profit	Total Sales	Total Ad Sales	Total Ad Spend	TACoS
$283,512 +30%	$1,142,872 +30%	$350,239 +30%	$31,082 +31%	6.5% -2.4

Workflow opportunities

New Keywords

6,155

Add the top search terms from your Sponsored Products campaigns as new keywords so that your ads appear in the best locations.

Add 6,155 search terms as new keywords →

Automatic to Manual

Take your best keywords from Automatic campaigns and add them to Manual to lower ad spe

nts

Lov mo phr

Discover Flywheel

Negative Keywords

43

Start removing the keywords from your ad groups that are costing you money without converting into sales!

Take action on 43 negative keywords →

ale

Watch now

Read now

Flywheel Bid Changes

Flywheel data science is constantly tuning your keyword bids to maximize sales and increase overall profitability.

84,391

This month

See Flywheel's recent bid changes →

Contents

- Motivation
- Developer experience
- Our stack
- Lessons learned

Motivation

I. Ops is intrinsic to ML

Motivation

I. Ops is intrinsic to ML

II. MLOps is unsustainable

Motivation

I. Ops is intrinsic to ML

II. MLOps is unsustainable

∴

Data scientists need to productionize their own models

Motivation

I. Ops is intrinsic to ML

II. MLOps is unsustainable

∴

Data scientists need to productionize their own models

III. Data scientists want to do data science

Motivation

I. Ops is intrinsic to ML

II. MLOps is unsustainable

∴

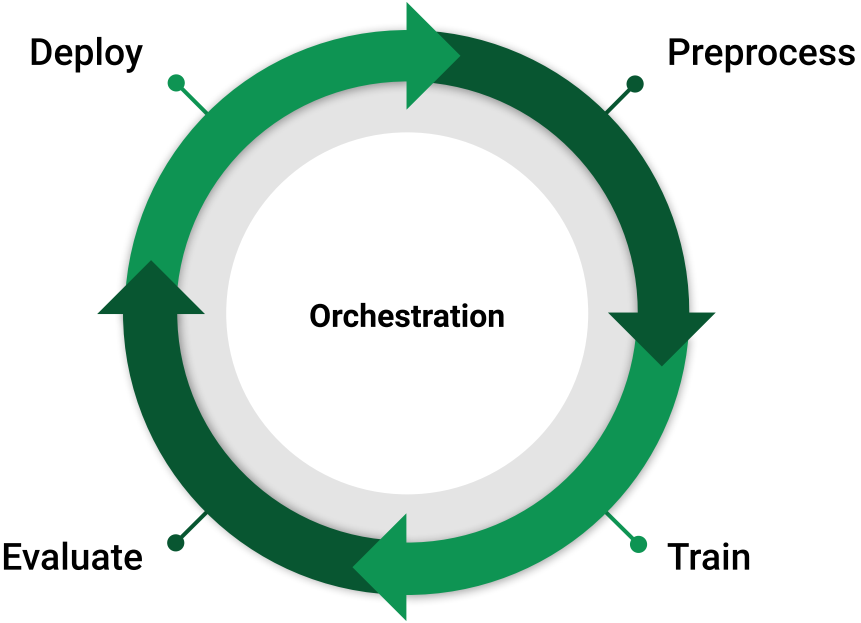
Data scientists need to productionize their own models

