亚马逊云科技

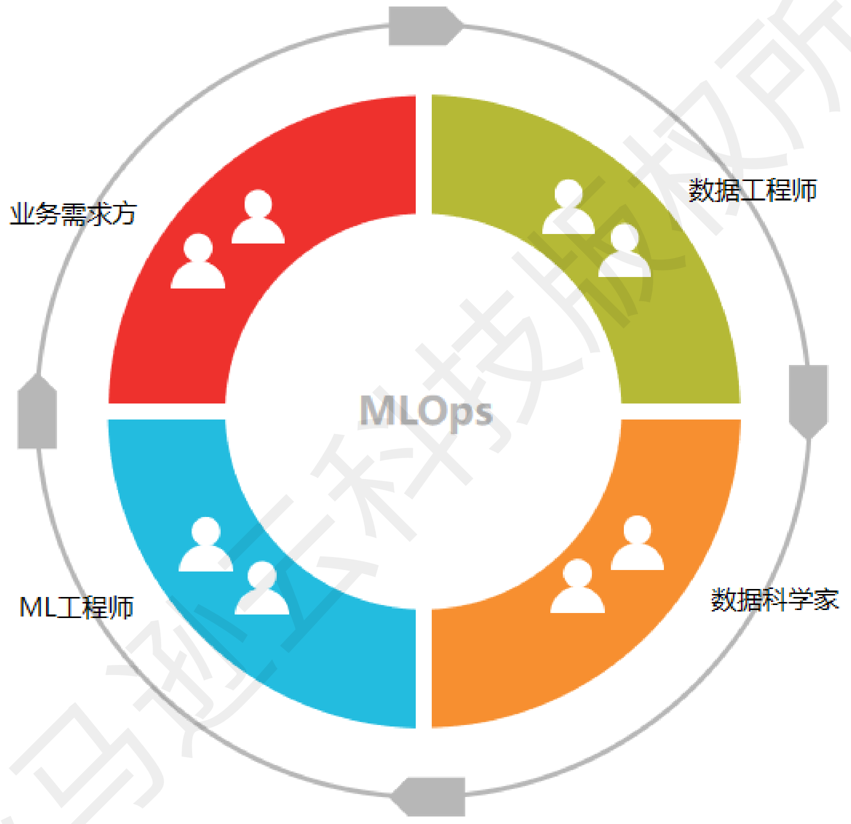
关于亚马逊团队实现MLOps这件事儿

郭韧

亚马逊云科技 高级技术专家

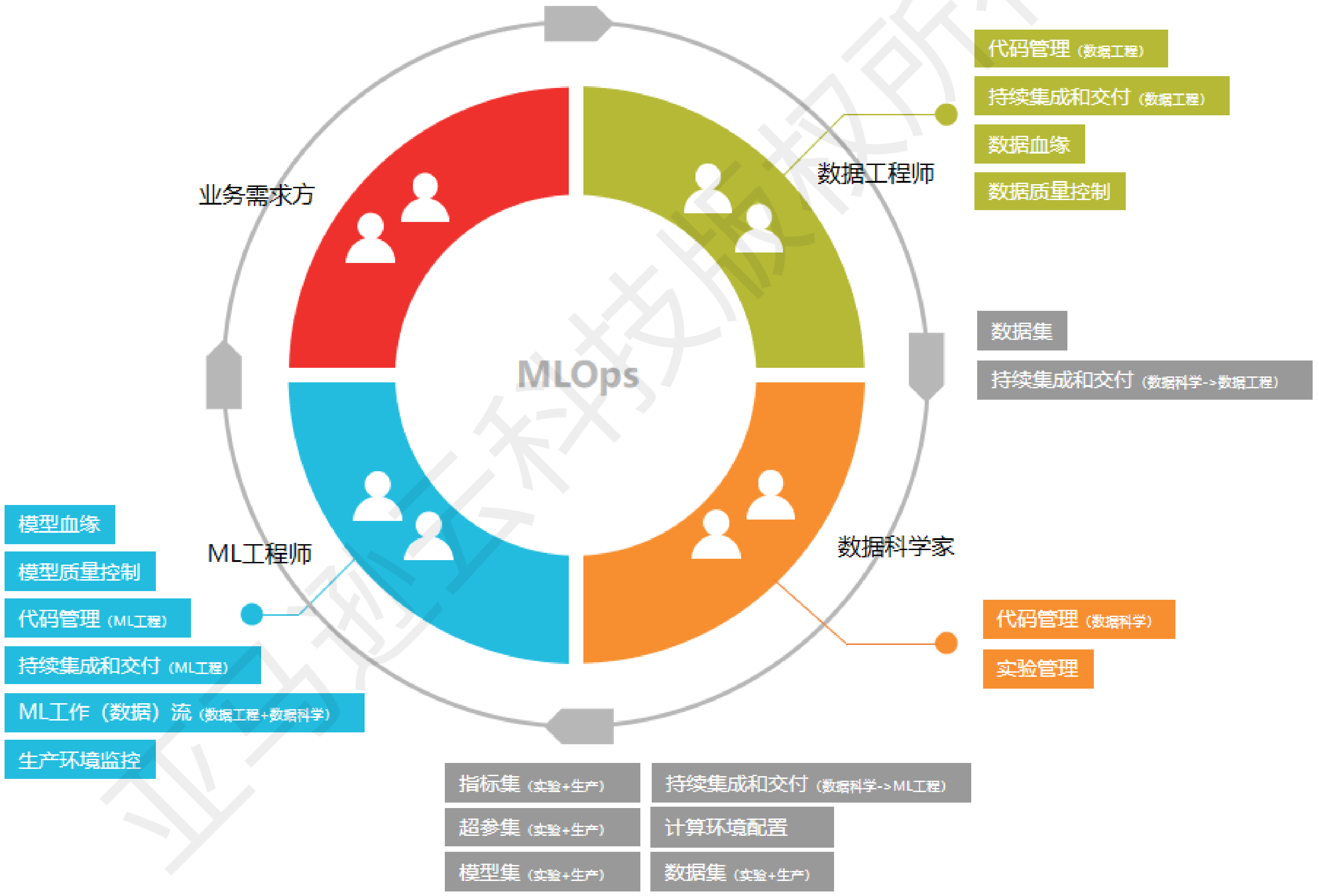
亚马逊云科技版权所有

MLOps是什么



亚马逊云科技版权所有

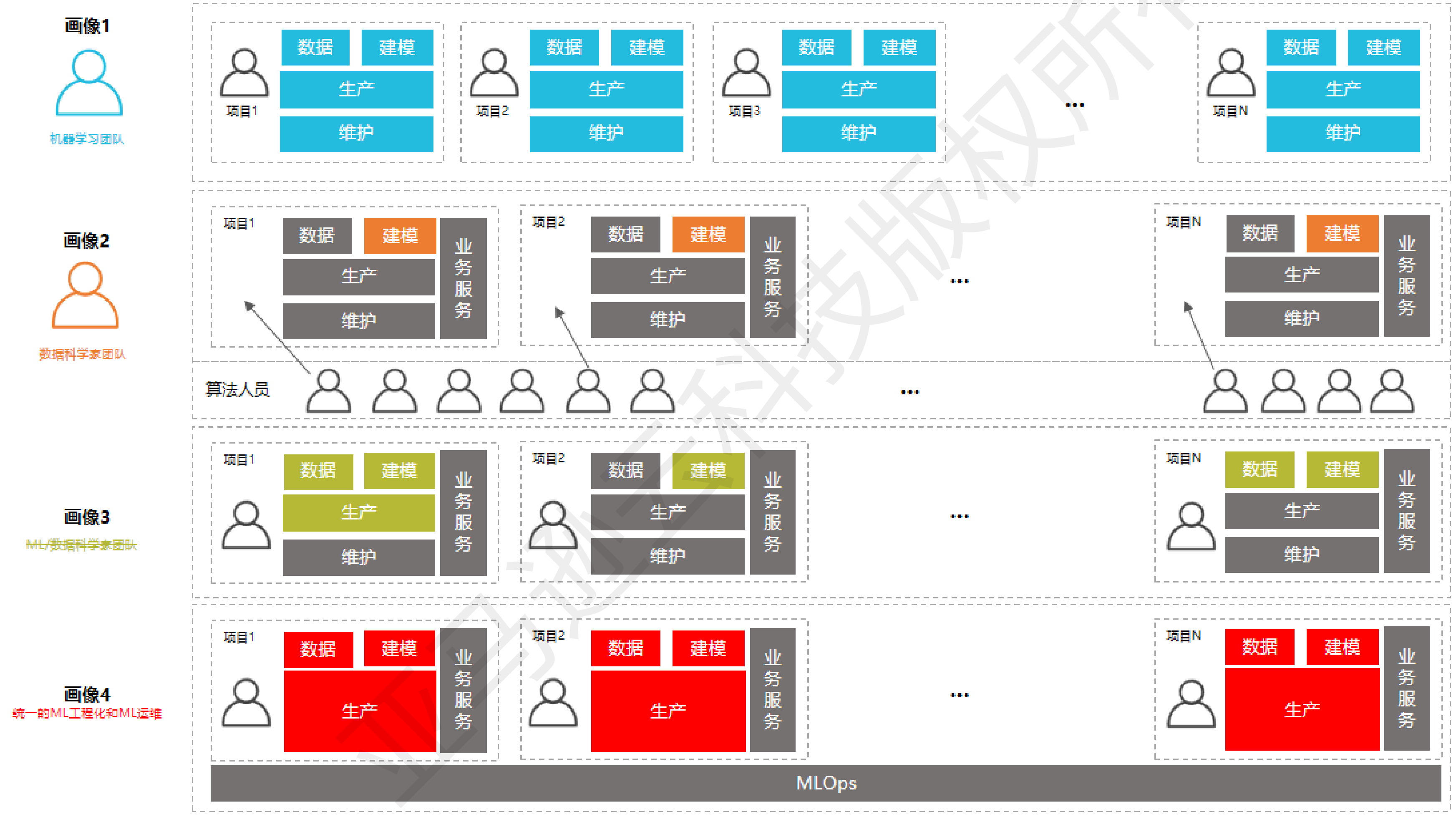
MLOps是什么



亚马逊云科技版权所有

MLOps能带来什么

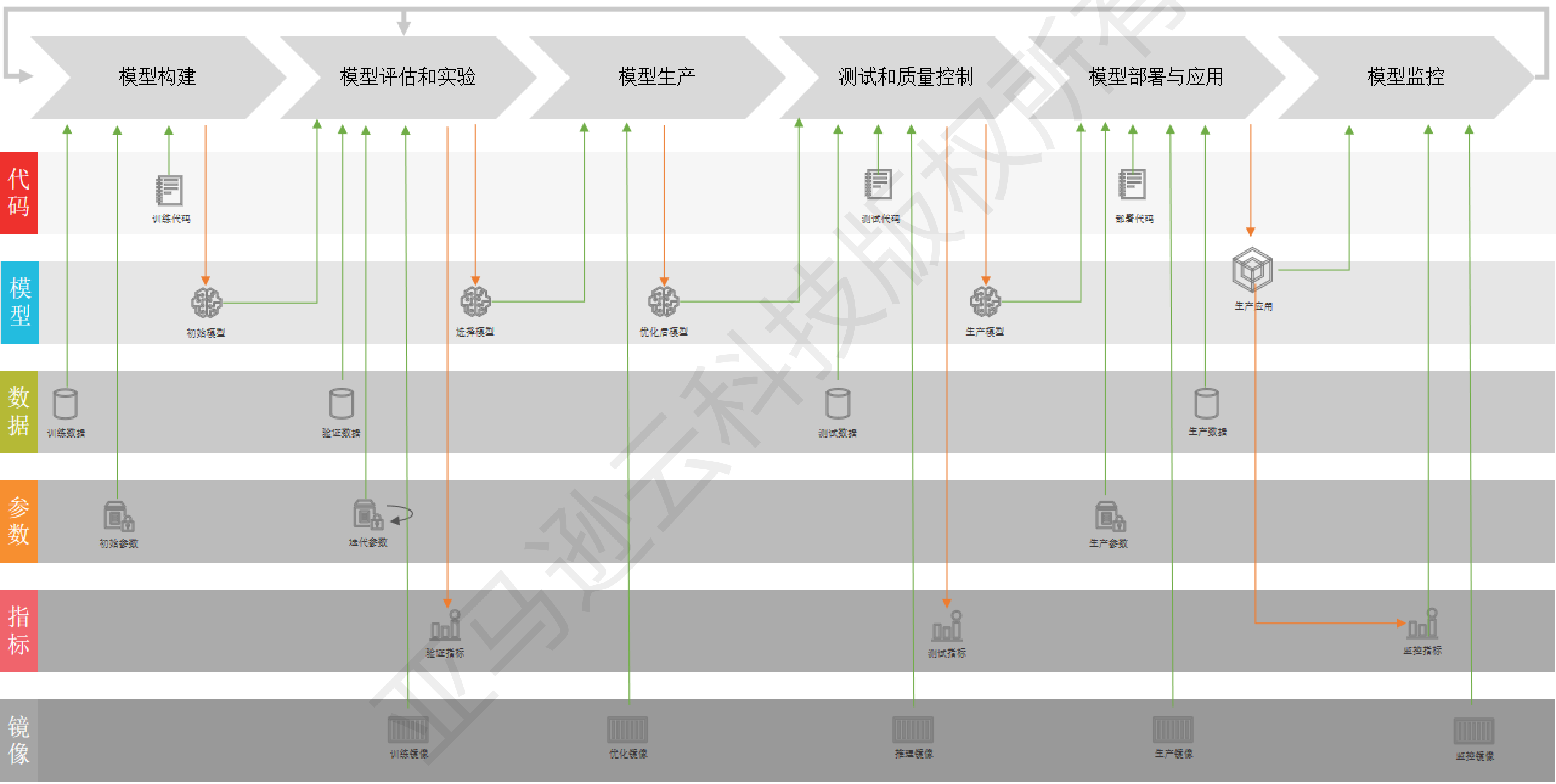
亚马逊云科技



亚马逊云科技版权所有

模型开发生命周期中的技术构件（数据科学家视角）

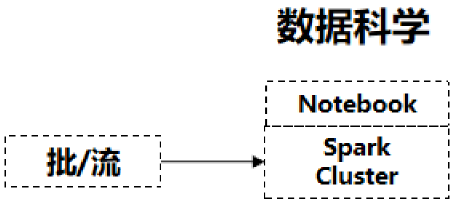
亚马逊云科技

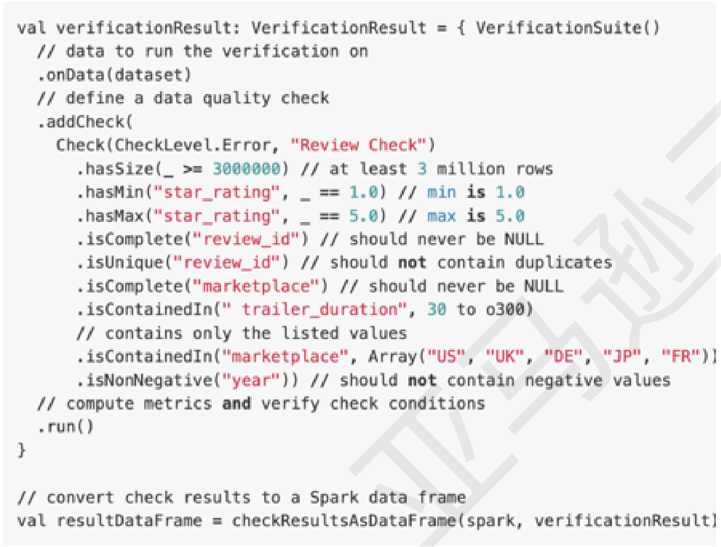


亚马逊云科技版权所有

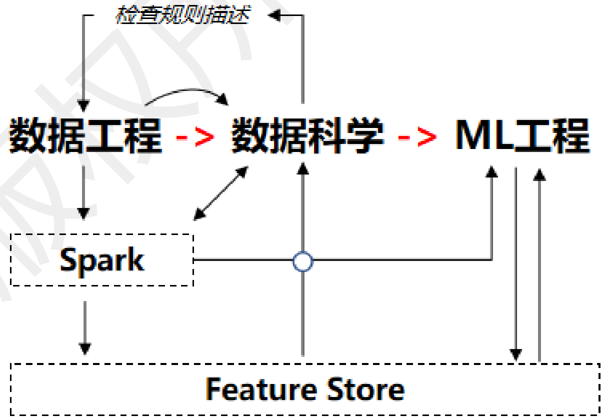
1-特征治理

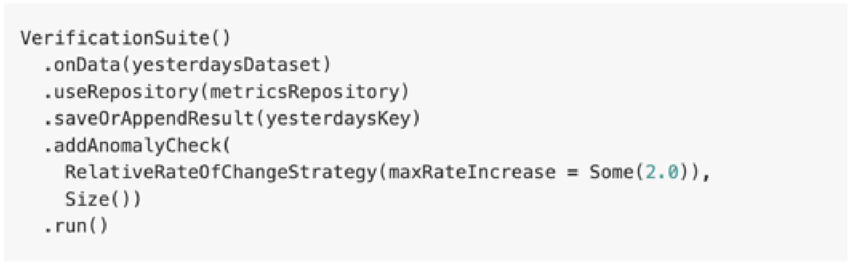
亚马逊云科技





```
val verificationResult: VerificationResult = { VerificationSuite()
