Reverse Engineer Machine Learning To Develop Performance Monitoring Rules

Craig Shallahamer
craig@orapub.com

ORACLE ACE Director

ORAPUB

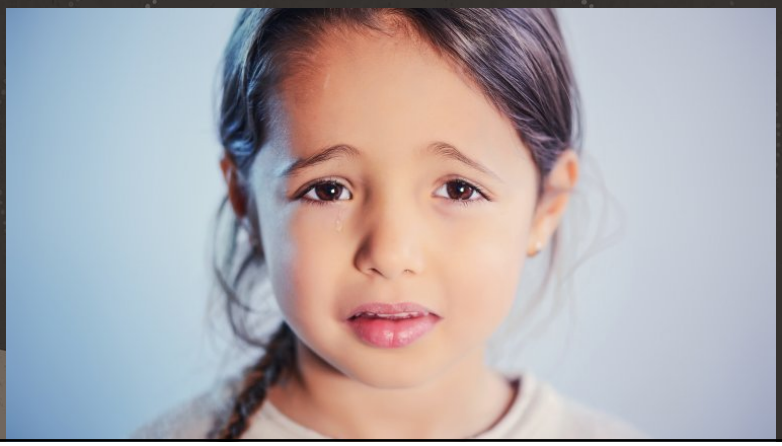
NYOUG

This presentation was given by Craig Shallahamer (craig@orapub.com) at the October 27, 2020 NYOUG/Viscosity Conference. There is likely a more recent version at www.orapub.com > Login > Presentations

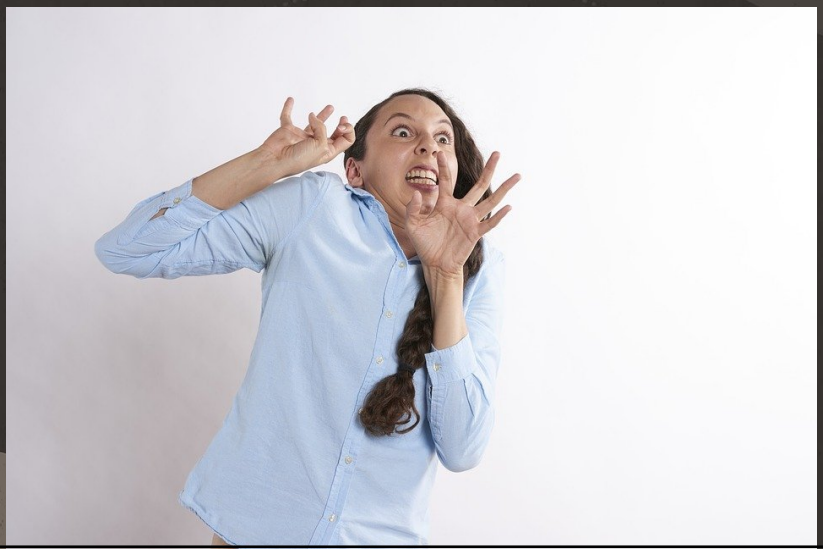
Machine Learning is all the rage!
It looks to be exciting and everyone is using it!
It being used in sales, marketing, finance, medical and geological research.
Of course, you want to get in on the fun!

But you're in IT.
While some IT organizations are investigating how to use ML, yours is NOT embracing or even considering ML.

Yet you're responsible for monitoring and keeping hundreds of big and small Oracle systems running.