III. Data scientists want to do data science

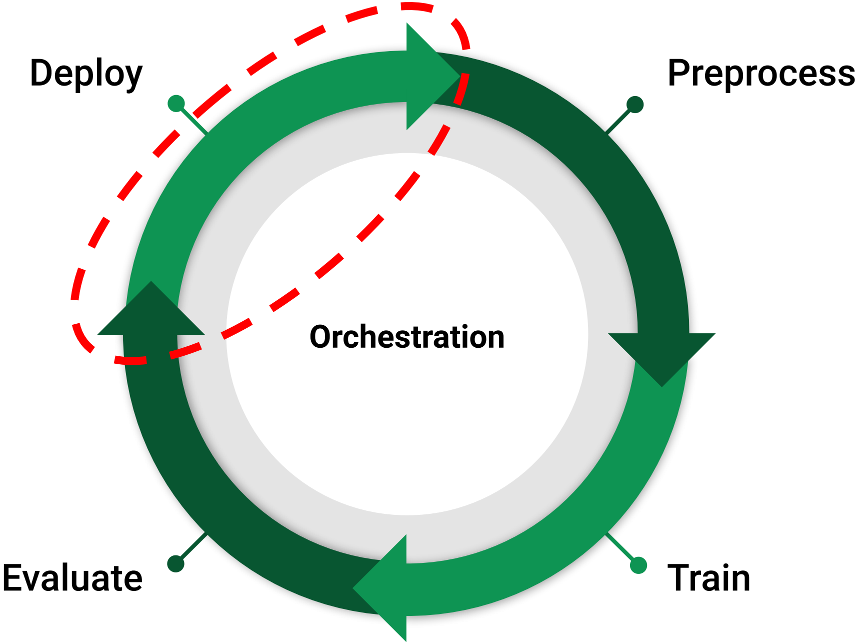
∴

We need tooling and services to minimize “ops” overhead

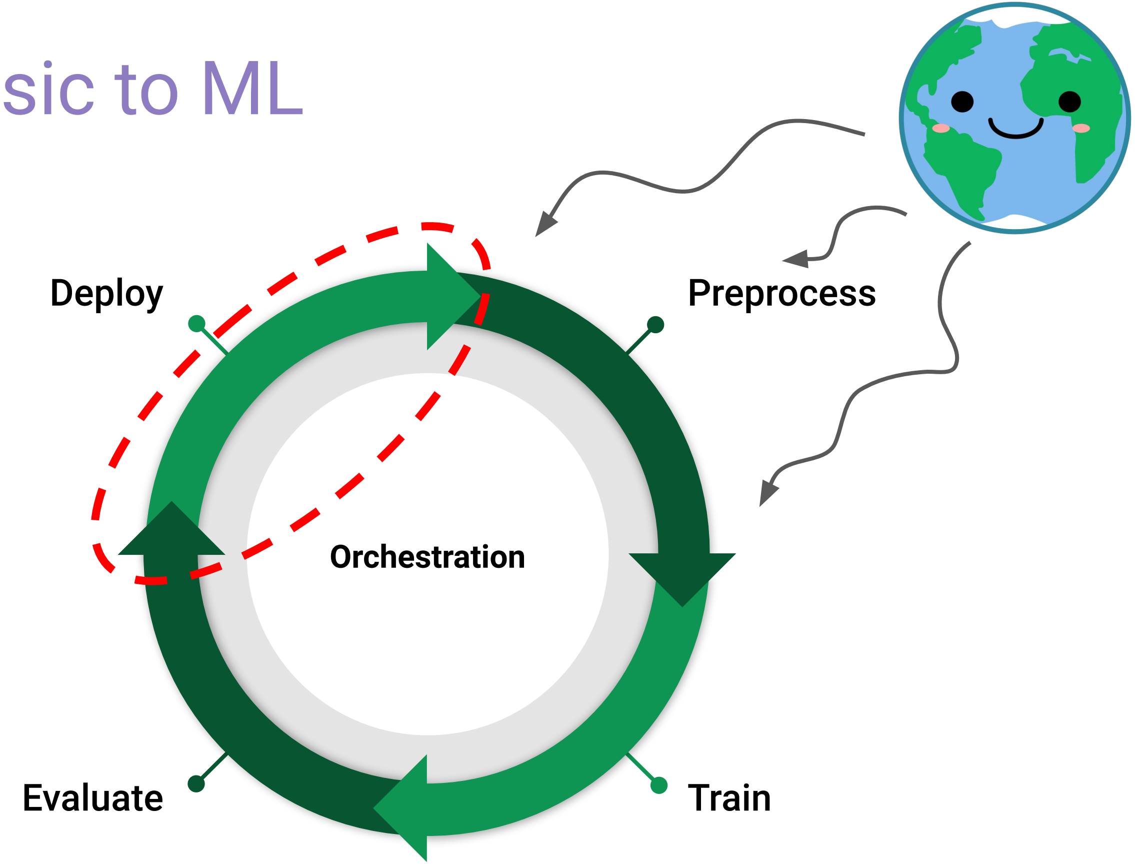
I. Ops is intrinsic to ML



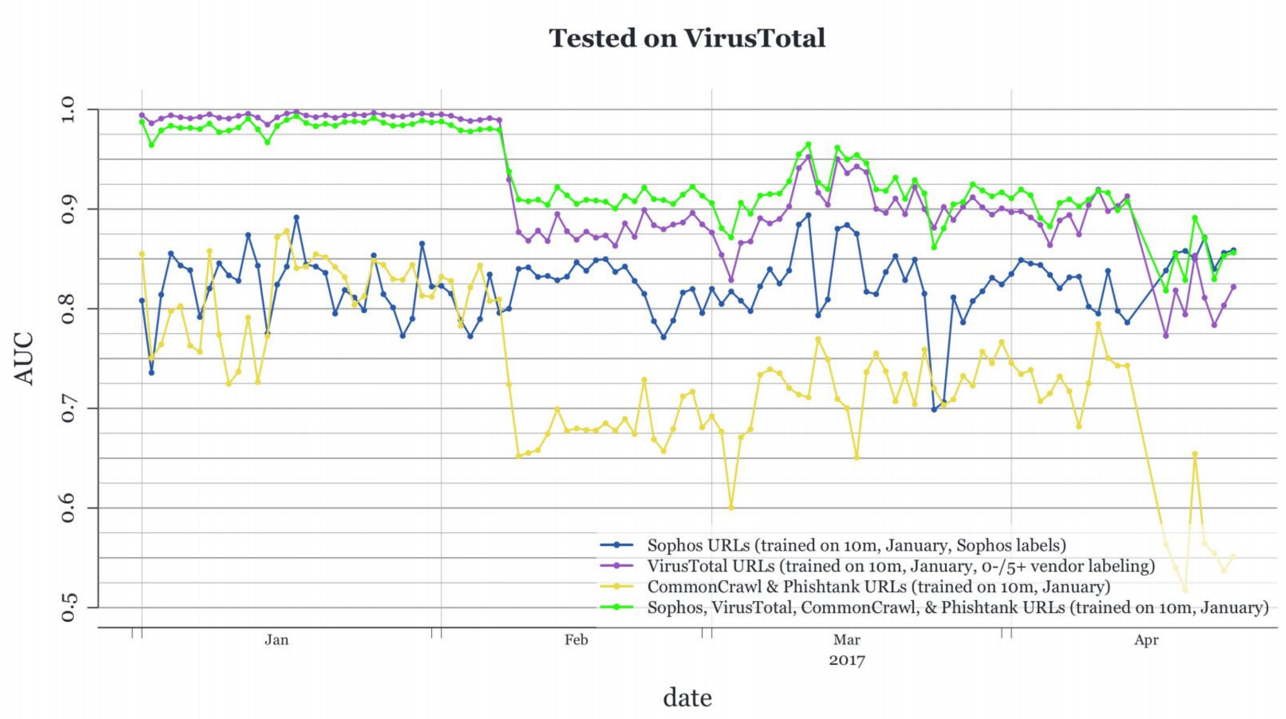
I. Ops is intrinsic to ML



I. Ops is intrinsic to ML

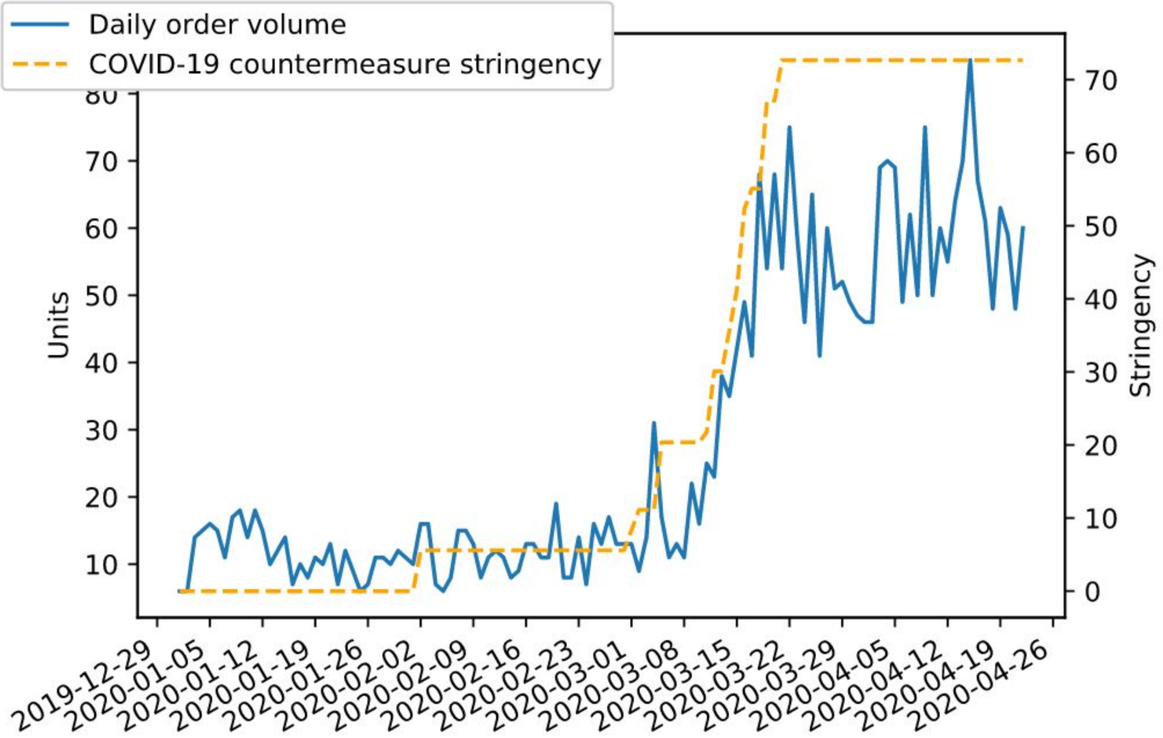


I. Ops is intrinsic to ML

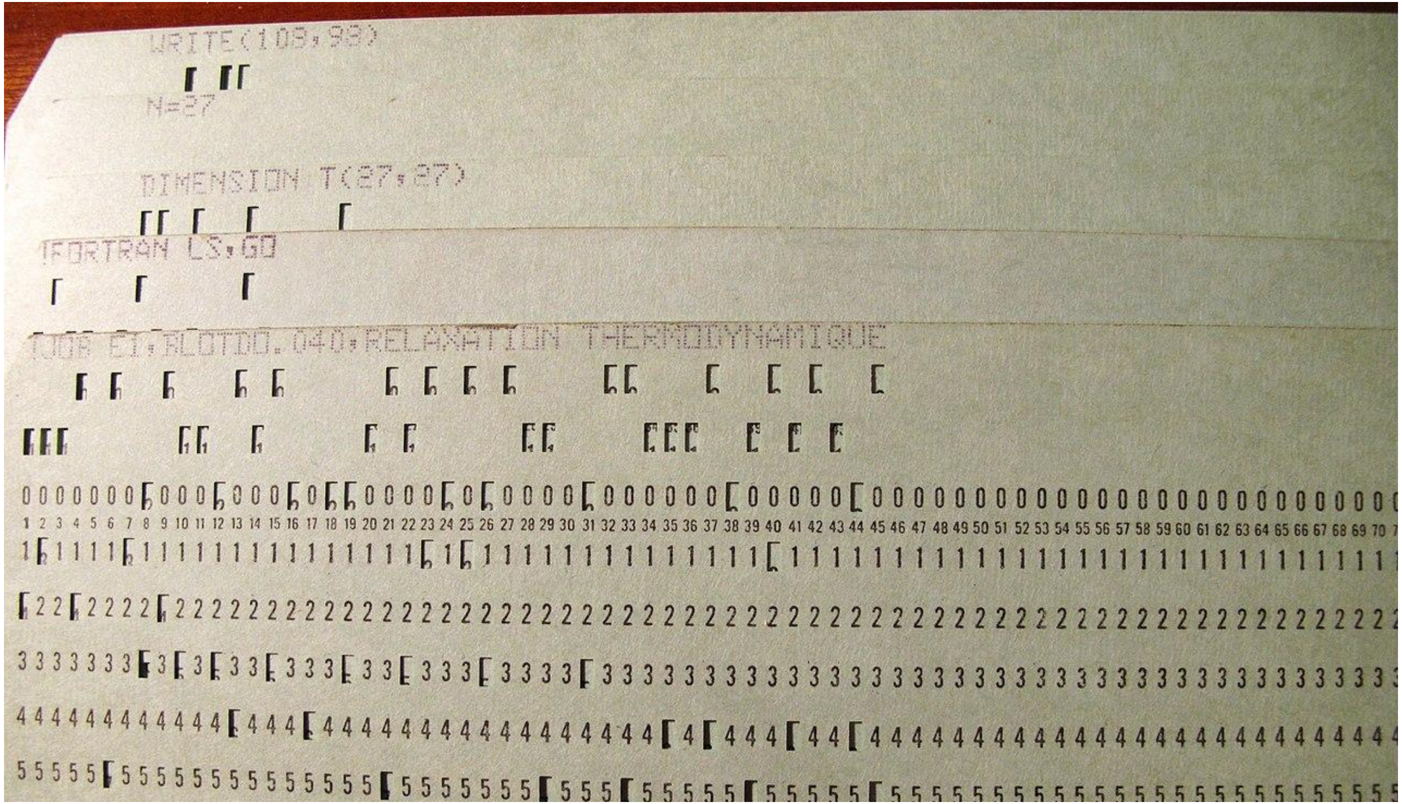


Sanders, H., & Saxe, J. (2017). Garbage in, garbage out: how purportedly great ML models can be screwed up by bad data.

I. Ops is intrinsic to ML

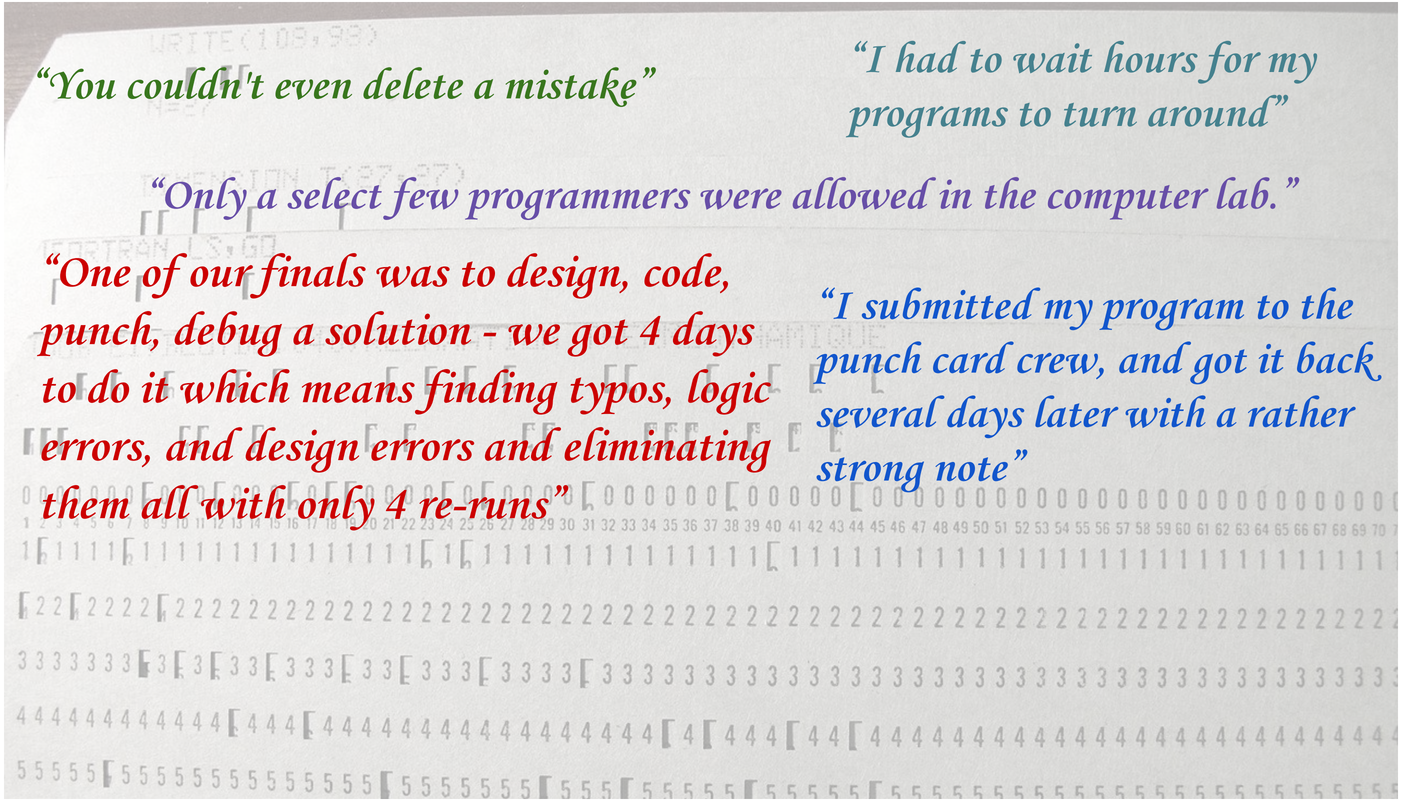


II. MLOps is unsustainable (in 1970)



```
WRITE(108,98)
N=27
DIMENSION T(27,27)
!FORTRAN LS,GO
!JOB E1,BLOTDO.040,RELAXATION THERMODYNAMIQUE
```

II. MLOps is unsustainable (in 1970)



"You couldn't even delete a mistake"

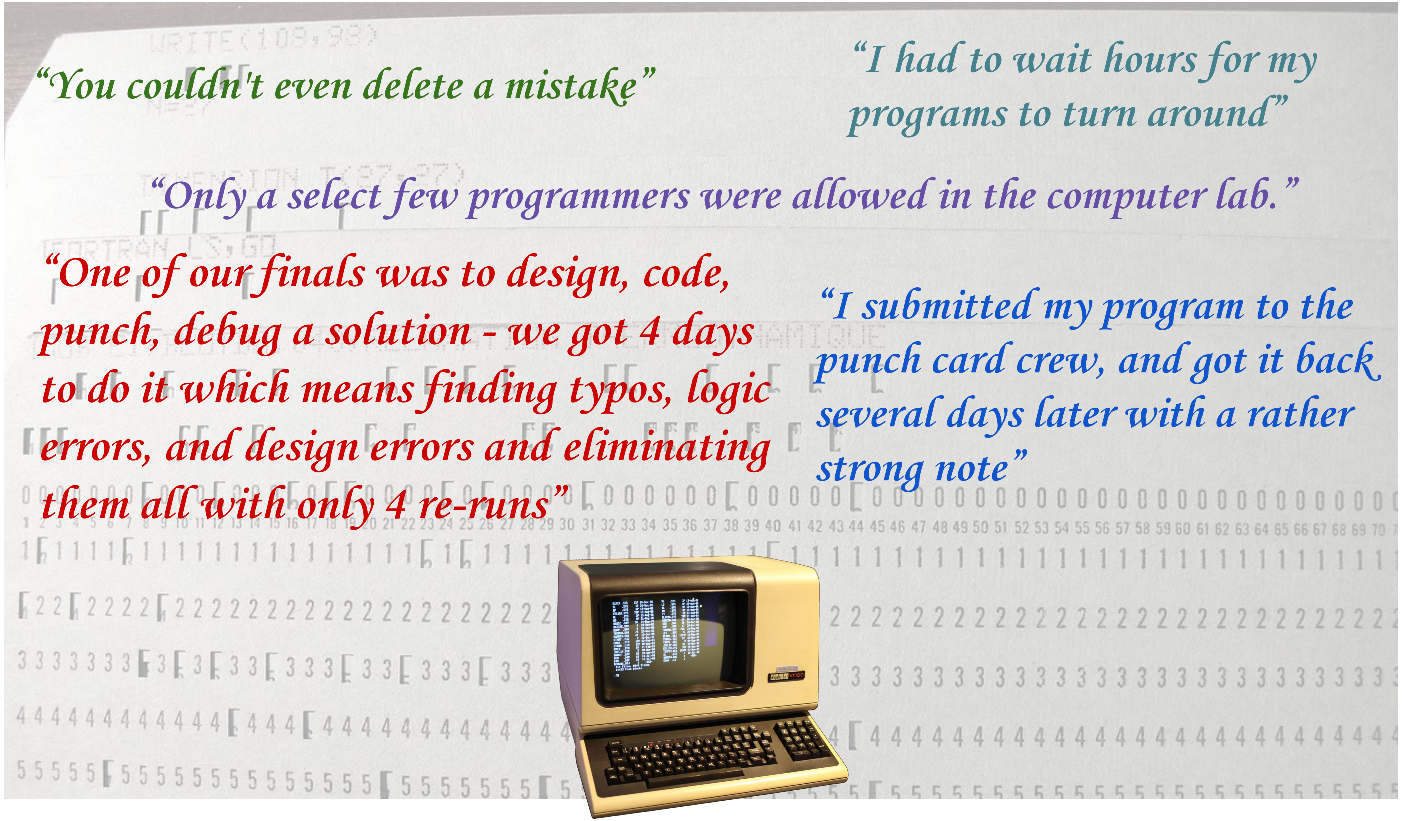
"I had to wait hours for my programs to turn around"

"Only a select few programmers were allowed in the computer lab."

