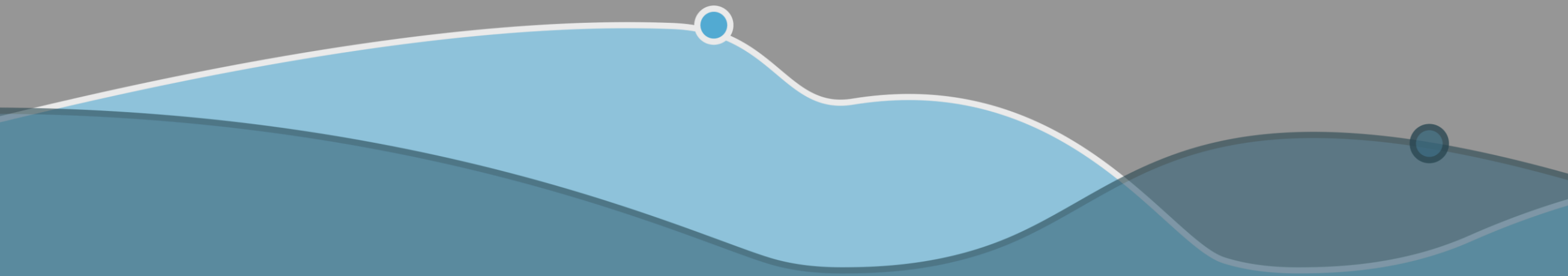




# Cortana Analytics Workshop

Sept 10 – 11, 2015 • MSCC



# Insights and Predictions: Integrating and Deploying Big Data Models through AzureML

Jeremy Reynolds  
Senior Data Scientist Lead

# Agenda

## Background

Revolution R Enterprise: Motivation

Revolution R Enterprise: Review

## Modeling with Revolution R Enterprise

Gaining Insights: Fitting and Tuning Models

Demonstration

## Deploying with AzureML

Scoring and Predictions: Locally

Scoring and Predictions as a Service: AzureML

Demonstration

## Conclusions

# Revolution R Enterprise: Motivation

Problem: R is not designed for Big Data

- Memory constraints

- Single-threaded

- Fundamental Design Decisions

Solution: Scalable Algorithms through Revolution R Enterprise (RRE)

- On-disk datasets work around memory ceiling of open source R

- Parallel External Memory Algorithms (PEMAs) allow for scalable performance across a variety of data platforms

- Designed for Big Data and Performance

# Revolution R Enterprise: Review

## Revolution R Enterprise Organization

Largely a set of additional functions that provide big data capability

These play nicely with open source R and additional packages available on the Comprehensive R Archive Network (CRAN)

# Revolution R Enterprise: Some Functions

`rxImport()`: Conversion to xdf format

`rxGetInfo()`: Extract meta-data about a dataset

`rxDataStep()`: Arbitrary transformations

`rxCrossTabs()`: Cross tabulation and mean computation

`rxSummary()`: Summary statistics

`rxLinMod()`: Ordinary Least Squares model estimation

# RRE Plays Nicely with Open Source R

```
inDS <- file.path(  
    rxGetOption("sampleDataDir"),  
    "DJIAdaily.xdf"  
)  
newDS <- rxDataStep(  
    inData = inDS,  
    transforms = list(  
        datestr = sprintf('%04d/%02d/%02d',  
                           Year, Month, DayOfMonth),  
        datevar = ymd(datestr)  
    ),  
    transformPackages = c("lubridate")  
)
```

# RRE Plays Nicely with Open Source R

```
inDS <- file.path(  
    rxGetOption("sampleDataDir"),  
    "DJIAdaily.xdf"  
)  
newDS <- rxDataStep(  
    inData = inDS,  
    transforms = list(  
        datestr = sprintf('%04d/%02d/%02d',  
                           Year, Month, DayOfMonth),  
        datevar = ymd(datestr)  
    ),  
    transformPackages = c("lubridate")  
)
```



# RRE Plays Nicely with Open Source R

A wide world of tools are at our disposal!

# Revolution R Enterprise: Platforms and Architectures

rxSetComputeContext()

RxLocalSeq

RxLocalPar

RxHadoopMR

RxInTeradata

RxForeachDoPar

# Revolution R Enterprise: Platforms and Architectures

We can leverage these tools on many platforms.

“Write once, evaluate anywhere.”