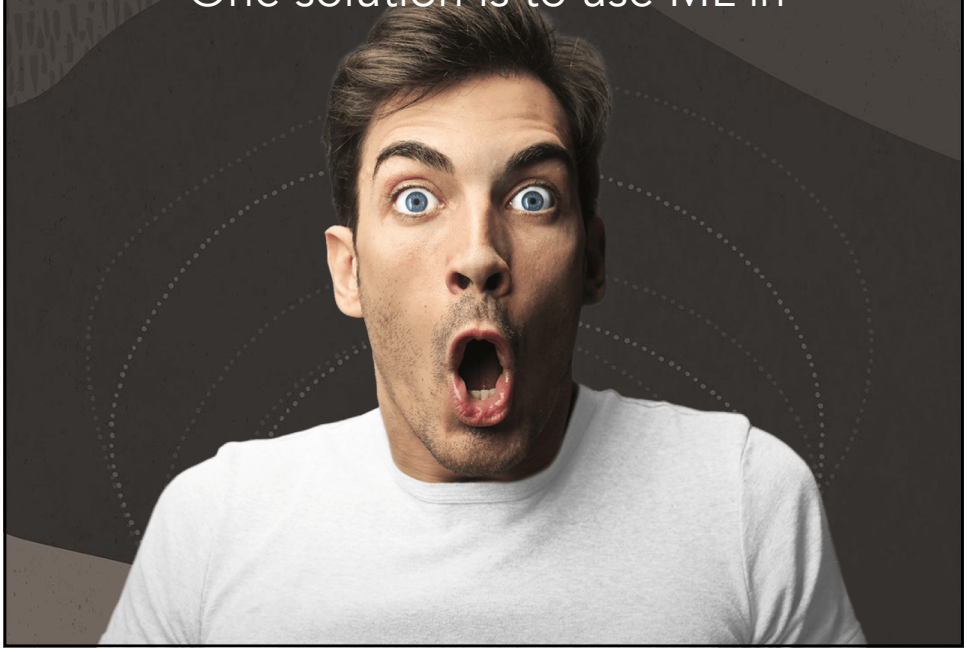


You know it's not sustainable.
You wonder…
Am I the only one in my organization that sees this?

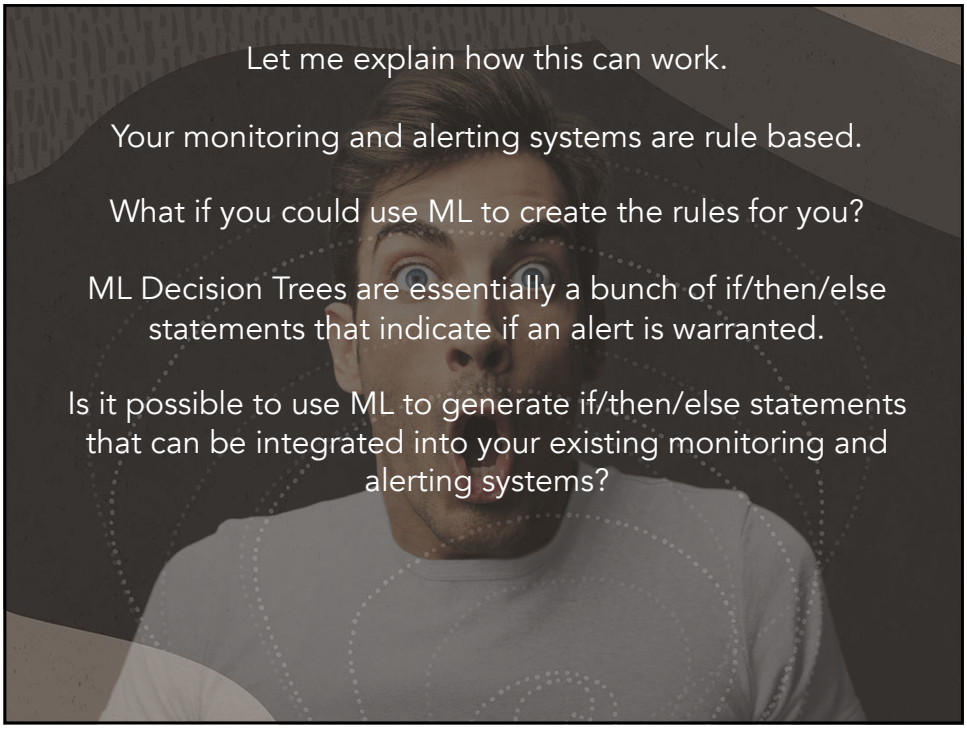


So the questions is…
Is there a way for you to investigate using ML in your work, even though your department is not interested?

One solution is to use ML in




One solution is to use ML in
"stealth mode"

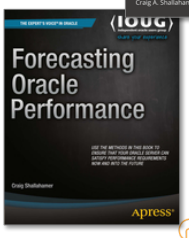

Let me explain how this can work.
Your monitoring and alerting systems are rule based.
What if you could use ML to create the rules for you?
ML Decision Trees are essentially a bunch of if/then/else statements that indicate if an alert is warranted.
Is it possible to use ML to generate if/then/else statements that can be integrated into your existing monitoring and alerting systems?

Of course!

And in this presentation, I will teach you the core building blocks to help you get started!

About Me...

- Long time Oracle DBA
- Specialize in **predictive analytics, machine learning** and Oracle performance tuning
- Performance researcher
- Blogger: A Wider View About Oracle Performance Tuning
- Author: Oracle Performance Firefighting and Forecasting Oracle Performance.
- Conference speaker
- Teacher and mentor
- Oracle ACE Director
- Quest/IOUG DBA Track Manager



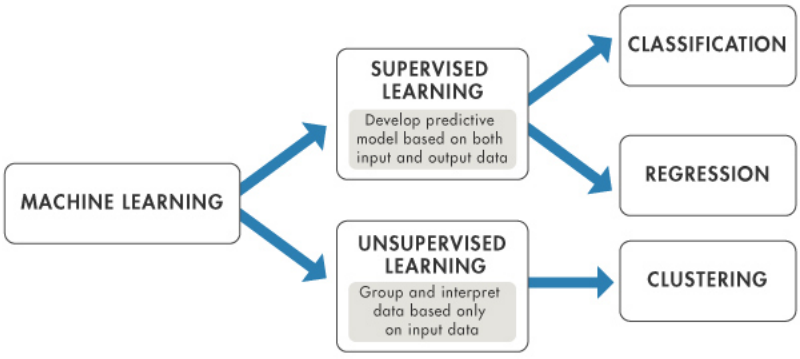
ORAPUB

OraPub works with IT to deploy machine learning into their monitoring and alerting processes.