"One of our finals was to design, code, punch, debug a solution - we got 4 days to do it which means finding typos, logic errors, and design errors and eliminating them all with only 4 re-runs"

"I submitted my program to the punch card crew, and got it back several days later with a rather strong note"

II. MLOps is unsustainable (in 1970)



“You couldn't even delete a mistake”

“I had to wait hours for my programs to turn around”

“Only a select few programmers were allowed in the computer lab.”

“One of our finals was to design, code, punch, debug a solution - we got 4 days to do it which means finding typos, logic errors, and design errors and eliminating them all with only 4 re-runs”

“I submitted my program to the punch card crew, and got it back several days later with a rather strong note”

II. MLOps is unsustainable (in 2000)

Code → QA → Release (?)

II. MLOps is unsustainable (in 2000)

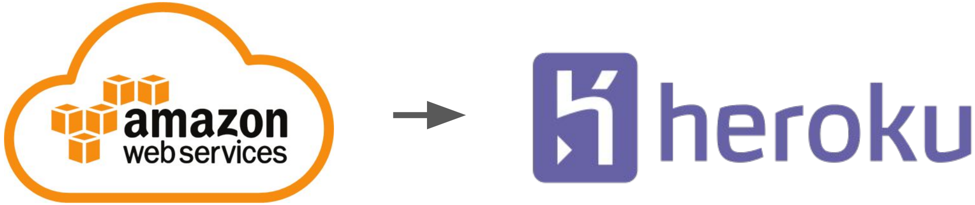
Code → QA → ~~Release (?)~~

II. MLOps is unsustainable (in 2000)

Code → QA → ~~Release (?)~~

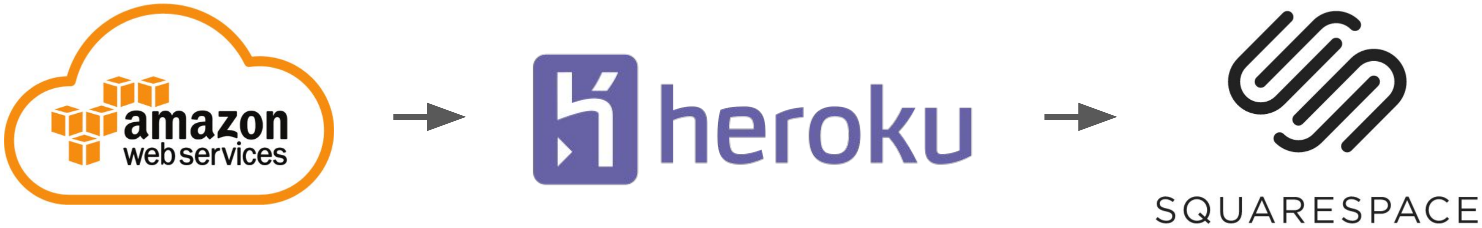
II. MLOps is unsustainable (in 2000)

Code → QA → ~~Release (?)~~



II. MLOps is unsustainable (in 2000)

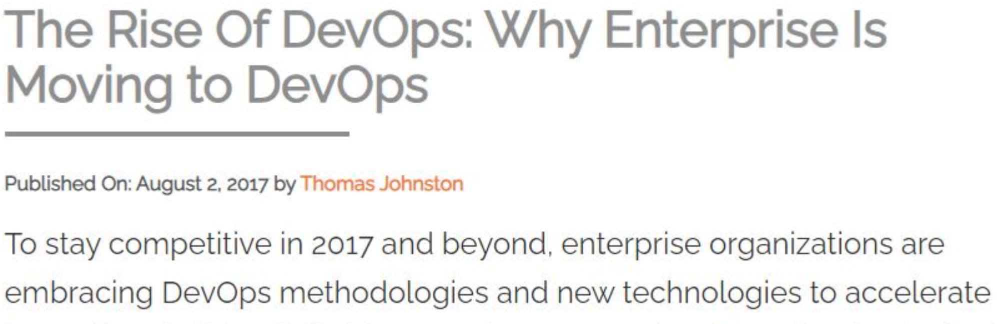
Code → QA → ~~Release (?)~~



II. MLOps is unsustainable (in 2000)

Code → QA → ~~Release (?)~~

amazon web services → heroku → SQUARESPACE



The Rise Of DevOps: Why Enterprise Is Moving to DevOps

Published On: August 2, 2017 by Thomas Johnston

To stay competitive in 2017 and beyond, enterprise organizations are embracing DevOps methodologies and new technologies to accelerate

II. MLOps is unsustainable (today)

“Here’s the model”

II. MLOps is unsustainable (today)

“Here’s the model”

“This data isn’t available yet”

II. MLOps is unsustainable (today)

“Here’s the model”

“This data isn’t available yet”

“Try this instead”

II. MLOps is unsustainable (today)

“Here’s the model”

“Try this instead”

“This data isn’t available yet”

“Wrong version of numpy”

II. MLOps is unsustainable (today)

“Here’s the model”

“This data isn’t available yet”

“Try this instead”

“Wrong version of numpy”

“That should be corrected”

II. MLOps is unsustainable (today)

“Here’s the model”

“Try this instead”

“That should be corrected”

“This data isn’t available yet”

“Wrong version of numpy”

“This null value isn’t handled”

II. MLOps is unsustainable (today)

“Here’s the model”

“Try this instead”

“That should be corrected”

“Try again?”

“This data isn’t available yet”

“Wrong version of numpy”

“This null value isn’t handled”

II. MLOps is unsustainable (today)

“Here’s the model”

“Try this instead”

“That should be corrected”

“Try again?”

“This data isn’t available yet”

“Wrong version of numpy”

“This null value isn’t handled”

“The graphs aren’t displaying”

II. MLOps is unsustainable (today)

“Here’s the model”

“Try this instead”

“That should be corrected”

“Try again?”

“OK, delete that part”

“This data isn’t available yet”

“Wrong version of numpy”

“This null value isn’t handled”

“The graphs aren’t displaying”

II. MLOps is unsustainable (today)

“Here’s the model”

“Try this instead”

“That should be corrected”

“Try again?”

“OK, delete that part”

“This data isn’t available yet”

“Wrong version of numpy”

“This null value isn’t handled”

“The graphs aren’t displaying”

“This takes too long in prod”

II. MLOps is unsustainable (today)

“Here’s the model”

“Try this instead”

“That should be corrected”

“Try again?”

“OK, delete that part”

“... Ready to try version two?”

“This data isn’t available yet”

“Wrong version of numpy”

“This null value isn’t handled”

“The graphs aren’t displaying”

“This takes too long in prod”

Developer experience

```
$ cookiecutter git@github.com:teikametrics/sagemaker-framework.git
github_username [my-github-username]: hchasestevens
project_name [my-sagemaker-model]: europython-example-model
project_slug [europython_example_model]:
model_name [europython-example-model]:
description [An ML model living on the SageMaker platform.]: An example model for
Europython 2020.
Select model_validation_metric:
1 - sklearn.metrics.mean_squared_error
2 - sklearn.metrics.r2_score
3 - sklearn.metrics.accuracy_score
4 - sklearn.metrics.log_loss
5 - sklearn.metrics.f1_score
6 - sagemaker_framework.utils.metrics.mean_absolute_percentage_error
Choose from 1, 2, 3, 4, 5, 6 (1, 2, 3, 4, 5, 6) [1]: 1
Select promotion_criterion:
1 - sagemaker_framework.utils.promotion.maximize
2 - sagemaker_framework.utils.promotion.minimize
3 - sagemaker_framework.utils.promotion.maximize_with_tol
4 - sagemaker_framework.utils.promotion.minimize_with_tol
5 - sagemaker_framework.utils.promotion.manual
6 - sagemaker_framework.utils.promotion.always_promote
Choose from 1, 2, 3, 4, 5, 6 (1, 2, 3, 4, 5, 6) [1]: 6
preprocessing_cpus [1]:
preprocessing_memory_in_gb [4]: 8
test_proportion [0.2]: 0.1
training_cpus [1]:
training_memory_in_gb [4]:
training_volume_size_in_gb [2]:
max_training_runtime_in_minutes [30]: 60
min_serving_instances [1]:
max_serving_instances [10]: 1
```

Developer experience

```
$ cookiecutter git@github.com:teikametrics/sagemaker-framework.git
github_username [my-github-username]: hchasestevens
project_name [my-sagemaker-model]: europython-example-model
project_slug [europython_example_model]:
model_name [europython-example-model]:
description [An ML model living on the SageMaker platform.]: An example model for
Europython 2020.
Select model_validation_metric:
1 - sklearn.metrics.mean_squared_error
2 - sklearn.metrics.r2_score
3 - sklearn.metrics.accuracy_score
4 - sklearn.metrics.log_loss
5 - sklearn.metrics.f1_score
6 - sagemaker_framework.utils.metrics.mean_absolute_percentage_error
Choose from 1, 2, 3, 4, 5, 6 (1, 2, 3, 4, 5, 6) [1]: 1
Select promotion_criterion:
1 - sagemaker_framework.utils.promotion.maximize
2 - sagemaker_framework.utils.promotion.minimize
3 - sagemaker_framework.utils.promotion.maximize_with_tol
4 - sagemaker_framework.utils.promotion.minimize_with_tol
5 - sagemaker_framework.utils.promotion.manual
6 - sagemaker_framework.utils.promotion.always_promote
Choose from 1, 2, 3, 4, 5, 6 (1, 2, 3, 4, 5, 6) [1]: 6
preprocessing_cpus [1]:
preprocessing_memory_in_gb [4]: 8
test_proportion [0.2]: 0.1
training_cpus [1]:
training_memory_in_gb [4]:
training_volume_size_in_gb [2]:
max_training_runtime_in_minutes [30]: 60
min_serving_instances [1]:
max_serving_instances [10]: 1
```

```
$ tree -a europython-example-model/
europython-example-model/
├── .bellybutton.yml
├── bin
│   ├── build-docker-image
│   └── deploy.sh
├── .circleci
│   └── config.yml
├── docker-compose.yml
├── Dockerfile
├── europython_example_model
│   ├── config.py
│   ├── __init__.py
│   └── model.py
├── .github
│   ├── CODEOWNERS
│   └── PULL_REQUEST_TEMPLATE.md
├── .gitignore
├── README.md
├── requirements.txt
├── sagemaker-config.yml
├── setup.py
└── tests
    ├── test_config.py
    ├── test_model.py
    └── test-model.txt

5 directories, 19 files
```

Developer experience

```
$ cookiecutter git@github.com:teikametrics/sagemaker-framework.git
github_username [my-github-username]: hchasestevens
project_name [my-sagemaker-model]: europython-example-model
project_slug [europython_example_model]:
model_name [europython-example-model]:
description [An ML model living on the SageMaker platform.]: An example model for
Europython 2020.
Select model_validation_metric:
1 - sklearn.metrics.mean_squared_error
2 - sklearn.metrics.r2_score
3 - sklearn.metrics.accuracy_score
4 - sklearn.metrics.log_loss
5 - sklearn.metrics.f1_score
6 - sagemaker_framework.utils.metrics.mean_absolute_percentage_error
Choose from 1, 2, 3, 4, 5, 6 (1, 2, 3, 4, 5, 6) [1]: 1
Select promotion_criterion:
1 - sagemaker_framework.utils.promotion.maximize
2 - sagemaker_framework.utils.promotion.minimize
3 - sagemaker_framework.utils.promotion.maximize_with_tol
4 - sagemaker_framework.utils.promotion.minimize_with_tol
5 - sagemaker_framework.utils.promotion.manual
6 - sagemaker_framework.utils.promotion.always_promote
Choose from 1, 2, 3, 4, 5, 6 (1, 2, 3, 4, 5, 6) [1]: 6
preprocessing_cpus [1]:
preprocessing_memory_in_gb [4]: 8
test_proportion [0.2]: 0.1
training_cpus [1]:
training_memory_in_gb [4]:
training_volume_size_in_gb [2]:
max_training_runtime_in_minutes [30]: 60
min_serving_instances [1]:
max_serving_instances [10]: 1
```

```
$ tree -a europython-example-model/
europython-example-model/
├── .bellybutton.yml
├── bin
│   ├── build-docker-image
│   └── deploy.sh
├── .circleci
│   └── config.yml
├── docker-compose.yml
├── Dockerfile
├── europython_example_model
│   ├── config.py
│   ├── __init__.py
│   └── model.py
├── .github
│   ├── CODEOWNERS
│   └── PULL_REQUEST_TEMPLATE.md
├── .gitignore
├── README.md
├── requirements.txt
├── sagemaker-config.yml
├── setup.py
└── tests
    ├── test_config.py
    ├── test_model.py
    └── test-model.txt

5 directories, 19 files
```