  // data to run the verification on
  .onData(dataset)
  // define a data quality check
  .addCheck(
    Check(CheckLevel.Error, "Review Check")
      .hasSize(_ >= 3000000) // at least 3 million rows
      .hasMin("star_rating", _ == 1.0) // min is 1.0
      .hasMax("star_rating", _ == 5.0) // max is 5.0
      .isComplete("review_id") // should never be NULL
      .isUnique("review_id") // should not contain duplicates
      .isComplete("marketplace") // should never be NULL
      .isContainedIn(" trailer_duration", 30 to o300)
      // contains only the listed values
      .isContainedIn("marketplace", Array("US", "UK", "DE", "JP", "FR")
      .isNonNegative("year")) // should not contain negative values
  // compute metrics and verify check conditions
  .run()
}

// convert check results to a Spark data frame
val resultDataFrame = checkResultsAsDataFrame(spark, verificationResult
```



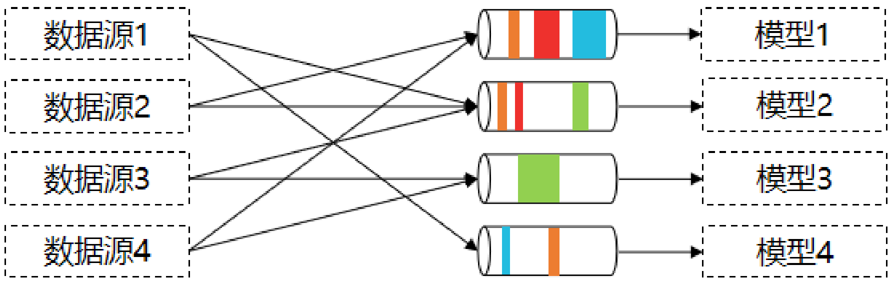


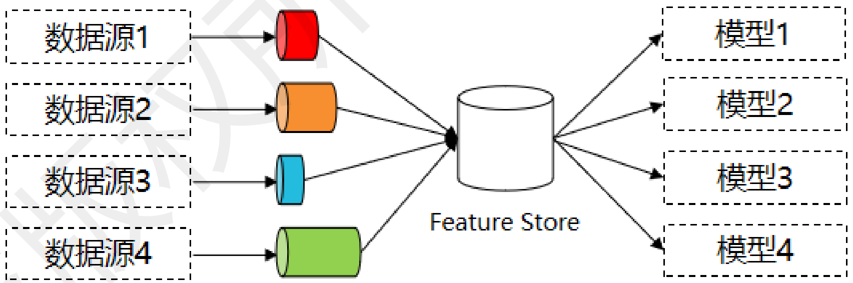
```
VerificationSuite()
  .onData(yesterdaysDataset)
  .useRepository(metricsRepository)
  .saveOrAppendResult(yesterdaysKey)
  .addAnomalyCheck(
    RelativeRateOfChangeStrategy(maxRateIncrease = Some(2.0)),
    Size())
  .run()
```