OraPub works with Oracle DBAs empowering them to beat bots, AI, machine learning and autonomous anything.

ORAPUB

Supervised and Unsupervised Learning



ORAPUB

Supervised Vs Un-Supervised

snap_id	uc_psec	aas	trx_psec	perf
1001	2500	34.25	9.45	bad
1002	1200	14.50	6.50	good
1003	1150	16.50	16.50	good
1004	1250	18.50	9.50	good
1005	1300	24.50	2.50	????

Supervised

Will snap_id 1005 result in "the phone ringing?"

Label

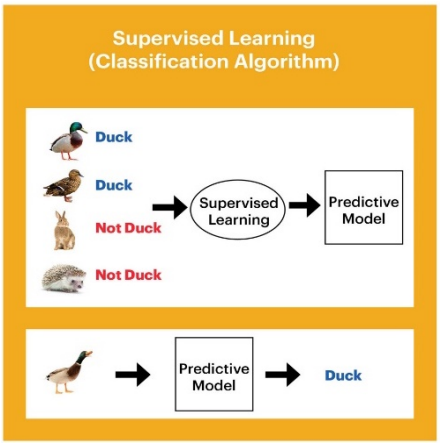
snap_id	uc_psec	aas	trx_psec
1001	2500	34.25	9.45
1002	1200	14.50	6.50
1003	1150	16.50	16.50
1004	1250	18.50	9.50
1005	1300	24.50	2.50

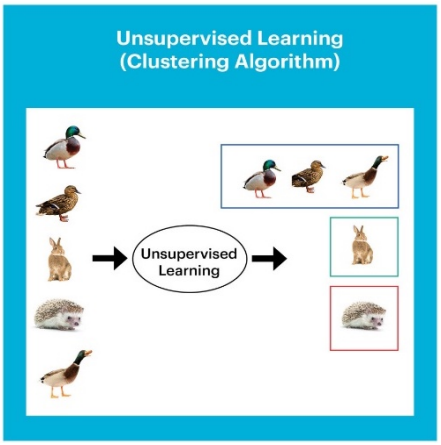
Un-Supervised

Is snap_id 1005 anomalous?

ORAPUB

Notice the difference?





ORAPUB

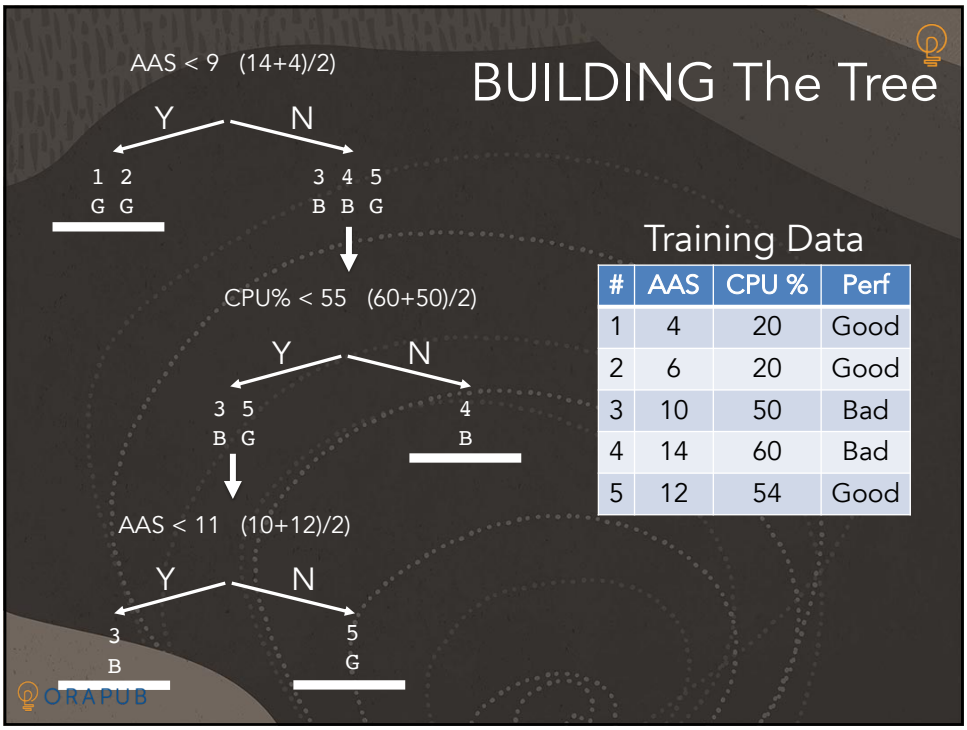
Training Data: Features and Label

Sample #	AAS	CPU %	Performance
1	4	20	**Good**
2	6	20	**Good**
3	10	50	**Bad**
4	14	60	**Bad**
5	12	54	**Good**