Developer experience

1

```
def preprocess_data(seed=None) -> PreprocessingResult:
    """Preprocess data for training."""
    fetch_adgroup_performances_query = """
        SELECT
            ad_group_id,
            SUM(lkr.conversions_7d_attr) AS conversions,
            SUM(lkr.sales_7d_attr) AS sales
        FROM main.transforms.latest_keyword_reports lkr
            WHERE lkr.conversions_7d_attr > 0
                AND lkr.sales_7d_attr > 0
                AND lkr.keyword_report_local_date >= current_date() - 30
            GROUP BY ad_group_id
        """
    return PreprocessingResult(
        training={
            'performances.msgpack': adgroup_performances[
                ~adgroup_performances.test
            ].apply(pd.to_numeric).to_msgpack(),
        }.items(),
        validation=(),
        testing=test_cases
    )
```

Developer experience

1

```python
def preprocess_data(seed=None) -> PreprocessingResult:
    """Preprocess data for training."""
    fetch_adgroup_performances_query = """
        SELECT
            ad_group_id,
            SUM(lkr.conversions_7d_attr) AS conversions,
            SUM(lkr.sales_7d_attr) AS sales
        FROM main.transforms.latest_keyword_reports lkr
            WHERE lkr.conversions_7d_attr > 0
                AND lkr.sales_7d_attr > 0
                AND lkr.keyword_report_local_date >= current_date() - 30
            GROUP BY ad_group_id
        """
    return PreprocessingResult(
        training={
            'performances.msgpack': adgroup_performances[
                ~adgroup_performances.test
            ].apply(pd.to_numeric).to_msgpack(),
        }.items(),
        validation=(),
        testing=test_cases
    )
```

2

```python
def train_model(training_path: Path, validation_path: Path) -> Artifacts:
    training_dfs = load_zipped_data(
        training_path,
        fnames=MSGPACK_FNAMES,
        deserializer=pd.read_msgpack
    )
    all_adgroup_prices = training_dfs['prices.msgpack']
    performances = training_dfs['performances.msgpack']
    results = {
        marketplace_id: train_marketplace_model(
            marketplace_id=marketplace_id,
            market_adgroup_prices=market_df,
            performances=performances,
        )._asdict()
        for marketplace_id, market_df in all_adgroup_prices.groupby('marketplace_id')
    }
    return Artifacts({MODEL_FNAME: json.dumps(results).encode('utf-8')})
```

Developer experience

1

```
def preprocess_data(seed=None) -> PreprocessingResult:
    """Preprocess data for training."""
    fetch_adgroup_performances_query = """
        SELECT
            ad_group_id,
            SUM(lkr.conversions_7d_attr) AS conversions,
            SUM(lkr.sales_7d_attr) AS sales
        FROM main.transforms.latest_keyword_reports lkr
            WHERE lkr.conversions_7d_attr > 0
                AND lkr.sales_7d_attr > 0
                AND lkr.keyword_report_local_date >= current_date() - 30
            GROUP BY ad_group_id
        """
    return PreprocessingResult(
        training={
            'performances.msgpack': adgroup_performances[
                ~adgroup_performances.test
            ].apply(pd.to_numeric).to_msgpack(),
        }.items(),
        validation=(),
        testing=test_cases
    )
```

2

```
def train_model(training_path: Path, validation_path: Path) -> Artifacts:
    training_dfs = load_zipped_data(
        training_path,
        fnames=MSGPACK_FNAMES,
        deserializer=pd.read_msgpack
    )
    all_adgroup_prices = training_dfs['prices.msgpack']
    performances = training_dfs['performances.msgpack']
    results = {
        marketplace_id: train_marketplace_model(
            marketplace_id=marketplace_id,
            market_adgroup_prices=market_df,
            performances=performances,
        )._asdict()
        for marketplace_id, market_df in all_adgroup_prices.groupby('marketplace_id')
    }
    return Artifacts({MODEL_FNAME: json.dumps(results).encode('utf-8')})
```

3

```
def load_model(path: Path) -> Model:
    with (path / MODEL_FNAME).open('r', encoding='utf-8') as f:
        parameters = {k: Parameters(**v) for k, v in json.load(f).items()}
    def model(configuration, instances) -> List[Optional[float]]:
        return [
            estimate_sales_per_conversion(...)
            for price, conversions, sales in instances
        ]
    return model
```

Developer experience

```
request_schema: !jsonschema {
  type: 'object',
  properties: {
    configuration: {
      type: 'object',
      properties: {
        marketplaceId: {type: 'string'}
      },
    },
    instances: {
      type: 'array',
      items: {
        type: 'array',
        items: [
          {type: 'number', description: "Price", exclusiveMinimum: 0},
          {type: 'number', description: "Conversions", exclusiveMinimum: 0},
          {type: 'number', description: "Sales", exclusiveMinimum: 0}
        ],
      },
    },
    requesterId: {type: 'string'}
  },
  required: ['instances', 'configuration', 'requesterId'],
}
```

```
response_schema: !jsonschema {
  type: 'array',
  items: {type: 'number'},
  description: "Estimated sales per conversion, in order corresponding to request order"
}
```

Developer experience

- Test suite
- Linting (pylint, mypy, bellybutton)
- Dockerization
- CI/CD
- Airflow DAG generation
- Training orchestration
- Automated model evaluation and promotion
- Gradual rollout

Developer experience

- Test suite
- Linting (pylint, mypy, bellybutton)
- Dockerization
- CI/CD
- Airflow DAG generation
- Training orchestration
- Automated model evaluation and promotion
- Gradual rollout
- Automated rollback
- Monitoring
- Alerting
- Diagnostics
- Autoscaling
- Schema validation
- Data capture
- Healthchecks
- Cost monitoring

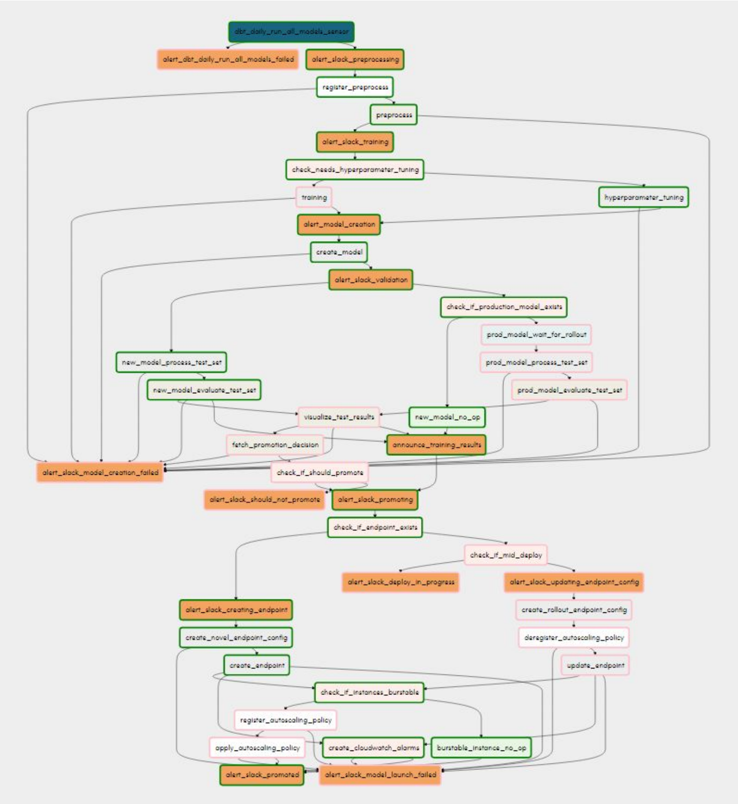
Developer experience

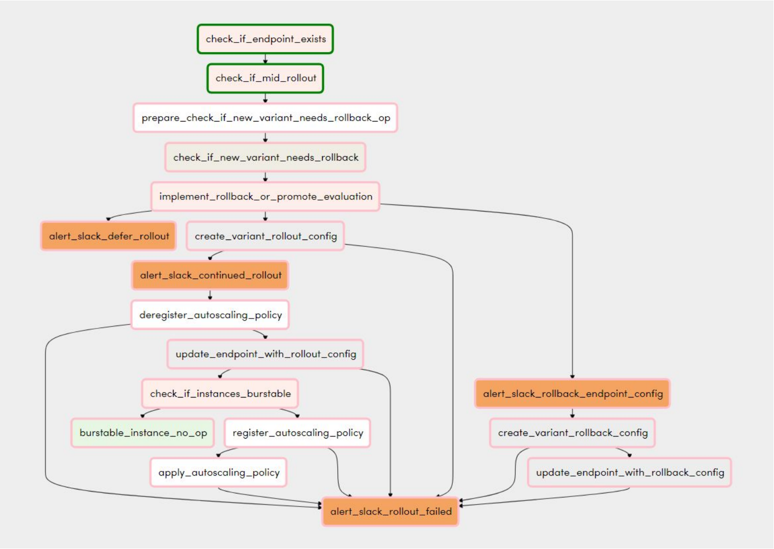
- Test suite
- Linting (pylint, mypy, bellybutton)
- Dockerization
- CI/CD
- Airflow DAG generation
- Training orchestration
- Automated model evaluation and promotion
- Gradual rollout

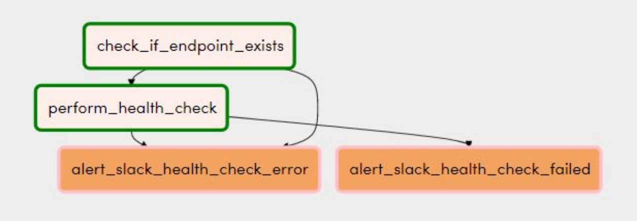
- Automated rollback
- Monitoring
- Alerting
- Diagnostics
- Autoscaling
- Schema validation
- Data capture
- Healthchecks
- Cost monitoring

III. Data scientists want to do data science

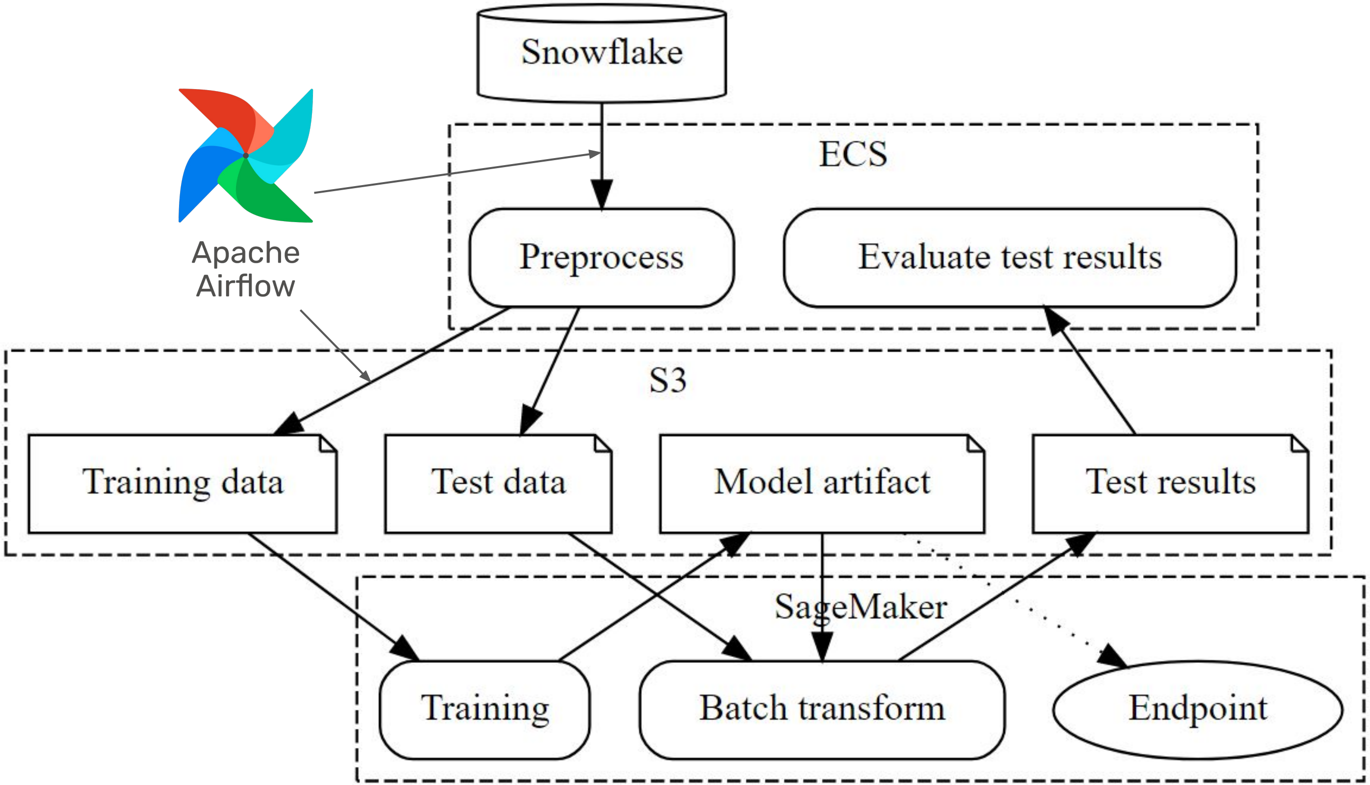
Developer experience


dbt_daily_run_all_models_sensor
alert_dbt_daily_run_all_models_failed
alert_slack_preprocessing
register_preprocess
preprocess
alert_slack_training
check_needs_hyperparameter_tuning
training
hyperparameter_tuning
alert_model_creation
create_model
alert_slack_validation
check_if_production_model_exists
prod_model_wait_for_rollout
new_model_process_test_set
prod_model_process_test_set
new_model_evaluate_test_set
prod_model_evaluate_test_set
visualize_test_results
new_model_no_op
fetch_promotion_decision
announce_training_results
alert_slack_model_creation_failed
check_if_should_promote
alert_slack_should_not_promote
alert_slack_promoting
check_if_endpoint_exists
check_if_mid_deploy
alert_slack_deploy_in_progress
alert_slack_updating_endpoint_config
alert_slack_creating_endpoint
create_rollout_endpoint_config
create_novel_endpoint_config
deregister_autoscaling_policy
create_endpoint
update_endpoint
check_if_instances_burstable
register_autoscaling_policy
apply_autoscaling_policy
create_cloudwatch_alarms
burstable_instance_no_op
alert_slack_promoted
alert_slack_model_launch_failed