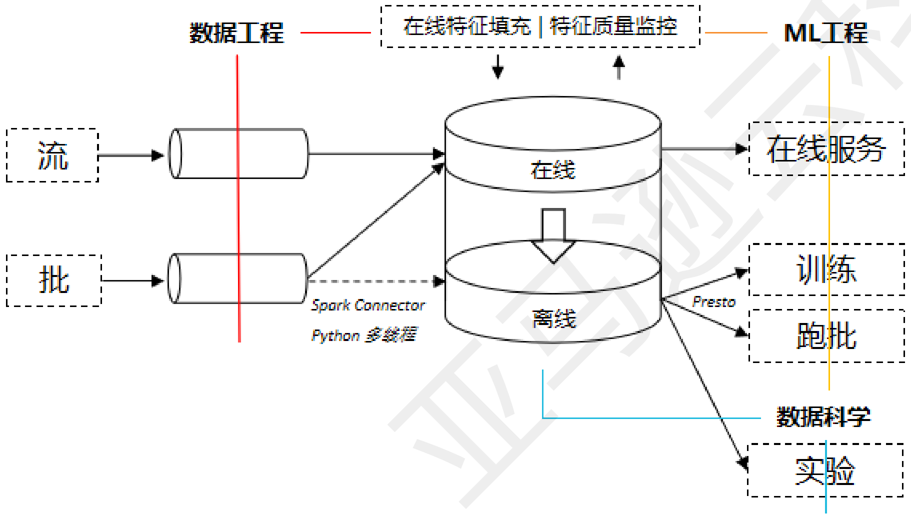
亚马逊云科技版权所有

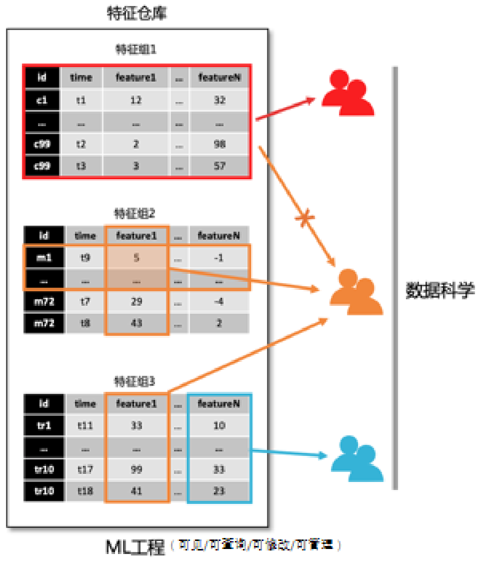
1-特征治理

亚马逊云科技









特征组1

id	time	feature1	...	featureN
c1	t1	12	...	32
...	...	...	...	...
c99	t2	2	...	98
c99	t3	3	...	57

特征组2

id	time	feature1	...	featureN
m1	t9	5	...	-1
...	...	...	...	...
m72	t7	29	...	-4
m72	t8	43	...	2

特征组3

id	time	feature1	...	featureN
tr1	t11	33	...	10
...	...	...	...	...
tr10	t17	99	...	33
tr10	t18	41	...	23

ML工程（可见/可查询/可修改/可管理）

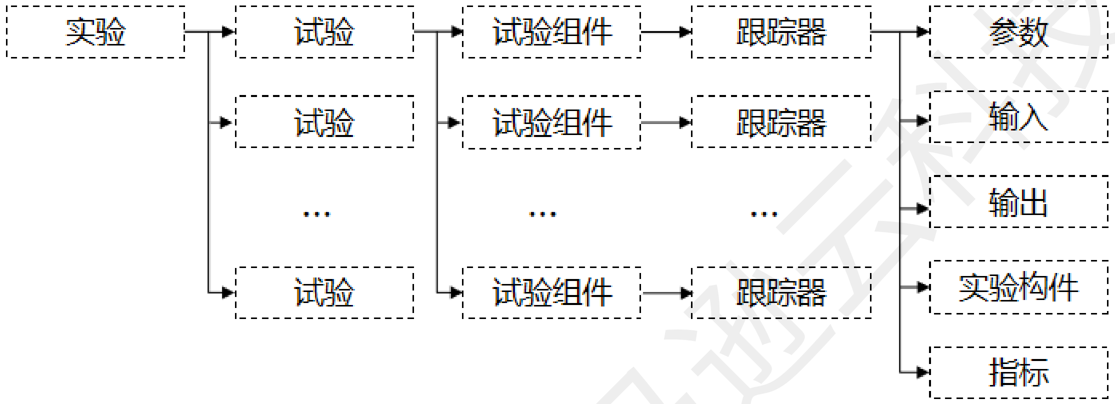
亚马逊云科技版权所有

2-机器学习实验管理

亚马逊云科技

数据科学 - 笔记本 | 训练代码 | 推理代码 | 模型对象 | 配置信息 | 实验指标

ML工程 - 版本管理 | 任务管理 | 模型管理



```
_experiment = Experiment.create(experiment_name = "ex-{}".format(strftime("%Y-%m-%d-%H-%M-%S")),
                                description = "My experiment",
                                tags = [{'Key': 'my-experiments', 'Value': '1st'}])
```

```
_trial = Trial.create(
        trial_name='trial-'+ts,
        experiment_name=_experiment.experiment_name,
    )
```

```
with Tracker.create(display_name="Preprocessing", ) as tracker:
    tracker.log_parameters({
        'train_test_split': 0.8
    })
    tracker.log_input(name='raw data', media_type='s3/uri', value=source_url)
    tracker.log_output(name='preprocessed data', media_type='s3/uri', value=processed_data_path)
    tracker.log_artifact(name='preprocessors', media_type='s3/uri', file_path='preprocessors.pickle')
    tracker.log_parameters({
        "normalization_mean": 0.1307,
        "normalization_std": 0.3081,
    })
```

```
_estimator.fit(processed_data_path,
                job_name=job_name,
                wait=False,
                experiment_config={
                                'ExperimentName': _experiment.experiment_name,
                                'TrialName': abalone_trial.trial_name,
                                'TrialComponentDisplayName': 'Training',
                                })
```