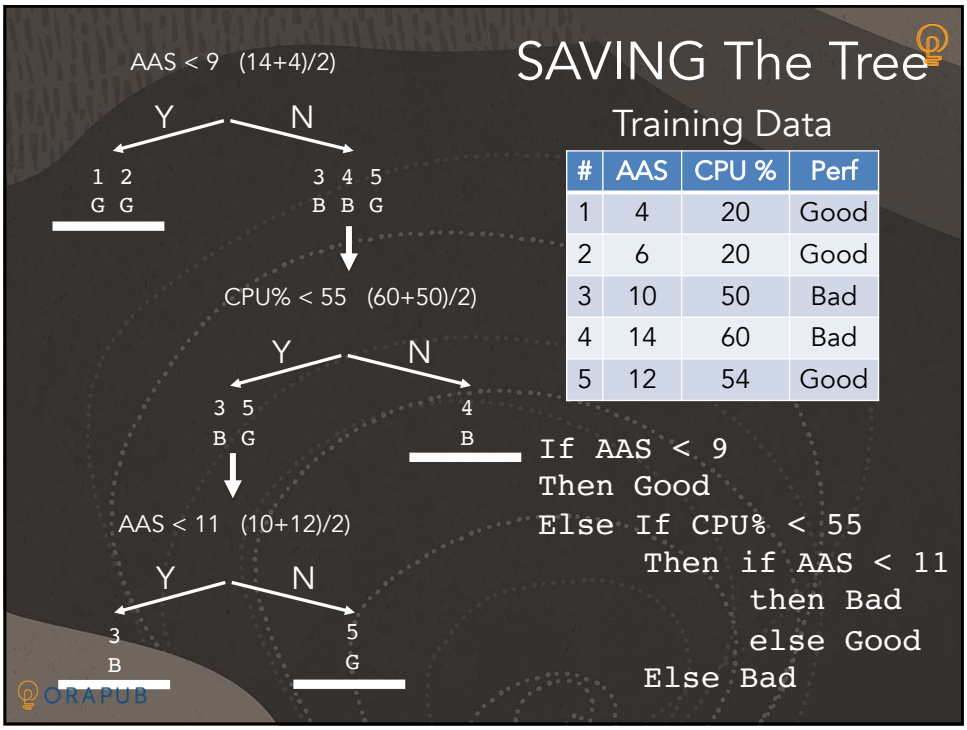
Craig's Decision Tree Algorithm Rules

1. Decisions based on features going left to right, round-robin.
2. Decision value is the remaining sample's feature data, (high+low)/2
3. Condition is less-than
4. Y branch left, N branch right

ORAPUB



This presentation was given by Craig Shallahamer (craig@orapub.com) at the October 27, 2020 NYOUG/Viscosity Conference. There is likely a more recent version at www.orapub.com > Login > Presentations



How to make our model better?

- Better accuracy determination
- Use the minimum and only most powerful features.
- Perhaps use ASH data.
- Address the "rare event" reality.
- Try other models.
- Tune the best model.
- …
- …

ORAPUB

Want my
updated slides?

`www.orapub.com >`
`Login >`
`Tools & Presentations`

ORAPUB

Super Small Dataset, 3 Features

```
In [2]: # Set Gobal Variables & Settings

        # Dataset Selection
        datasetSize = "small" #   3 features
        #datasetSize = "real" # 161 features

        # Decision Tree Model Parameters
        maxTreeDepth  = 4      # 3 nice, but may not resolve
        treeRandomNo  = 123

        # Number of "important" features to use
        numImportantFeatures = 3 # Demo: 4 real, 3 small

        # Scoring Parameters (Repeated Stratified K-Folds)
        scoringKFolds=3
        scoringRepeats=3

        os.chdir("/Users/cshallah/Desktop/DS-working")  # <<<<<---
        print(os.getcwd())

        import warnings
        warnings.filterwarnings("ignore", category=FutureWarning)
        warnings.filterwarnings("ignore", category=UserWarning)

        testing = False # for development purposesonly

        print("Done.")

        /Users/cshallah/Desktop/DS-working
        Done.
```