check_if_endpoint_exists
check_if_mid_rollout
prepare_check_if_new_variant_needs_rollback_op
check_if_new_variant_needs_rollback
implement_rollback_or_promote_evaluation
alert_slack_defer_rollout
create_variant_rollout_config
alert_slack_continued_rollout
deregister_autoscaling_policy
update_endpoint_with_rollout_config
check_if_instances_burstable
alert_slack_rollback_endpoint_config
burstable_instance_no_op
register_autoscaling_policy
create_variant_rollback_config
apply_autoscaling_policy
update_endpoint_with_rollback_config
alert_slack_rollout_failed


check_if_endpoint_exists
perform_health_check
alert_slack_health_check_error
alert_slack_health_check_failed

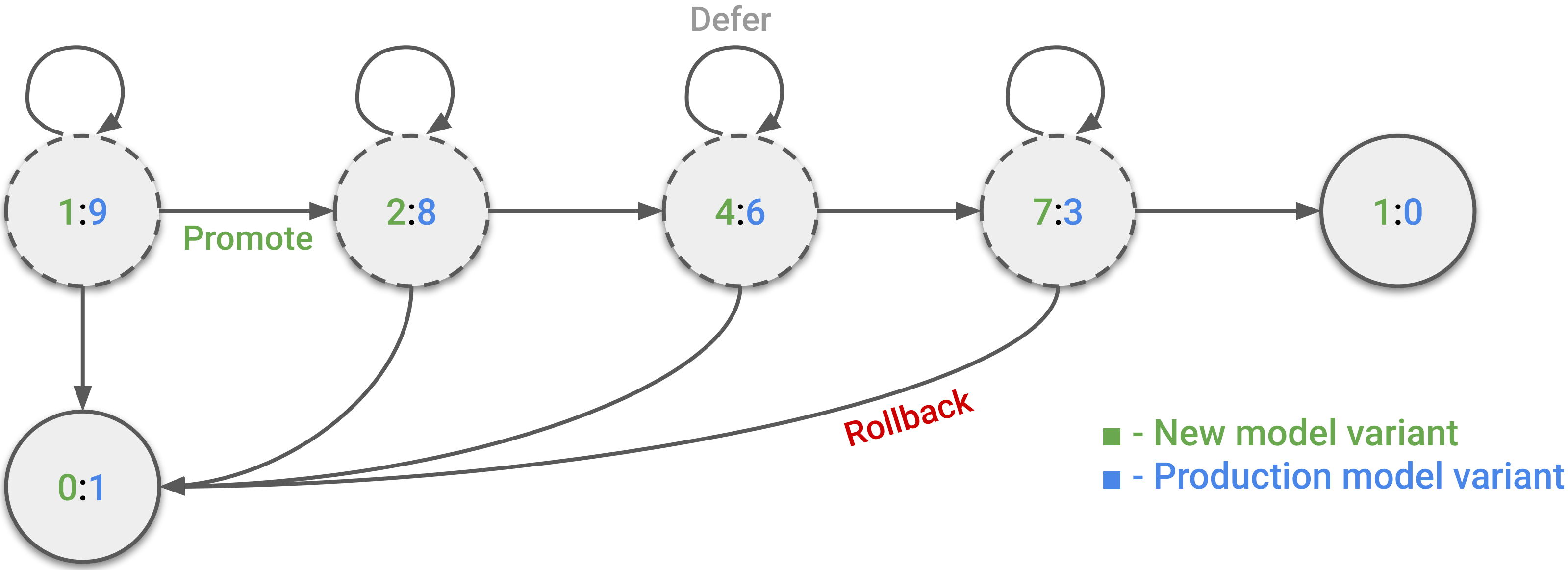
Our stack


Snowflake
Apache
Airflow
ECS
Preprocess
Evaluate test results
S3
Training data
Test data
Model artifact
Test results
SageMaker
Training
Batch transform
Endpoint

Our stack

Technology	Application
AWS SageMaker	Model training, hosting; provenance info
Airflow (Astronomer.io)	Model lifecycle orchestration
Docker	Model packaging
Cookiecutter	Model repo templating
Jsonschema	Schema definition; PBT
Flask, gunicorn	Model server
DBT	Scalable data processing (in-warehouse)
Slack	Notifications, diagnostics
Pylint, mypy, bellybutton	Linting
Pytest, hypothesis, hypothesis-jsonschema	Test suite

Our stack



Defer

1:9

Promote

2:8

4:6

7:3

1:0

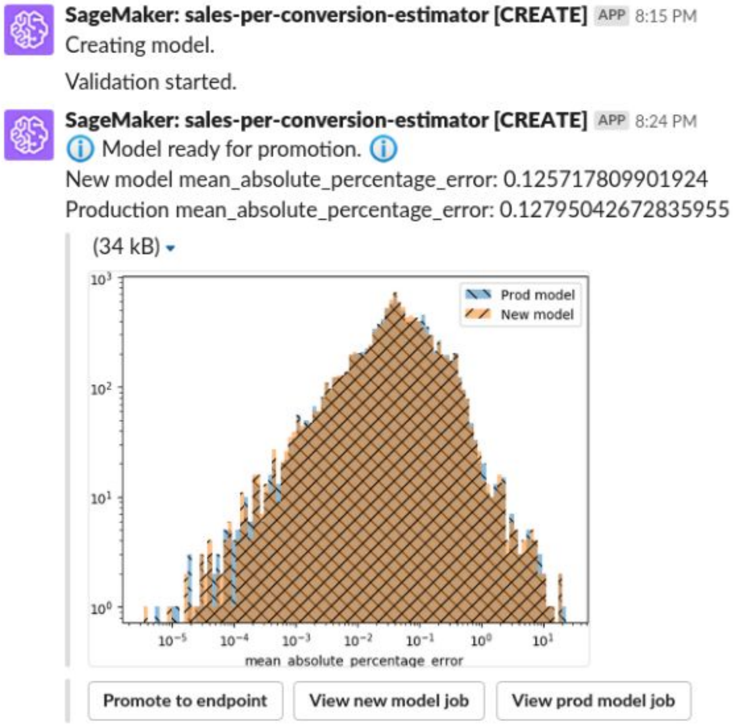
0:1

Rollback

🟩 - New model variant

🟦 - Production model variant

Lessons learned



SageMaker: sales-per-conversion-estimator [CREATE] APP 8:15 PM

Creating model.

Validation started.

SageMaker: sales-per-conversion-estimator [CREATE] APP 8:24 PM

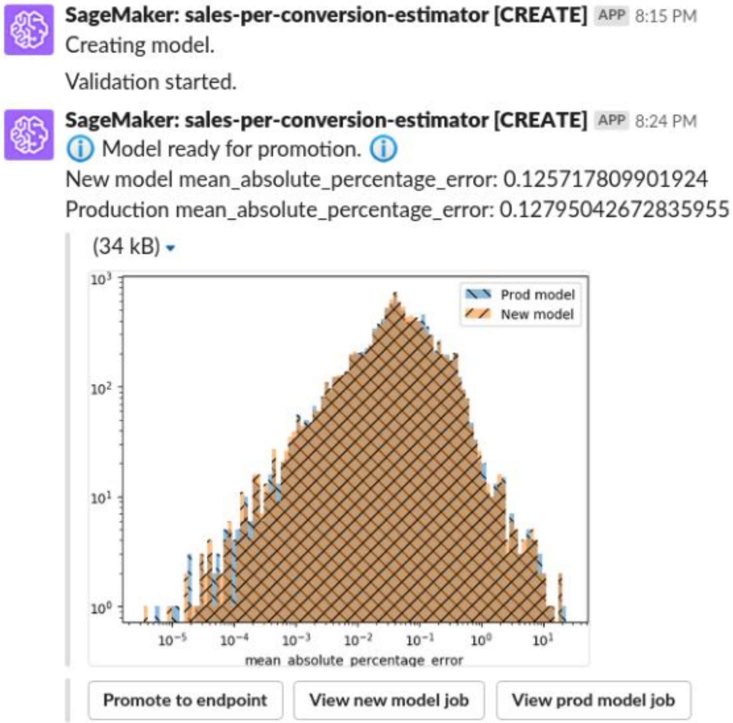
ⓘ Model ready for promotion. ⓘ

New model mean_absolute_percentage_error: 0.125717809901924

Production mean_absolute_percentage_error: 0.12795042672835955

(34 kB) ▾

Lessons learned



SageMaker: sales-per-conversion-estimator [CREATE] APP 8:15 PM

Creating model.

Validation started.

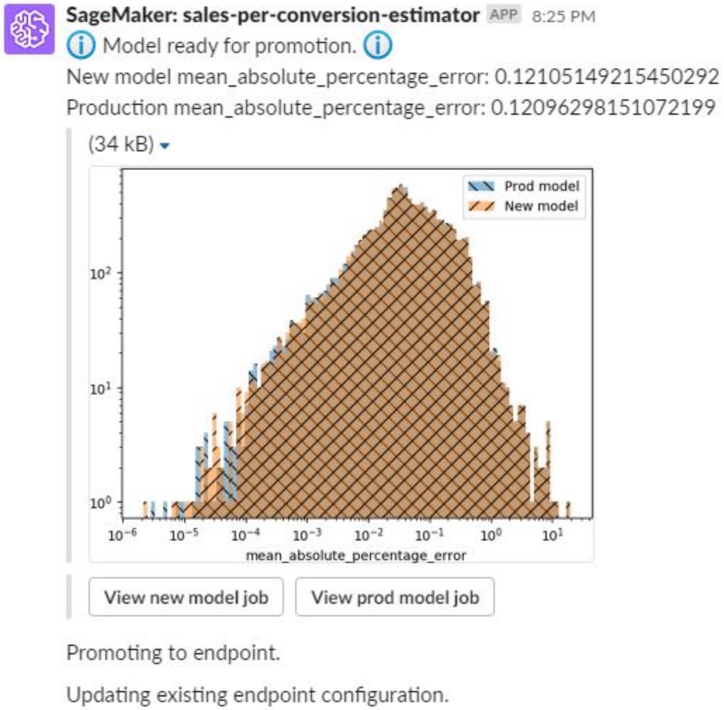
SageMaker: sales-per-conversion-estimator [CREATE] APP 8:24 PM

ⓘ Model ready for promotion. ⓘ

New model mean_absolute_percentage_error: 0.125717809901924

Production mean_absolute_percentage_error: 0.12795042672835955

(34 kB)



SageMaker: sales-per-conversion-estimator APP 8:25 PM

ⓘ Model ready for promotion. ⓘ

New model mean_absolute_percentage_error: 0.12105149215450292

Production mean_absolute_percentage_error: 0.12096298151072199

(34 kB)

Promoting to endpoint.

Updating existing endpoint configuration.