亚马逊云科技版权所有

2-机器学习实验管理

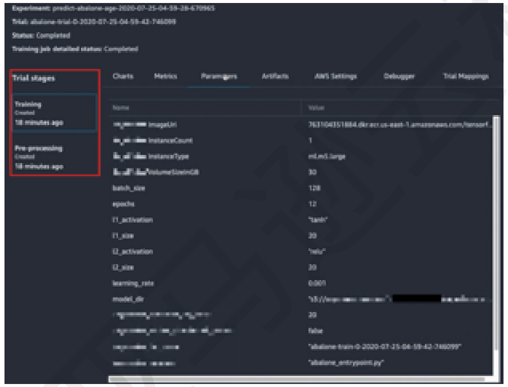
亚马逊云科技

亚马逊云科技版权所有

实验查询和比较

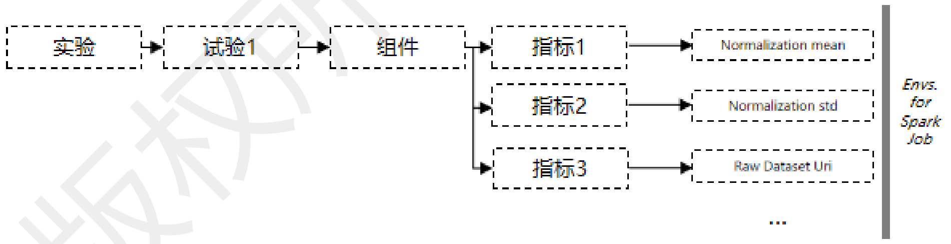
```
search_expression = {
    "Filters": [
        {
            "Name": "DisplayName",
            "Operator": "Equals",
            "Value": "Training",
        }
    ],
}

trial_component_analytics = ExperimentAnalytics(
    experiment_name=_experiment.experiment_name,
    search_expression=search_expression,
    sort_by="metrics.test:accuracy.max",
    sort_order="Descending",
    metric_names=["test:accuracy"],
    parameter_names=["hidden_channels", "epochs", "dropout", "optimizer"],
)

analytic_table = analytic_table = trial_component_analytics.dataframe()
```

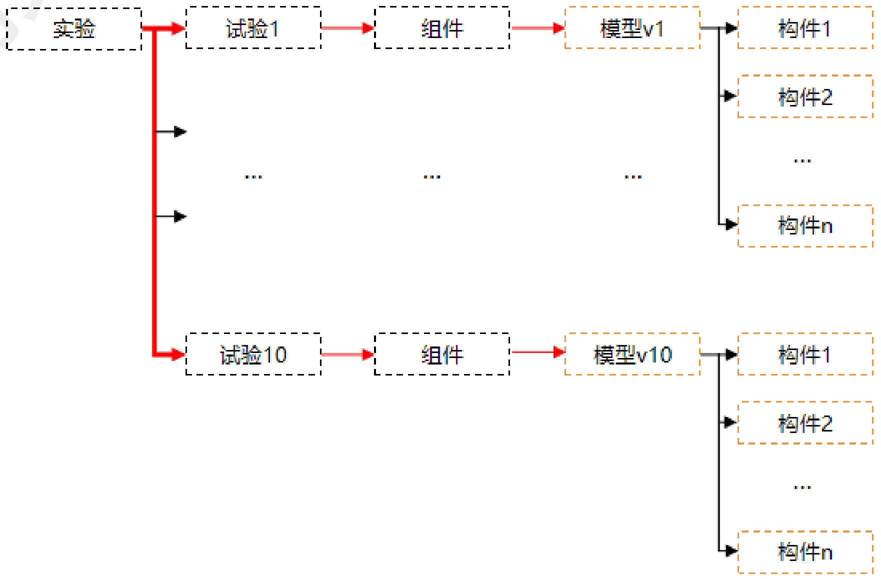
```
for col in analytic_table.columns:
    print(col)

TrialComponentName
DisplayName
SourceArn
dropout
epochs
hidden_channels
optimizer
test:accuracy - Min
test:accuracy - Max
test:accuracy - Avg
test:accuracy - StdDev
test:accuracy - Last
test:accuracy - Count
```


Experiment: predict-abalone-age-2020-07-25-04-59-28-670965
Trial: abalone-trial-0-2020-07-25-04-59-42-746099
Status: Completed
Training job detailed status: Completed

Name	Value
[illegible] ImageUri	763104351884.dkr.ecr.us-east-1.amazonaws.com/tensorf...
[illegible] InstanceCount	1
[illegible] InstanceType	ml.m5.large
[illegible] VolumeSizeInGB	30
batch_size	128
epochs	12
l1_activation	"tanh"
l1_size	20
l2_activation	"relu"
l2_size	20
learning_rate	0.001
model_dir	"s3://[illegible] [redacted] [illegible]
[illegible]	20
[illegible]	false
[illegible]	"abalone-train-0-2020-07-25-04-59-42-746099"
[illegible]	"abalone_entrypoint.py"

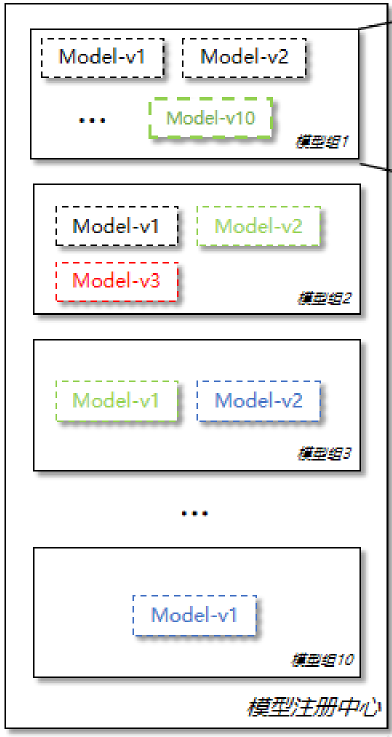
工程上线



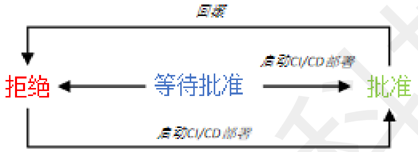


3-模型治理

ML工程



Version	Name	Tags	Description	Activity	Serving Image	Model Location	Modified on	Modified by	Create on	Pipeline	Execution	Metrics		
												Test:acc	Test:f1	...
1	m1	-	-	-	-	-	-	-	-	-	-	-	-	-
2	m2	-	-	-	-	-	-	-	-	-	-	-	-	-
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
10	m10	-	-	-	-	-	-	-	-	-	-	-	-	-



Event	Comment	Modified by	Last modification
等待批准	-	-	-
批准	-	-	-
...	...	...	...
拒绝	-	-	-