# Next Step: Gaining Insights

More to Data Science than Data Manipulation

Statistics, Machine Learning, and Algorithms

A Number of Scalable Algorithms are Already Implemented so your Data Science team can spend their time on data science rather than algorithm development

# Revolution R Enterprise: Algorithms

## Regression

Ordinary Least Squares and Generalized Linear Models: `[rxLinMod(); rxGlm()]`

Regression Decision Tree: `rxDTree()`

Regression Decision Forest: `rxDForest()`

Boosted Regression Trees: `rxBTrees()`

## Classification

Logistic Regression: `[rxLogit(); rxGlm()]`

Classification Decision Tree: `rxDTree()`

Classification Decision Forest: `rxDForest()`

Boosted Classification Trees: `rxBTrees()`

Naïve Bayes: `rxNaiveBayes()`

## Clustering

k-Means: `rxKmeans()`

# Revolution R Enterprise: Platforms and Architectures

We can leverage these tools on many platforms.

“Write once, evaluate anywhere.”

# Demo

Jeremy Reynolds



# Model Estimation Summary

A Variety of Tools and Algorithms are Available

On a Variety of Platforms

In the Cloud or On-premises



# Scoring a Model

We Have a Good Model...

Now what?

# Some Goals of Modeling

We gained some insight, and the insight holds the value.

We are happy with our model and we want to use it to generate new predictions

Retail Forecasting

Predictive Maintenance

Loan Application Scoring

# Generating Scores: Two Options

## Local Scoring

We have complete control and might be the only team with access  
e.g. Hold-out sample testing

## As a Service

We want to allow other members of the organization to score new observations  
e.g. A finance firm wants its loan officers to be able to leverage an internally estimated model predicting default status

# Local Scoring with Revolution R Enterprise

```
myDSwithPreds <- rxPredict(  
    modelObject = estModel,  
    data = holdOutData,  
    outData = newDS  
)
```

# Scoring as a Service

A very simple tool to facilitate this that builds on and leverages the Cortana Analytics stack.

Remember...

# RRE Plays Nicely with Open Source R

A wide world of tools are at our disposal!

# AzureML Package on CRAN

<http://cran.us.r-project.org/web/packages/AzureML/index.html>



# AzureML Package on CRAN

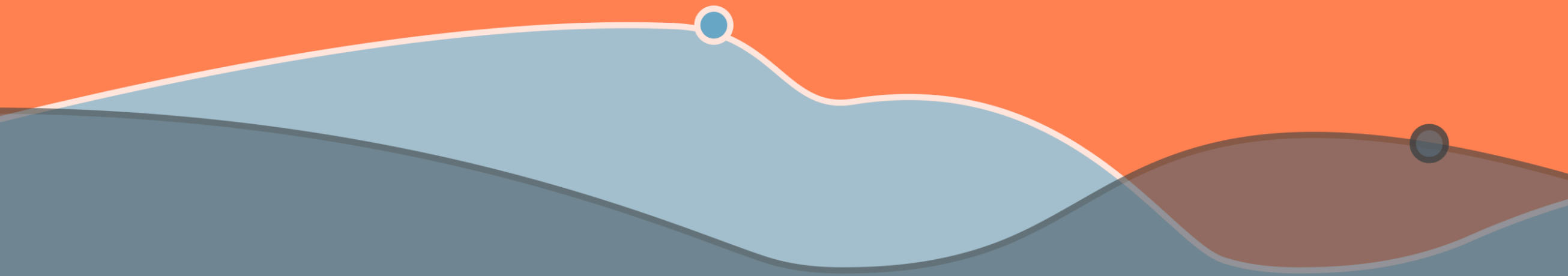
Provides a simple interface for easily discovering, publishing, and consuming web services

# Scoring as a Service

```
myScoringFun <- function(...){  
    ...  
}  
library(AzureML)  
serviceInfo <- publishWebService(  
    functionName = "myScoringFun"  
    serviceName = "myScoringService",  
    inputSchema = list(...),  
    outputSchema = list(...),  
    wkID = myWorkspaceID,  
    authToken = myAuthorizationToken  
)
```

# Demo

Jeremy Reynolds



# Deployment Summary

Local Scoring: rxPredict()

Scoring as a Service: AzureML

Does **NOT** depend on having access to full dataset

**Substantially decreases** deployment time to other teams and applications

Provides a clear path of value to adopting cloud-based computing for at least a subset of operations.

# Conclusions

You can analyze and gain insights from your Big Data either on-premises or in the cloud using Revolution R Enterprise.

Regardless of your data's location, you can leverage the [AzureML](#) package on CRAN in conjunction with RRE in order to dramatically simplify and quicken your deployment process.

Thank you.