Lessons learned

"Best Practices" (whatever that means):

```
                7
             +-----+
             |     |  5
 Arbitrary   |     +-----+
Productivity |     |     |
   Units     |     |     |
             |     |     |
             |     |     |
             +-----+-----+
                X     Y
```

Gonzales, G. (2016). Worst practices should be hard. http://www.haskellforall.com/2016/04/worst-practices-should-be-hard.html

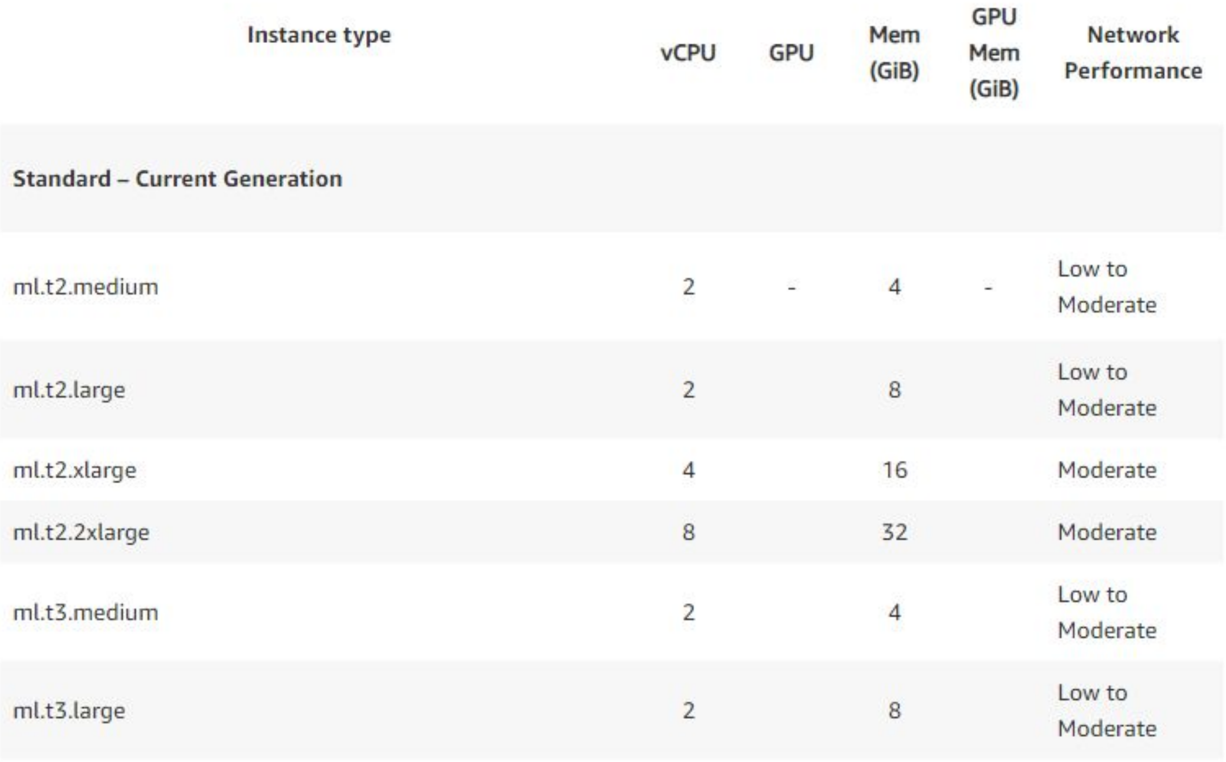
Lessons learned

```
"Best Practices" (whatever that means):

                7
             +-----+
             |     |  5
 Arbitrary   |     +-----+
Productivity |     |     |
   Units     |     |     |
             |     |     |
             |     |     |
             +-----+-----+
                X     Y
```

```
"Worst Practices" (whatever that means):

                9
             +-----+
             |     |
             |     |
 Arbitrary   |     |
Productivity |     |
   Units     |     |  3
             |     +-----+
             |     |     |
             |     |     |
             +-----+-----+
                X     Y
```

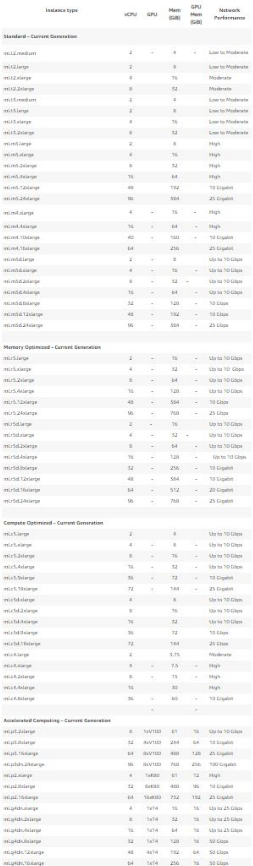
Gonzales, G. (2016). Worst practices should be hard. http://www.haskellforall.com/2016/04/worst-practices-should-be-hard.html

Lessons learned



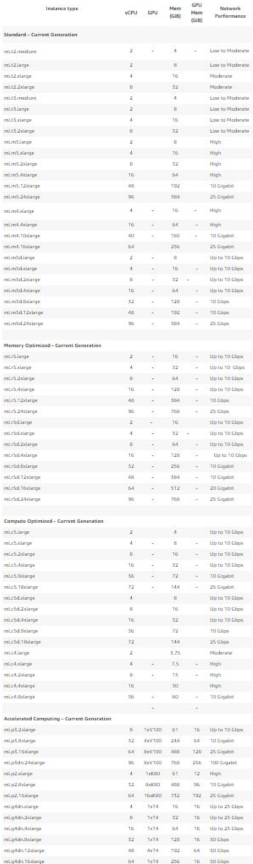
Instance type	vCPU	GPU	Mem (GiB)	GPU Mem (GiB)	Network Performance
Standard – Current Generation					
ml.t2.medium	2	-	4	-	Low to Moderate
ml.t2.large	2		8		Low to Moderate
ml.t2.xlarge	4		16		Moderate
ml.t2.2xlarge	8		32		Moderate
ml.t3.medium	2		4		Low to Moderate
ml.t3.large	2		8		Low to Moderate

Lessons learned

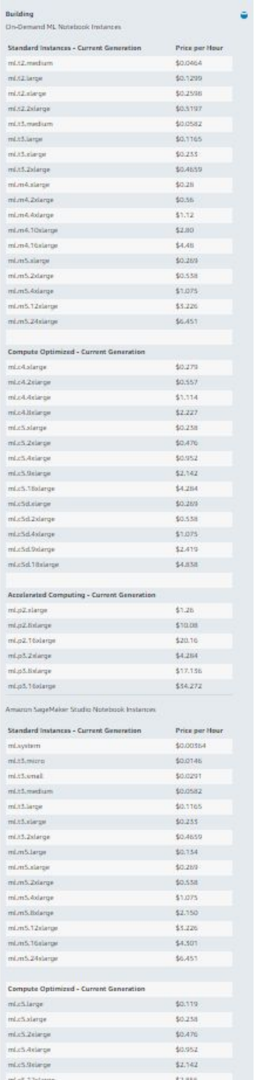


Instance type	vCPU	GPU	Mem (GiB)	GPU Mem (GiB)	Network Performance
Standard – Current Generation					
ml.t2.medium	2	-	4	-	Low to Moderate
ml.t2.large	2		8		Low to Moderate
ml.t2.xlarge	4		16		Moderate
ml.t2.2xlarge	8		32		Moderate
ml.t3.medium	2		4		Low to Moderate
ml.t3.large	2		8		Low to Moderate
ml.t3.xlarge	4		16		Low to Moderate
ml.t3.2xlarge	8		32		Low to Moderate
ml.m5.large	2		8		High
ml.m5.xlarge	4		16		High
ml.m5.2xlarge	8		32		High
ml.m5.4xlarge	16		64		High
ml.m5.12xlarge	48		192		10 Gigabit
ml.m5.24xlarge	96		384		25 Gigabit
ml.m4.xlarge	4	-	16	-	High
ml.m4.4xlarge	16	-	64	-	High
ml.m4.10xlarge	40	-	160	-	10 Gigabit
ml.m4.16xlarge	64		256		25 Gigabit
ml.m5d.large	2	-	8		Up to 10 Gbps
ml.m5d.xlarge	4	-	16	-	Up to 10 Gbps
ml.m5d.2xlarge	8	-	32	-	Up to 10 Gbps
ml.m5d.4xlarge	16	-	64	-	Up to 10 Gbps
ml.m5d.8xlarge	32	-	128	-	10 Gbps
ml.m5d.12xlarge	48	-	192	-	10 Gbps
ml.m5d.24xlarge	96	-	384	-	25 Gbps
Memory Optimized – Current Generation					
ml.r5.large	2	-	16	-	Up to 10 Gbps
ml.r5.xlarge	4	-	32	-	Up to 10 Gbps
ml.r5.2xlarge	8	-	64	-	Up to 10 Gbps
ml.r5.4xlarge	16	-	128	-	Up to 10 Gbps
ml.r5.12xlarge	48	-	384	-	10 Gbps
ml.r5.24xlarge	96	-	768	-	25 Gbps
ml.r5d.large	2	-	16		Up to 10 Gbps
ml.r5d.xlarge	4	-	32	-	Up to 10 Gbps
ml.r5d.2xlarge	8	-	64	-	Up to 10 Gbps
ml.r5d.4xlarge	16	-	128	-	Up to 10 Gbps
ml.r5d.8xlarge	32	-	256	-	10 Gigabit
ml.r5d.12xlarge	48	-	384	-	10 Gigabit
ml.r5d.16xlarge	64	-	512	-	20 Gigabit
ml.r5d.24xlarge	96	-	768	-	25 Gigabit
Compute Optimized – Current Generation					
ml.c5.large	2		4		Up to 10 Gbps
ml.c5.xlarge	4	-	8	-	Up to 10 Gbps
ml.c5.2xlarge	8	-	16	-	Up to 10 Gbps
ml.c5.4xlarge	16	-	32	-	Up to 10 Gbps
ml.c5.9xlarge	36	-	72	-	10 Gigabit
ml.c5.18xlarge	72	-	144	-	25 Gigabit
ml.c5d.xlarge	4		8		Up to 10 Gbps
ml.c5d.2xlarge	8		16		Up to 10 Gbps
ml.c5d.4xlarge	16		32		Up to 10 Gbps
ml.c5d.9xlarge	36		72		10 Gbps
ml.c5d.18xlarge	72		144		25 Gbps
ml.c4.large	2		3.75		Moderate
ml.c4.xlarge	4	-	7.5	-	High
ml.c4.2xlarge	8	-	15	-	High
ml.c4.4xlarge	16		30		High
ml.c4.8xlarge	36	-	60	-	10 Gigabit
		-		-	
Accelerated Computing – Current Generation					
ml.p3.2xlarge	8	1xV100	61	16	Up to 10 Gbps
ml.p3.8xlarge	32	4xV100	244	64	10 Gigabit
ml.p3.16xlarge	64	8xV100	488	128	25 Gigabit
ml.p3dn.24xlarge	96	8xV100	768	256	100 Gigabit
ml.p2.xlarge	4	1xK80	61	12	High
ml.p2.8xlarge	32	8xK80	488	96	10 Gigabit
ml.p2.16xlarge	64	16xK80	732	192	25 Gigabit
ml.g4dn.xlarge	4	1xT4	16	16	Up to 25 Gbps
ml.g4dn.2xlarge	8	1xT4	32	16	Up to 25 Gbps
ml.g4dn.4xlarge	16	1xT4	64	16	Up to 25 Gbps
ml.g4dn.8xlarge	32	1xT4	128	16	50 Gbps
ml.g4dn.12xlarge	48	4xT4	192	64	50 Gbps
ml.g4dn.16xlarge	64	1xT4	256	16	50 Gbps