- *模型编录*
- *版本控制*
- *元数据管理*
- *上线状态管理*
- *自动化部署*

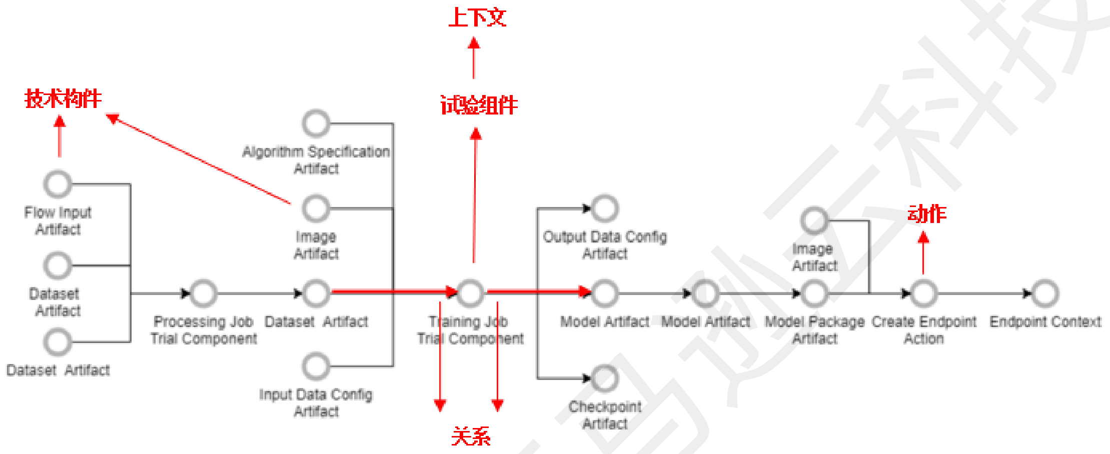
亚马逊云科技版权所有

3-模型治理

亚马逊云科技

模型血缘

1. 保留和探索模型的运行记录
2. 满足审计和合规性
3. 克隆并重新运行 ML 工作流，以便随时可以复现
4. 协同场景下，可以共享和复现工作流
5. 在生产环境中，用以trouble shooting



```python
code_location_arn = artifact.Artifact.create(
    artifact_name='source-code-location',
    source_uri='s3://...',
    artifact_type='code-location'
).artifact_arn

# Similar constructs for train_data_location_arn and test_data_location_arn

model_location_arn = artifact.Artifact.create(
    artifact_name='model-location',
    source_uri='s3://...',
    artifact_type='model-location'
).artifact_arn
```

```
启动训练任务，获取trial_component_arn
```

```python
input_artifacts = [code_location_arn, train_data_location_arn, test_data_location_arn]
for artifact_arn in input_artifacts:
    try:
        association.Association.create(
            source_arn=artifact_arn,
            destination_arn=trial_component_arn,
            association_type='ContributedTo'
        )
    except:
        logging.info('association between {} and {} already exists', artifact_arn, trial_component_arn)

output_artifacts = [model_location_arn]
for artifact_arn in output_artifacts:
    try:
         association.Association.create(
            source_arn=trial_component_arn,
            destination_arn=artifact_arn,
            association_type='Produced'
        )
    except:
        logging.info('association between {} and {} already exists', artifact_arn, trial_component_arn)
```

亚马逊云科技版权所有

3-模型治理

亚马逊云科技

模型血缘

```
trial_component_arn = _client.list_associations(
    DestinationArn=endpoint_context_arn)['AssociationSummaries'][0]['SourceArn']
```

```
train_data_location_artifact_arn = _client.list_associations(
    DestinationArn=trial_component_arn, SourceType='Model')['AssociationSummaries'][0]['SourceArn']
```

```
train_data_location = _client.describe_artifact(
    ArtifactArn=train_data_location_artifact_arn)['Source']['SourceUri']
    print(train_data_location)

s3://buckeet-sample-data-us-east-2-xx/tf/predict/train
```

血缘检索

1. 对于当前的线上模型，上游都有哪些数据源?
2. 都有哪些试验参与了当前线上模型的生产?
3. 在端到端的流程中，从raw data到模型上线前都有哪些试验和环节?
4. …

```
query_filter = LineageFilter(
    entities=[LineageEntityEnum.ARTIFACT], sources=[LineageSourceEnum.DATASET]
)

query_result = LineageQuery().query(
    start_arns=[endpoint_context.context_arn],
    query_filter=query_filter,
    direction=LineageQueryDirectionEnum.ASCENDANTS,
    include_edges=False,
)

dataset_artifacts = []
for vertex in query_result.vertices:
    dataset_artifacts.append(vertex.to_lineage_object().source.source_uri)
```

亚马逊云科技版权所有

4-工作流设计

组件 - 数据处理任务（Spark Job | Sklearn | Customization)

组件 - 模型训练任务（Tensorflow | PyTorch | MXNet | XGBoost | Sklearn | Customization)

组件 - 超参调优任务

组件 - 模型创建

组件 - 模型注册

组件 - 模型批量推理

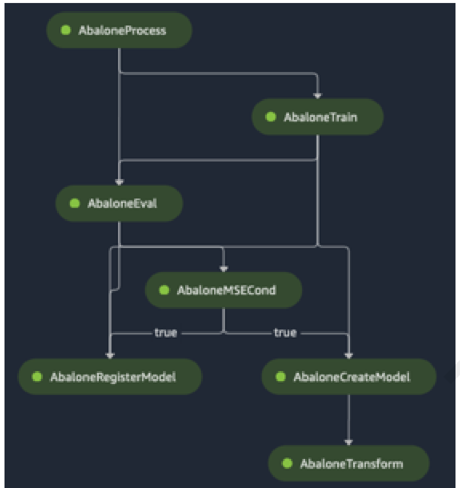
组件 - 特征质量检查

组件 - 可解释性检查

组件 - 条件分支

组件 - EMR

组件 - 回调 *(token with status)*



```
best_model = Model(
    image_uri=image_uri,
    model_data=step_tuning.get_top_model_s3_uri(
        top_k=0,
        s3_bucket=_session.default_bucket()
    ),
    ...
)
```

```
pipeline_model = PipelineModel(models=[sklearn_model,xgboost_model],role=role)

step_register = RegisterModel(
 name="AbaloneRegisterModel",
 model=pipeline_model,
 content_types=["application/json"],
 response_types=["application/json"],
 inference_instances=["ml.t2.medium", "ml.m5.xlarge"],
 transform_instances=["ml.m5.xlarge"],
 model_package_group_name='sipgroup',
)
```

```
cond_lte = ConditionLessThanOrEqualTo(
    left=JsonGet(
        step_name=step_eval.name,
        property_file=evaluation_report,
        json_path="regression_metrics.mse.value"
    ),
    right=6.0
)

step_cond = ConditionStep(
    name="AbaloneMSECond",
    conditions=[cond_lte],
    if_steps=[step_register, step_create_model, step_transform],
    else_steps=[]
)
```

ConditionEquals
ConditionGreaterThan
ConditionGreaterThanOrEqualTo
ConditionLessThan
ConditionLessThanOrEqualTo
ConditionIn
ConditionNot
ConditionOr

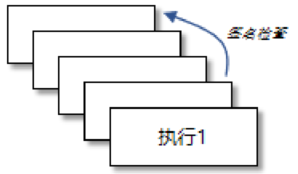
```
<step_name>.properties.<property>.<property>
step_process.properties.ProcessingOutputConfig.Outputs["train_data"].S3Output.S3Uri
```