Super Small Dataset, 3 Features

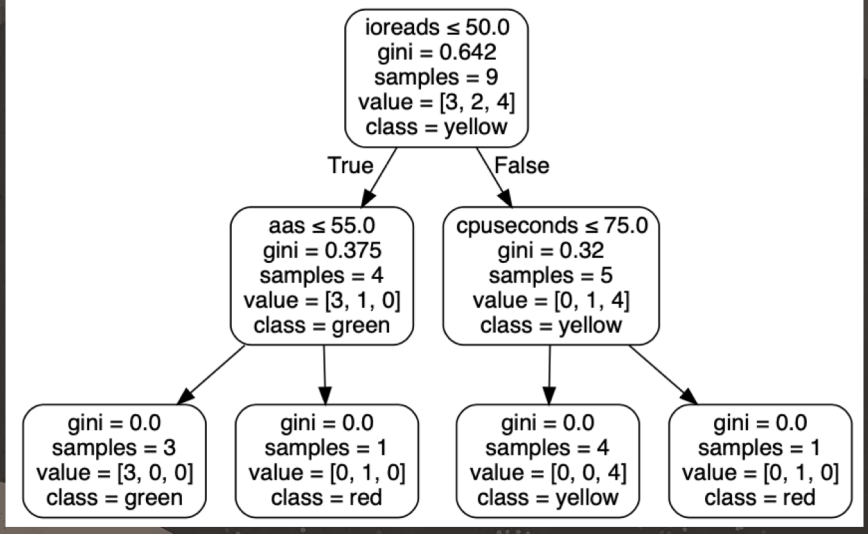
```
['cpuseconds', 'ioreads', 'aas', 'snap_id']
['snap_id', 'indicator_level']

fullDF.shape, showXfeaturesDF.shape, ticketDF2.shape (9, 5) (9, 4) (9, 2)
```

	cpuseconds	ioreads	aas	snap_id	indicator_level
0	10	10	100	1001	red
1	90	90	10	1002	red
2	60	90	10	1003	yellow
3	60	10	10	1004	green
4	60	90	9	1005	yellow
5	60	10	10	1006	green
6	60	90	9	1007	yellow
7	60	10	10	1008	green
8	60	90	9	1009	yellow

	cpuseconds	ioreads	aas	snap_id	indicator_level
0	10	10	100	1001	red
1	90	90	10	1002	red
2	60	90	10	1003	yellow
3	60	10	10	1004	green

```
(9, 3) ['cpuseconds', 'ioreads', 'aas']

|--- ioreads <= 50.00
|    |--- aas <= 55.00
|    |    |--- class: green
|    |--- aas >  55.00
|    |    |--- class: red
|--- ioreads >  50.00
|    |--- cpuseconds <= 75.00
|    |    |--- class: yellow
|    |--- cpuseconds >  75.00
|    |    |--- class: red
```

```
XfeaturesDF.shape, ylabelDF.shape (9, 3) (9, 1)
Xfeatures.shape, ylabel.shape     (9, 3) (9,)
['green' 'red' 'yellow'] [3 2 4]
```



This presentation was given by Craig Shallahamer (craig@orapub.com) at the October 27, 2020 NYOUG/Viscosity Conference. There is likely a more recent version at www.orapub.com > Login > Presentations

```
Siutation Function:

    if (features[ioreads] <= 50.000)
    {
        if (features[aas] <= 55.000)
        {
            situation is green 3.0
        }
        else
        {
            situation is red 1.0
        }
    }
    else
    {
        if (features[cpuseconds] <= 75.000)
        {
            situation is yellow 4.0
        }
        else
        {
            situation is red 1.0
        }
    }
```

1259 Snapshots, 161 Features
AWR Sysmetric Summary Dataset

```
In [2]: # Set Gobal Variables & Settings

        # Dataset Selection
        datasetSize = "small" #   3 features
        datasetSize = "real" # 161 features

        # Decision Tree Model Parameters
        maxTreeDepth  = 4      # 3 nice, but may not resolve
        treeRandomNo  = 123

        # Number of "important" features to use
        numImportantFeatures = 4 # 3 nice, but model will lack knowledge breath

        # Scoring Parameters (Repeated Stratified K-Folds)
        scoringKFolds=3
        scoringRepeats=3

        os.chdir("/Users/cshallah/Desktop/DS-working")  # <<<<<----------------- Change for your c
        print(os.getcwd())

        import warnings
        warnings.filterwarnings("ignore", category=FutureWarning)
        warnings.filterwarnings("ignore", category=UserWarning)

        testing = False # for development purposesonly

        print("Done.")

        /Users/cshallah/Desktop/DS-working
        Done.
```