Lessons learned



Instance type	vCPU	GPU	Mem (GiB)	GPU Mem (GiB)	Network Performance
Standard – Current Generation					
ml.t2.medium	2	-	4	-	Low to Moderate
ml.t2.large	2		8		Low to Moderate
ml.t2.xlarge	4		16		Moderate
ml.t2.2xlarge	8		32		Moderate
ml.t3.medium	2		4		Low to Moderate
ml.t3.large	2		8		Low to Moderate
ml.t3.xlarge	4		16		Low to Moderate
ml.t3.2xlarge	8		32		Low to Moderate
ml.m5.large	2		8		High
ml.m5.xlarge	4		16		High
ml.m5.2xlarge	8		32		High
ml.m5.4xlarge	16		64		High
ml.m5.12xlarge	48		192		10 Gigabit
ml.m5.24xlarge	96		384		25 Gigabit
ml.m4.xlarge	4	-	16	-	High
ml.m4.4xlarge	16	-	64	-	High
ml.m4.10xlarge	40	-	160	-	10 Gigabit
ml.m4.16xlarge	64		256		25 Gigabit
ml.m5d.large	2	-	8		Up to 10 Gbps
ml.m5d.xlarge	4	-	16	-	Up to 10 Gbps
ml.m5d.2xlarge	8	-	32	-	Up to 10 Gbps
ml.m5d.4xlarge	16	-	64	-	Up to 10 Gbps
ml.m5d.8xlarge	32	-	128	-	10 Gbps
ml.m5d.12xlarge	48	-	192	-	10 Gbps
ml.m5d.24xlarge	96	-	384	-	25 Gbps
Memory Optimized – Current Generation					
ml.r5.large	2	-	16	-	Up to 10 Gbps
ml.r5.xlarge	4	-	32	-	Up to 10 Gbps
ml.r5.2xlarge	8	-	64	-	Up to 10 Gbps
ml.r5.4xlarge	16	-	128	-	Up to 10 Gbps
ml.r5.12xlarge	48	-	384	-	10 Gbps
ml.r5.24xlarge	96	-	768	-	25 Gbps
ml.r5d.large	2	-	16		Up to 10 Gbps
ml.r5d.xlarge	4	-	32	-	Up to 10 Gbps
ml.r5d.2xlarge	8	-	64	-	Up to 10 Gbps
ml.r5d.4xlarge	16	-	128	-	Up to 10 Gbps
ml.r5d.8xlarge	32	-	256	-	10 Gigabit
ml.r5d.12xlarge	48	-	384	-	10 Gigabit
ml.r5d.16xlarge	64	-	512	-	20 Gigabit
ml.r5d.24xlarge	96	-	768	-	25 Gigabit
Compute Optimized – Current Generation					
ml.c5.large	2		4		Up to 10 Gbps
ml.c5.xlarge	4	-	8	-	Up to 10 Gbps
ml.c5.2xlarge	8	-	16	-	Up to 10 Gbps
ml.c5.4xlarge	16	-	32	-	Up to 10 Gbps
ml.c5.9xlarge	36	-	72	-	10 Gigabit
ml.c5.18xlarge	72	-	144	-	25 Gigabit
ml.c5d.xlarge	4		8		Up to 10 Gbps
ml.c5d.2xlarge	8		16		Up to 10 Gbps
ml.c5d.4xlarge	16		32		Up to 10 Gbps
ml.c5d.9xlarge	36		72		10 Gbps
ml.c5d.18xlarge	72		144		25 Gbps
ml.c4.large	2		3.75		Moderate
ml.c4.xlarge	4	-	7.5	-	High
ml.c4.2xlarge	8	-	15	-	High
ml.c4.4xlarge	16		30		High
ml.c4.8xlarge	36	-	60	-	10 Gigabit
		-		-	
Accelerated Computing – Current Generation					
ml.p3.2xlarge	8	1xV100	61	16	Up to 10 Gbps
ml.p3.8xlarge	32	4xV100	244	64	10 Gigabit
ml.p3.16xlarge	64	8xV100	488	128	25 Gigabit
ml.p3dn.24xlarge	96	8xV100	768	256	100 Gigabit
ml.p2.xlarge	4	1xK80	61	12	High
ml.p2.8xlarge	32	8xK80	488	96	10 Gigabit
ml.p2.16xlarge	64	16xK80	732	192	25 Gigabit
ml.g4dn.xlarge	4	1xT4	16	16	Up to 25 Gbps
ml.g4dn.2xlarge	8	1xT4	32	16	Up to 25 Gbps
ml.g4dn.4xlarge	16	1xT4	64	16	Up to 25 Gbps
ml.g4dn.8xlarge	32	1xT4	128	16	50 Gbps
ml.g4dn.12xlarge	48	4xT4	192	64	50 Gbps
ml.g4dn.16xlarge	64	1xT4	256	16	50 Gbps



Building

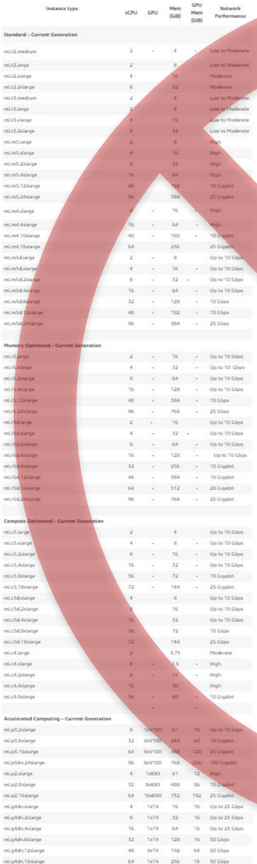
On-Demand ML Notebook Instances

Standard Instances - Current Generation	Price per Hour
ml.t2.medium	$0.0464
ml.t2.large	$0.1299
ml.t2.xlarge	$0.2598
ml.t2.2xlarge	$0.5197
ml.t3.medium	$0.0582
ml.t3.large	$0.1165
ml.t3.xlarge	$0.233
ml.t3.2xlarge	$0.4659
ml.m4.xlarge	$0.28
ml.m4.2xlarge	$0.56
ml.m4.4xlarge	$1.12
ml.m4.10xlarge	$2.80
ml.m4.16xlarge	$4.48
ml.m5.xlarge	$0.269
ml.m5.2xlarge	$0.538
ml.m5.4xlarge	$1.075
ml.m5.12xlarge	$3.226
ml.m5.24xlarge	$6.451
Compute Optimized - Current Generation	
ml.c4.xlarge	$0.279
ml.c4.2xlarge	$0.557
ml.c4.4xlarge	$1.114
ml.c4.8xlarge	$2.227
ml.c5.xlarge	$0.238
ml.c5.2xlarge	$0.476
ml.c5.4xlarge	$0.952
ml.c5.9xlarge	$2.142
ml.c5.18xlarge	$4.284
ml.c5d.xlarge	$0.269
ml.c5d.2xlarge	$0.538
ml.c5d.4xlarge	$1.075
ml.c5d.9xlarge	$2.419
ml.c5d.18xlarge	$4.838
Accelerated Computing - Current Generation	
ml.p2.xlarge	$1.26
ml.p2.8xlarge	$10.08
ml.p2.16xlarge	$20.16
ml.p3.2xlarge	$4.284
ml.p3.8xlarge	$17.136
ml.p3.16xlarge	$34.272

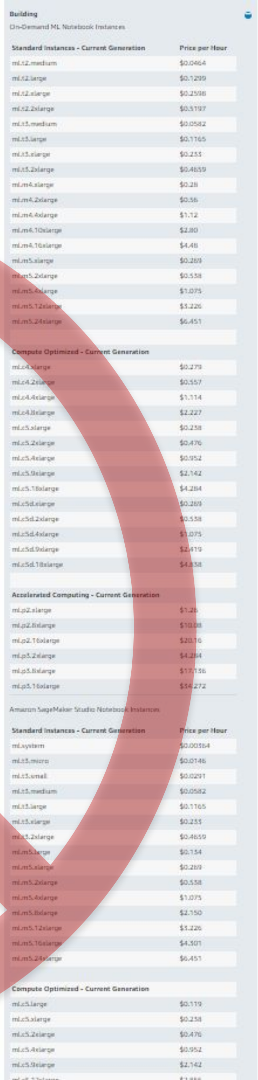
Amazon SageMaker Studio Notebook Instances

Standard Instances - Current Generation	Price per Hour
ml.system	$0.00364
ml.t3.micro	$0.0146
ml.t3.small	$0.0291
ml.t3.medium	$0.0582
ml.t3.large	$0.1165
ml.t3.xlarge	$0.233
ml.t3.2xlarge	$0.4659
ml.m5.large	$0.134
ml.m5.xlarge	$0.269
ml.m5.2xlarge	$0.538
ml.m5.4xlarge	$1.075
ml.m5.8xlarge	$2.150
ml.m5.12xlarge	$3.226
ml.m5.16xlarge	$4.301
ml.m5.24xlarge	$6.451
Compute Optimized - Current Generation	
ml.c5.large	$0.119
ml.c5.xlarge	$0.238
ml.c5.2xlarge	$0.476
ml.c5.4xlarge	$0.952
ml.c5.9xlarge	$2.142

Lessons learned



Instance type	vCPU	GPU	Mem (GiB)	GPU Mem (GiB)	Network Performance
Standard – Current Generation					
ml.t2.medium	2	-	4	-	Low to Moderate
ml.t2.large	2		8		Low to Moderate
ml.t2.xlarge	4		16		Moderate
ml.t2.2xlarge	8		32		Moderate
ml.t3.medium	2		4		Low to Moderate
ml.t3.large	2		8		Low to Moderate
ml.t3.xlarge	4		16		Low to Moderate
ml.t3.2xlarge	8		32		Low to Moderate
ml.m5.large	2		8		High
ml.m5.xlarge	4		16		High
ml.m5.2xlarge	8		32		High
ml.m5.4xlarge	16		64		High
ml.m5.12xlarge	48		192		10 Gigabit
ml.m5.24xlarge	96		384		25 Gigabit
ml.m4.xlarge	4	-	16	-	High
ml.m4.4xlarge	16	-	64	-	High
ml.m4.10xlarge	40	-	160	-	10 Gigabit
ml.m4.16xlarge	64		256		25 Gigabit
ml.m5d.large	2	-	8		Up to 10 Gbps
ml.m5d.xlarge	4	-	16	-	Up to 10 Gbps
ml.m5d.2xlarge	8	-	32	-	Up to 10 Gbps
ml.m5d.4xlarge	16	-	64	-	Up to 10 Gbps
ml.m5d.8xlarge	32	-	128	-	10 Gbps
ml.m5d.12xlarge	48	-	192	-	10 Gbps
ml.m5d.24xlarge	96	-	384	-	25 Gbps
Memory Optimized – Current Generation					
ml.r5.large	2	-	16	-	Up to 10 Gbps
ml.r5.xlarge	4	-	32	-	Up to 10 Gbps
ml.r5.2xlarge	8	-	64	-	Up to 10 Gbps
ml.r5.4xlarge	16	-	128	-	Up to 10 Gbps
ml.r5.12xlarge	48	-	384	-	10 Gbps
ml.r5.24xlarge	96	-	768	-	25 Gbps
ml.r5d.large	2	-	16		Up to 10 Gbps
ml.r5d.xlarge	4	-	32	-	Up to 10 Gbps
ml.r5d.2xlarge	8	-	64	-	Up to 10 Gbps
ml.r5d.4xlarge	16	-	128	-	Up to 10 Gbps
ml.r5d.8xlarge	32	-	256	-	10 Gigabit
ml.r5d.12xlarge	48	-	384	-	10 Gigabit
ml.r5d.16xlarge	64	-	512	-	20 Gigabit
ml.r5d.24xlarge	96	-	768	-	25 Gigabit
Compute Optimized – Current Generation					
ml.c5.large	2		4		Up to 10 Gbps
ml.c5.xlarge	4	-	8	-	Up to 10 Gbps
ml.c5.2xlarge	8	-	16	-	Up to 10 Gbps
ml.c5.4xlarge	16	-	32	-	Up to 10 Gbps
ml.c5.9xlarge	36	-	72	-	10 Gigabit
ml.c5.18xlarge	72	-	144	-	25 Gigabit
ml.c5d.xlarge	4		8		Up to 10 Gbps
ml.c5d.2xlarge	8		16		Up to 10 Gbps
ml.c5d.4xlarge	16		32		Up to 10 Gbps
ml.c5d.9xlarge	36		72		10 Gbps
ml.c5d.18xlarge	72		144		25 Gbps
ml.c4.large	2		3.75		Moderate
ml.c4.xlarge	4	-	7.5	-	High
ml.c4.2xlarge	8	-	15	-	High
ml.c4.4xlarge	16		30		High
ml.c4.8xlarge	36	-	60	-	10 Gigabit
		-		-	
Accelerated Computing – Current Generation					
ml.p3.2xlarge	8	1xV100	61	16	Up to 10 Gbps
ml.p3.8xlarge	32	4xV100	244	64	10 Gigabit
ml.p3.16xlarge	64	8xV100	488	128	25 Gigabit
ml.p3dn.24xlarge	96	8xV100	768	256	100 Gigabit
ml.p2.xlarge	4	1xK80	61	12	High
ml.p2.8xlarge	32	8xK80	488	96	10 Gigabit
ml.p2.16xlarge	64	16xK80	732	192	25 Gigabit
ml.g4dn.xlarge	4	1xT4	16	16	Up to 25 Gbps
ml.g4dn.2xlarge	8	1xT4	32	16	Up to 25 Gbps
ml.g4dn.4xlarge	16	1xT4	64	16	Up to 25 Gbps
ml.g4dn.8xlarge	32	1xT4	128	16	50 Gbps
ml.g4dn.12xlarge	48	4xT4	192	64	50 Gbps
ml.g4dn.16xlarge	64	1xT4	256	16	50 Gbps