亚马逊云科技版权所有

4-工作流设计

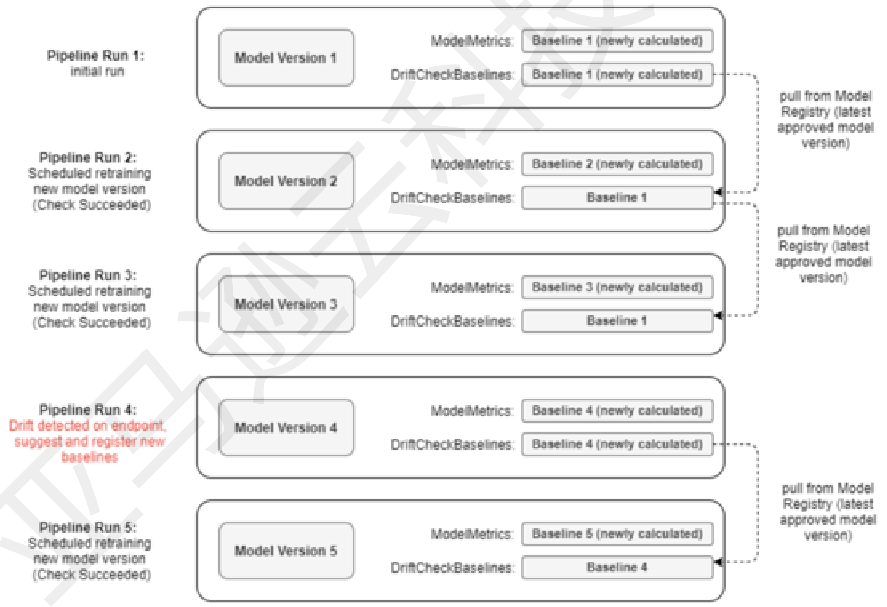
亚马逊云科技

缓存机制



```
cache_config = CacheConfig(enable_caching=True, expire_after="3D12H")
```

特征质量检查&可解释性检查中的 Baseline Evolution

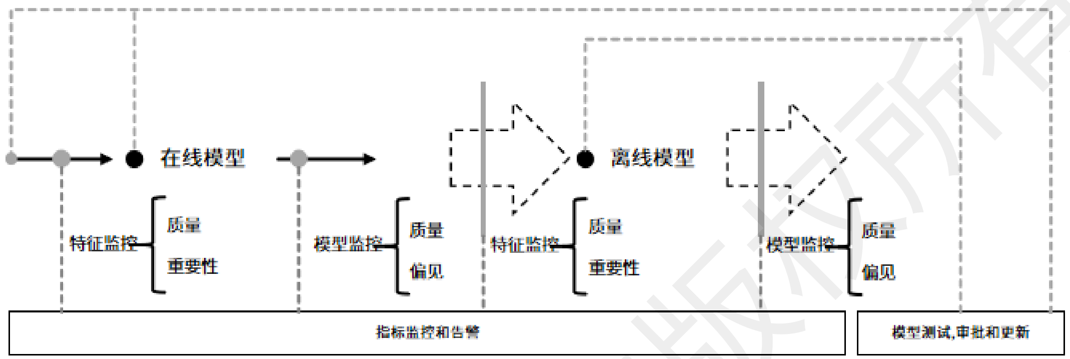


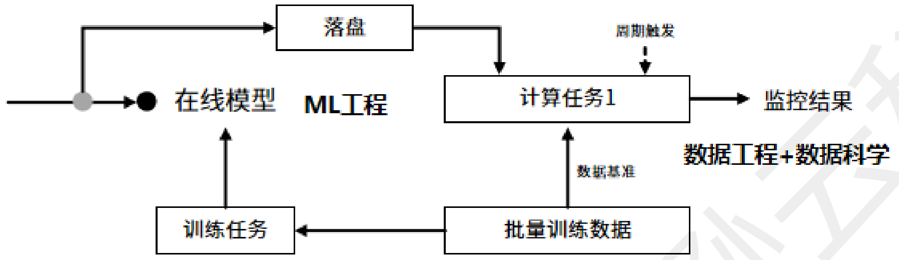
Pipeline Run 1:
initial run
Model Version 1
ModelMetrics:
Baseline 1 (newly calculated)
DriftCheckBaselines:
Baseline 1 (newly calculated)
pull from Model Registry (latest approved model version)
Pipeline Run 2:
Scheduled retraining new model version (Check Succeeded)
Model Version 2
ModelMetrics:
Baseline 2 (newly calculated)
DriftCheckBaselines:
Baseline 1
pull from Model Registry (latest approved model version)
Pipeline Run 3:
Scheduled retraining new model version (Check Succeeded)
Model Version 3
ModelMetrics:
Baseline 3 (newly calculated)
DriftCheckBaselines:
Baseline 1
Pipeline Run 4:
Drift detected on endpoint, suggest and register new baselines
Model Version 4
ModelMetrics:
Baseline 4 (newly calculated)
DriftCheckBaselines:
Baseline 4 (newly calculated)
pull from Model Registry (latest approved model version)
Pipeline Run 5:
Scheduled retraining new model version (Check Succeeded)
Model Version 5
ModelMetrics:
Baseline 5 (newly calculated)
DriftCheckBaselines:
Baseline 4