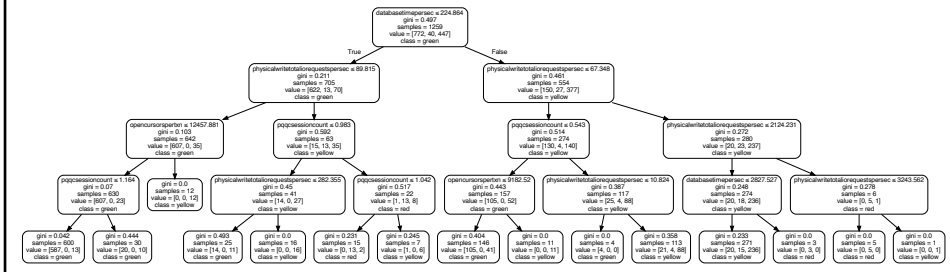
```
Model score BEFORE reducing features
  XfeaturesDF.shape (1259, 161)
  Test Sample Sets (folds*repeats) 9  Mean 0.850  Median 0.852

The most important 4  features:

1. feature 27 (0.507888)  databasetimepersec
2. feature 105 (0.193574)  physicalwritetotaliorequestspersec
3. feature 106 (0.059366)  pqqcsessioncount
4. feature 82 (0.050225)  opencursorspertxn

['databasetimepersec', 'physicalwritetotaliorequestspersec', 'pqqcsessioncount', 'opencu
rsorspertxn']

Done.
```

```
Model score AFTER reducing features
  XfeaturesDF.shape (1259, 4)
  Test Sample Sets (folds*repeats) 9  Mean 0.857  Median 0.855
(1259, 4)

Done.
```

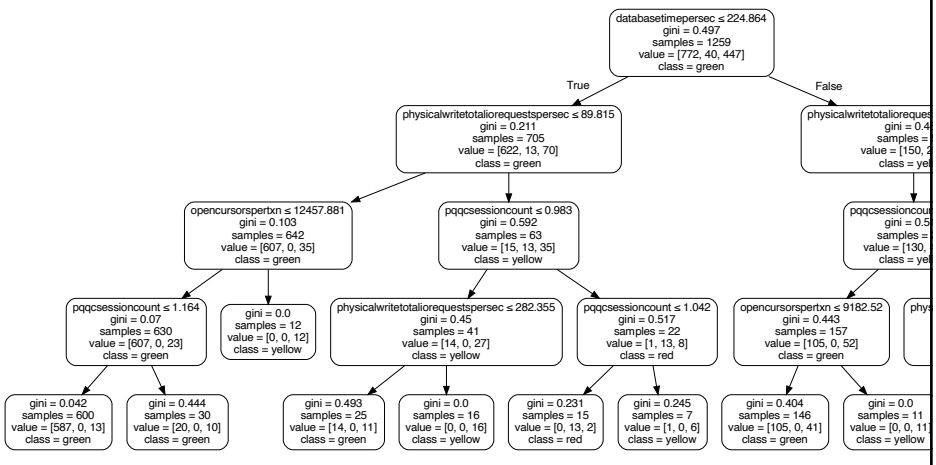
```
['databasetimepersec', 'physicalwritetotaliorequestspersec', 'pqqcsessioncount', 'opencu
rsorspertxn', 'snap_id']
['snap_id', 'indicator_level']

fullDF.shape, showXfeaturesDF.shape, ticketDF2.shape (1259, 6) (1259, 5) (1259, 2)

      databasetimepersec  physicalwritetotaliorequestspersec  \
0              25.729532                           10.688656
1              31.764333                           11.524863
2             192.718273                           29.671952
3             115.680235                           27.431951
4              61.694656                           18.233577
...                  ...                                 ...
1254          368.283606                           10.088674
1255           66.448166                          209.772049
1256          417.764680                          431.484811
1257          525.220110                           45.840548
1258           91.045015                           21.273919

      pqqcsessioncount  opencursorspertxn  snap_id indicator_level
0             1.000000        3403.642628    22572           green
1             0.100000        3344.923935    22573           green
2             0.131148        3563.701166    22574           green
3             0.220339        3746.751766    22575           green
4             0.150000        4459.560902    22576           green
...                ...                ...      ...             ...
1254          0.200000        6970.981624    23826          yellow
1255          0.000000         185.244398    23827           green
1256          0.000000      189269.050466    23828             red
1257          0.066667        1603.864693    23829          yellow
1258          0.000000        6673.475141    23830           green

[1259 rows x 6 columns]
```

```
(1259, 4) ['databasetimepersec', 'physicalwritetotaliorequestspersec', 'pqqcsessioncount
', 'opencursorspertxn']

|--- databasetimepersec <= 224.86
|   |--- physicalwritetotaliorequestspersec <= 89.81
|   |   |--- opencursorspertxn <= 12457.88
|   |   |   |--- pqqcsessioncount <= 1.16
|   |   |   |   |--- class: green
|   |   |   |--- pqqcsessioncount >  1.16
|   |   |   |   |--- class: green
|   |   |--- opencursorspertxn >  12457.88
|   |   |   |--- class: yellow
|   |--- physicalwritetotaliorequestspersec >  89.81
|   |   |--- pqqcsessioncount <= 0.98
|   |   |   |--- physicalwritetotaliorequestspersec <= 282.35
|   |   |   |   |--- class: green
|   |   |   |--- physicalwritetotaliorequestspersec >  282.35
|   |   |   |   |--- class: yellow
|   |   |--- pqqcsessioncount >  0.98
|   |   |   |--- pqqcsessioncount <= 1.04
|   |   |   |   |--- class: red
|   |   |   |--- pqqcsessioncount >  1.04
|   |   |   |   |--- class: yellow
|--- databasetimepersec >  224.86
|   |--- physicalwritetotaliorequestspersec <= 67.35
|   |   |--- pqqcsessioncount <= 0.54
|   |   |   |--- opencursorspertxn <= 9182.52
|   |   |   |   |--- class: green
|   |   |   |--- opencursorspertxn >  9182.52
|   |   |   |   |--- class: yellow
|   |   |--- pqqcsessioncount >  0.54
|   |   |   |--- physicalwritetotaliorequestspersec <= 10.82
|   |   |   |   |--- class: green
|   |   |   |--- physicalwritetotaliorequestspersec >  10.82
|   |   |   |   |--- class: yellow
|   |--- physicalwritetotaliorequestspersec >  67.35
|   |   |--- physicalwritetotaliorequestspersec <= 2124.23
|   |   |   |--- databasetimepersec <= 2827.53
|   |   |   |   |--- class: yellow
|   |   |   |--- databasetimepersec >  2827.53
|   |   |   |   |--- class: red
|   |   |--- physicalwritetotaliorequestspersec >  2124.23
```

```
XfeaturesDF.shape, ylabelDF.shape (1259, 4) (1259, 1)
Xfeatures.shape, ylabel.shape     (1259, 4) (1259,)
['green' 'red' 'yellow'] [772  40 447]
```