Building

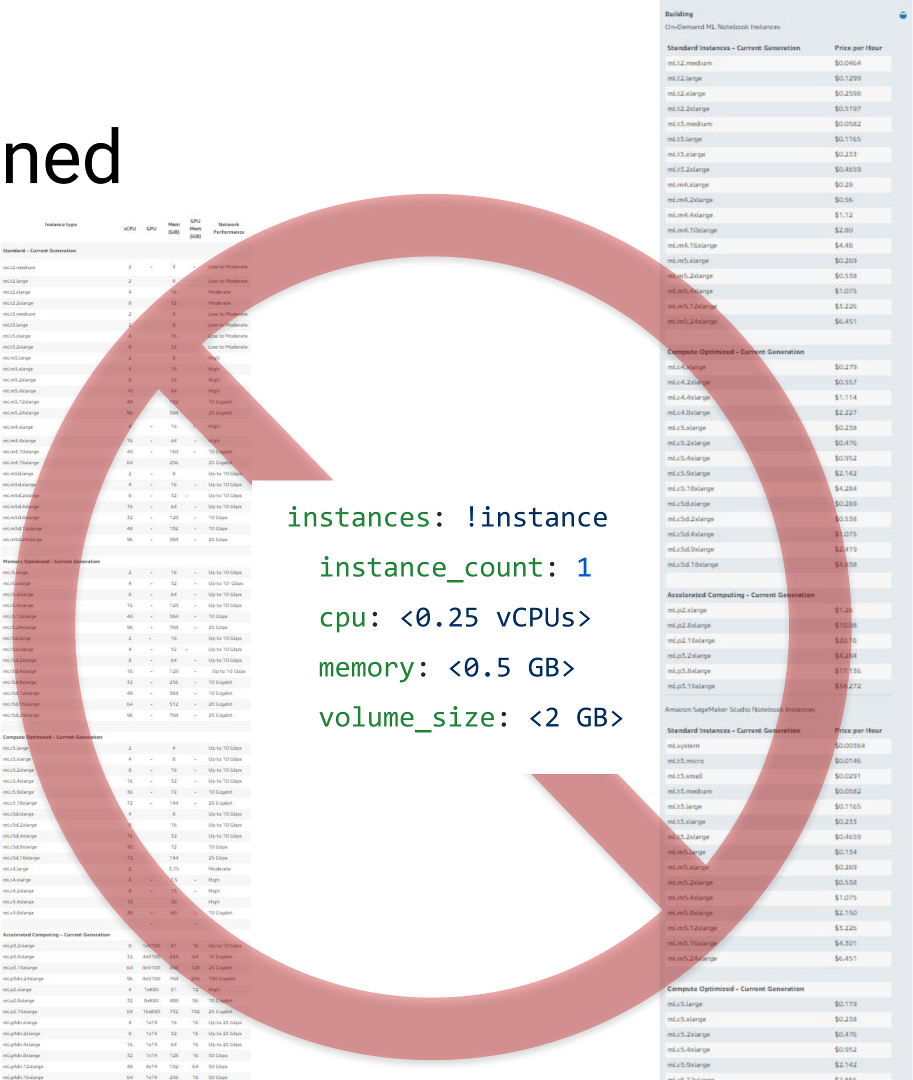
On-Demand ML Notebook Instances

Standard Instances - Current Generation	Price per Hour
ml.t2.medium	$0.0464
ml.t2.large	$0.1299
ml.t2.xlarge	$0.2598
ml.t2.2xlarge	$0.5197
ml.t3.medium	$0.0582
ml.t3.large	$0.1165
ml.t3.xlarge	$0.233
ml.t3.2xlarge	$0.4659
ml.m4.xlarge	$0.28
ml.m4.2xlarge	$0.56
ml.m4.4xlarge	$1.12
ml.m4.10xlarge	$2.80
ml.m4.16xlarge	$4.48
ml.m5.xlarge	$0.269
ml.m5.2xlarge	$0.538
ml.m5.4xlarge	$1.075
ml.m5.12xlarge	$3.226
ml.m5.24xlarge	$6.451
Compute Optimized - Current Generation	
ml.c4.xlarge	$0.279
ml.c4.2xlarge	$0.557
ml.c4.4xlarge	$1.114
ml.c4.8xlarge	$2.227
ml.c5.xlarge	$0.238
ml.c5.2xlarge	$0.476
ml.c5.4xlarge	$0.952
ml.c5.9xlarge	$2.142
ml.c5.18xlarge	$4.284
ml.c5d.xlarge	$0.269
ml.c5d.2xlarge	$0.538
ml.c5d.4xlarge	$1.075
ml.c5d.9xlarge	$2.419
ml.c5d.18xlarge	$4.838
Accelerated Computing - Current Generation	
ml.p2.xlarge	$1.26
ml.p2.8xlarge	$10.08
ml.p2.16xlarge	$20.16
ml.p3.2xlarge	$4.284
ml.p3.8xlarge	$17.136
ml.p3.16xlarge	$34.272

Amazon SageMaker Studio Notebook Instances

Standard Instances - Current Generation	Price per Hour
ml.system	$0.00364
ml.t3.micro	$0.0146
ml.t3.small	$0.0291
ml.t3.medium	$0.0582
ml.t3.large	$0.1165
ml.t3.xlarge	$0.233
ml.t3.2xlarge	$0.4659
ml.m5.large	$0.134
ml.m5.xlarge	$0.269
ml.m5.2xlarge	$0.538
ml.m5.4xlarge	$1.075
ml.m5.8xlarge	$2.150
ml.m5.12xlarge	$3.226
ml.m5.16xlarge	$4.301
ml.m5.24xlarge	$6.451
Compute Optimized - Current Generation	
ml.c5.large	$0.119
ml.c5.xlarge	$0.238
ml.c5.2xlarge	$0.476
ml.c5.4xlarge	$0.952
ml.c5.9xlarge	$2.142

Lessons learned



Instance type	vCPU	GPU	Mem (GiB)	GPU Mem (GiB)	Network Performance
Standard – Current Generation					
ml.t2.medium	2	-	4	-	Low to Moderate
ml.t2.large	2		8		Low to Moderate
ml.t2.xlarge	4		16		Moderate
ml.t2.2xlarge	8		32		Moderate
ml.t3.medium	2		4		Low to Moderate
ml.t3.large	2		8		Low to Moderate
ml.t3.xlarge	4		16		Low to Moderate
ml.t3.2xlarge	8		32		Low to Moderate
ml.m5.large	2		8		High
ml.m5.xlarge	4		16		High
ml.m5.2xlarge	8		32		High
ml.m5.4xlarge	16		64		High
ml.m5.12xlarge	48		192		10 Gigabit
ml.m5.24xlarge	96		384		25 Gigabit
ml.m4.xlarge	4	-	16	-	High
ml.m4.4xlarge	16	-	64	-	High
ml.m4.10xlarge	40	-	160	-	10 Gigabit
ml.m4.16xlarge	64		256		25 Gigabit
ml.m5d.large	2	-	8		Up to 10 Gbps
ml.m5d.xlarge	4	-	16	-	Up to 10 Gbps
ml.m5d.2xlarge	8	-	32	-	Up to 10 Gbps
ml.m5d.4xlarge	16	-	64	-	Up to 10 Gbps
ml.m5d.8xlarge	32	-	128	-	10 Gbps
ml.m5d.12xlarge	48	-	192	-	10 Gbps
ml.m5d.24xlarge	96	-	384	-	25 Gbps
Memory Optimized – Current Generation					
ml.r5.large	2	-	16	-	Up to 10 Gbps
ml.r5.xlarge	4	-	32	-	Up to 10 Gbps
ml.r5.2xlarge	8	-	64	-	Up to 10 Gbps
ml.r5.4xlarge	16	-	128	-	Up to 10 Gbps
ml.r5.12xlarge	48	-	384	-	10 Gbps
ml.r5.24xlarge	96	-	768	-	25 Gbps
ml.r5d.large	2	-	16		Up to 10 Gbps
ml.r5d.xlarge	4	-	32	-	Up to 10 Gbps
ml.r5d.2xlarge	8	-	64	-	Up to 10 Gbps
ml.r5d.4xlarge	16	-	128	-	Up to 10 Gbps
ml.r5d.8xlarge	32	-	256	-	10 Gigabit
ml.r5d.12xlarge	48	-	384	-	10 Gigabit
ml.r5d.16xlarge	64	-	512	-	20 Gigabit
ml.r5d.24xlarge	96	-	768	-	25 Gigabit
Compute Optimized – Current Generation					
ml.c5.large	2		4		Up to 10 Gbps
ml.c5.xlarge	4	-	8	-	Up to 10 Gbps
ml.c5.2xlarge	8	-	16	-	Up to 10 Gbps
ml.c5.4xlarge	16	-	32	-	Up to 10 Gbps
ml.c5.9xlarge	36	-	72	-	10 Gigabit
ml.c5.18xlarge	72	-	144	-	25 Gigabit
ml.c5d.xlarge	4		8		Up to 10 Gbps
ml.c5d.2xlarge	8		16		Up to 10 Gbps
ml.c5d.4xlarge	16		32		Up to 10 Gbps
ml.c5d.9xlarge	36		72		10 Gbps
ml.c5d.18xlarge	72		144		25 Gbps
ml.c4.large	2		3.75		Moderate
ml.c4.xlarge	4	-	7.5	-	High
ml.c4.2xlarge	8	-	15	-	High
ml.c4.4xlarge	16		30		High
ml.c4.8xlarge	36	-	60	-	10 Gigabit
		-		-	
Accelerated Computing – Current Generation					
ml.p3.2xlarge	8	1xV100	61	16	Up to 10 Gbps
ml.p3.8xlarge	32	4xV100	244	64	10 Gigabit
ml.p3.16xlarge	64	8xV100	488	128	25 Gigabit
ml.p3dn.24xlarge	96	8xV100	768	256	100 Gigabit
ml.p2.xlarge	4	1xK80	61	12	High
ml.p2.8xlarge	32	8xK80	488	96	10 Gigabit
ml.p2.16xlarge	64	16xK80	732	192	25 Gigabit
ml.g4dn.xlarge	4	1xT4	16	16	Up to 25 Gbps
ml.g4dn.2xlarge	8	1xT4	32	16	Up to 25 Gbps
ml.g4dn.4xlarge	16	1xT4	64	16	Up to 25 Gbps
ml.g4dn.8xlarge	32	1xT4	128	16	50 Gbps
ml.g4dn.12xlarge	48	4xT4	192	64	50 Gbps
ml.g4dn.16xlarge	64	1xT4	256	16	50 Gbps

```
instances: !instance
  instance_count: 1
  cpu: <0.25 vCPUs>
  memory: <0.5 GB>
  volume_size: <2 GB>
```

Building

On-Demand ML Notebook Instances

Standard Instances - Current Generation	Price per Hour
ml.t2.medium	$0.0464
ml.t2.large	$0.1299
ml.t2.xlarge	$0.2598
ml.t2.2xlarge	$0.5197
ml.t3.medium	$0.0582
ml.t3.large	$0.1165
ml.t3.xlarge	$0.233
ml.t3.2xlarge	$0.4659
ml.m4.xlarge	$0.28
ml.m4.2xlarge	$0.56
ml.m4.4xlarge	$1.12
ml.m4.10xlarge	$2.80
ml.m4.16xlarge	$4.48
ml.m5.xlarge	$0.269
ml.m5.2xlarge	$0.538
ml.m5.4xlarge	$1.075
ml.m5.12xlarge	$3.226
ml.m5.24xlarge	$6.451
Compute Optimized - Current Generation	
ml.c4.xlarge	$0.279
ml.c4.2xlarge	$0.557
ml.c4.4xlarge	$1.114
ml.c4.8xlarge	$2.227
ml.c5.xlarge	$0.238
ml.c5.2xlarge	$0.476
ml.c5.4xlarge	$0.952
ml.c5.9xlarge	$2.142
ml.c5.18xlarge	$4.284
ml.c5d.xlarge	$0.269
ml.c5d.2xlarge	$0.538
ml.c5d.4xlarge	$1.075
ml.c5d.9xlarge	$2.419
ml.c5d.18xlarge	$4.838
Accelerated Computing - Current Generation	
ml.p2.xlarge	$1.26
ml.p2.8xlarge	$10.08
ml.p2.16xlarge	$20.16
ml.p3.2xlarge	$4.284
ml.p3.8xlarge	$17.136
ml.p3.16xlarge	$34.272

Amazon SageMaker Studio Notebook Instances

Standard Instances - Current Generation	Price per Hour
ml.system	$0.00364
ml.t3.micro	$0.0146
ml.t3.small	$0.0291
ml.t3.medium	$0.0582
ml.t3.large	$0.1165
ml.t3.xlarge	$0.233
ml.t3.2xlarge	$0.4659
ml.m5.large	$0.134
ml.m5.xlarge	$0.269
ml.m5.2xlarge	$0.538
ml.m5.4xlarge	$1.075
ml.m5.8xlarge	$2.150
ml.m5.12xlarge	$3.226
ml.m5.16xlarge	$4.301
ml.m5.24xlarge	$6.451
Compute Optimized - Current Generation	
ml.c5.large	$0.119
ml.c5.xlarge	$0.238
ml.c5.2xlarge	$0.476
ml.c5.4xlarge	$0.952
ml.c5.9xlarge	$2.142

Lessons learned

Airflow:

- Hosting our own stack
- Deployment interruptions
- Not all contributions created equal

Questions?

H. Chase Stevens
Principal Data Science Engineer, teikametrics
Boston, MA
chase@chasestevens.com
@hchasestevens

https://www.teikametrics.com/company.html#careers

Europython 2020