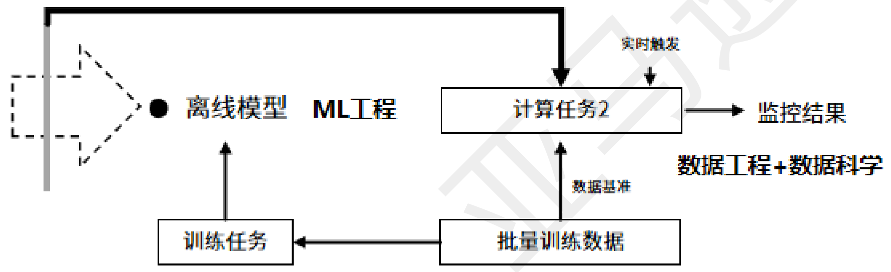
亚马逊云科技版权所有

5-模型监控

亚马逊云科技





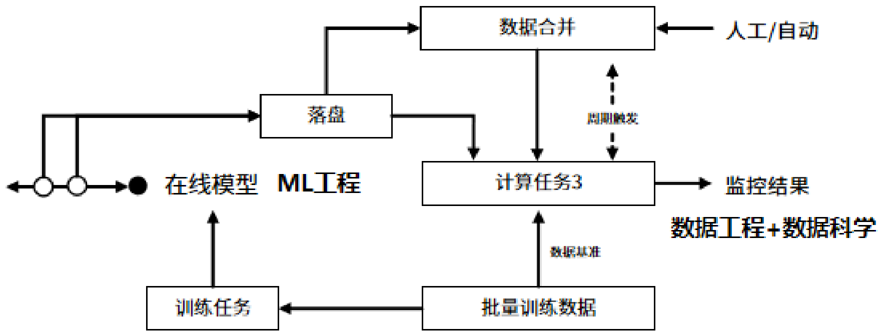


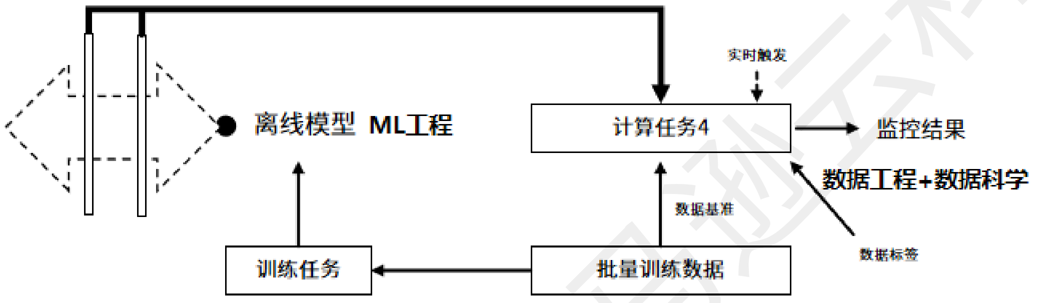
数据基准					
Fractional	空值	均值	分桶1	分位数1	特征1重要性
Integral	奇异值	期望	分桶2	分位数2	特征2重要性
		标准差	分桶3	分位数3	…
		最小值	…	…	特征N重要性
		最大值	分桶N	分位数M	特征排名指标

数据检查	
数据类型	一致性检查，阈值判定
完整度	完整比例检查，阈值判定
数据漂移	数据分布差异（距离）检查，阈值判定
…	特征列一致性检查，分类特征值检查…
特征排名指标	指标差异检查，阈值判定

亚马逊云科技版权所有

5-模型监控





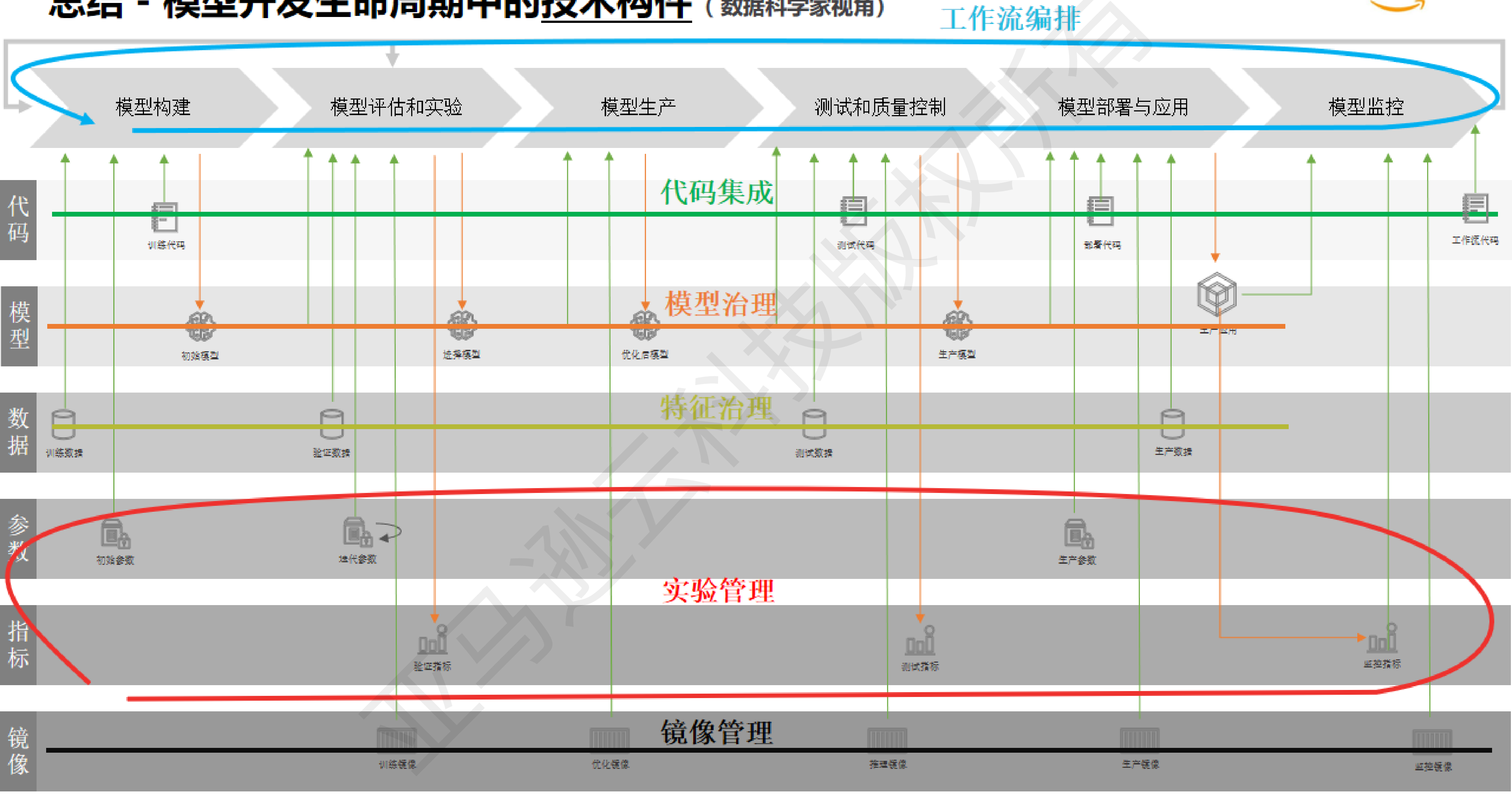
数据基准	
预测标签中正比率的差异	$n_a'^{(1)}/n_a - n_d'^{(1)}/n_d$
查全差异	$TP_a/(TP_a+FN_a) - TP_d/(TP_d+FN_d)$
查准差异	$TP_a/(TP_a+FP_a) - TP_d/(TP_d+FP_d)$
…	…

模型检查	
分类质量检查	混淆矩阵/RECALL/PRECISION/ACCURACY/TP/FP/TN/FN/AUC/F1/F2…
回归质量检查	MAE/MSE/RMSE…
预测标签中正比率的差异	差异变化距离检查，阈值判定
查全差异	
查准差异	
…	…

亚马逊云科技版权所有

总结 - 模型开发生命周期中的技术构件（数据科学家视角）

亚马逊云科技



亚马逊云科技版权所有

亚马逊云科技

谢谢

郭韧

亚马逊云科技 高级技术专家

亚马逊云科技版权所有