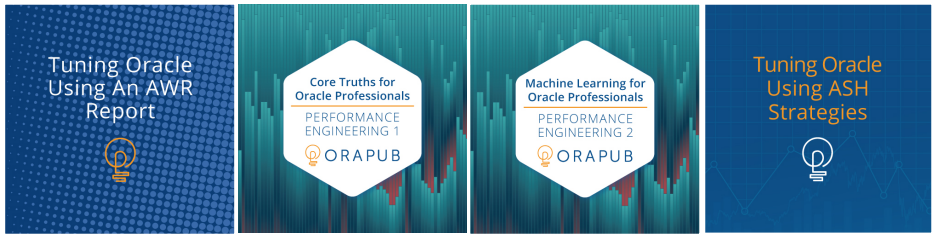

```
XfeaturesDF.shape, ylabelDF.shape (1259, 4) (1259, 1)
Xfeatures.shape, ylabel.shape     (1259, 4) (1259,)
['green' 'red' 'yellow'] [772  40 447]
```

Resource listing

- **OraPub Membership** **for premium content**
  - "How To" Webinars – two each month. Over 100 recorded!!
  - Video Seminars – any device, any time, high quality
  - 20% LVC discounts!
  - Learning paths, mentoring, assessments and certificates, priority response
  - SLACK forum exclusively for paid members
- **Live Virtual Classroom (LVC) Training**
  - **Machine Learning For Oracle Professionals**
  - Tuning Oracle Using An AWR Report
  - Tuning Oracle Using Active Session History (ASH) Strategies
  - **Core Truths For Oracle Professionals**
- **Toolkits** – Many tools available at orapub.com
- **Craig's Blog & Website** – Search: "uowtba", "queue"
- **Presentations** – www.orapub.com
- **Books:** Oracle Performance Firefighting. Forecasting Oracle Performance.

ORAPUB



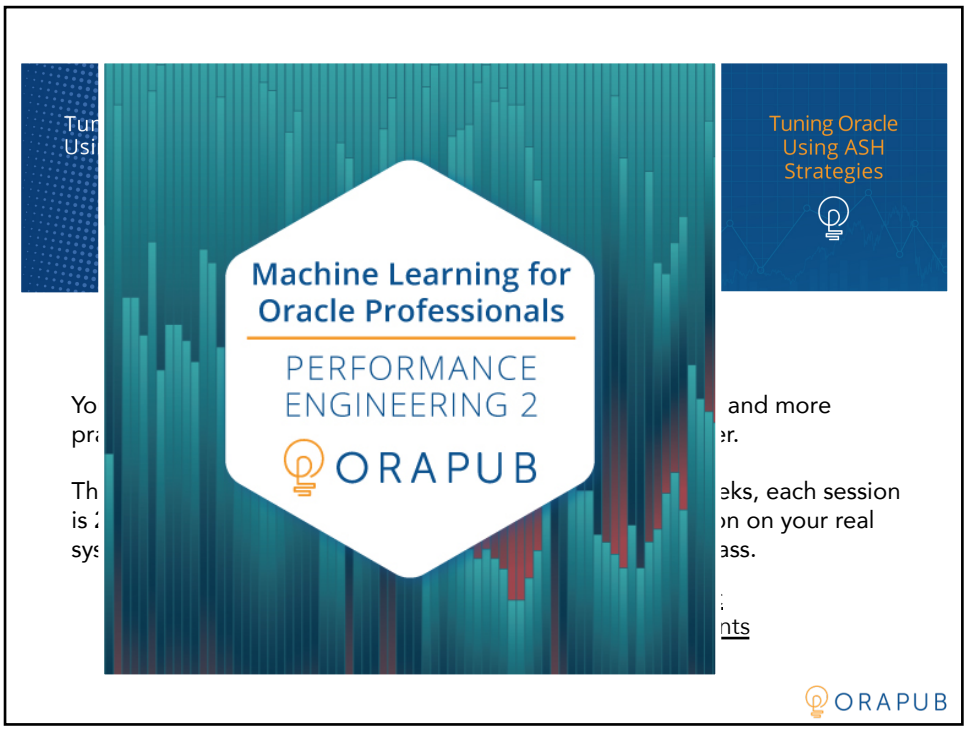
OraPub LVCs Are Specifically Designed For DBAs

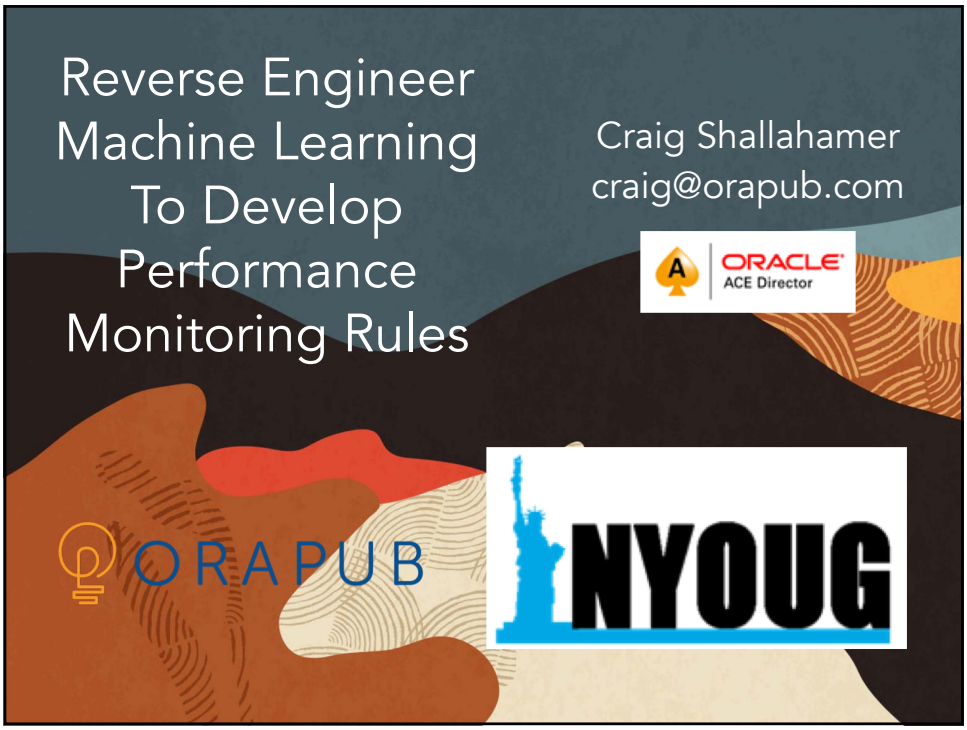
You learn to master the topic at a deeper, higher confidence and more practical level compared to any other teaching method I offer.

This occurs because the class is spread out over multiple weeks, each session is 2 hours with a day in between, you do homework/activation on your real systems and I personally work with you through the entire class.

Details & Registration: www.orapub.com/lvc
Event Calendar: https://www.orapub.com/events

ORAPUB


Machine Learning for Oracle Professionals
PERFORMANCE ENGINEERING 2
ORAPUB
Tuning Oracle Using ASH Strategies
and more
ORAPUB


Reverse Engineer Machine Learning To Develop Performance Monitoring Rules
Craig Shallahamer
craig@orapub.com
ORACLE ACE Director
ORAPUB
NYOUG