

An end-to-end open source machine learning platform

TensorFlow For JavaScript For Mobile & IoT For Production

The core open source library to help you develop and train ML models. Get started quickly by running Colab notebooks directly in your browser.

Get started with TensorFlow

Why TensorFlow?

```
1 # https://archive.ics.uci.edu/ml/datasets/Individual+household+electric+power+consumption
2
3 library(tidyverse)
4 library(rvest)
5 library(readr)
6 library(janitor)
7 library(lubridate)
8 library(vroom)
9
10
11
12
13 data_raw_tbl <- vroom::vroom("data/household_power_consumption.txt",
14   col_types = colx(
15     Date = col_character(),
16     Time = col_character(),
17     Global_active_power = col_double(),
18     Global_reactive_power = col_double(),
19     Voltage = col_double(),
20     Global_intensity = col_double(),
21     Sub_metering_1 = col_double(),
22     Sub_metering_2 = col_double(),
23     Sub_metering_3 = col_double()
24   ))
25
26 #> # Top Level : R Script
```

Console Terminal Jobs

```
~/Desktop/learning/lab3/tensorflow_energy.R
```

```
#> # Data frame: 2,075,259 x 8
#> # Columns: Date,Time,global_active_power,global_reactive_power,voltage,global_intensity,sub_metering_1,sub_metering_2,sub_metering_3
#> # # ... with 2,075,249 more rows, and 2 more variables: sub_metering_2 <dbl>, sub_metering_3 <dbl>
#>
```

TensorFlow for Multivariate Forecasting Energy Data [Time Series]

Matt Dancho & David Curry
Business Science Learning Lab





Learning Lab Structure

- **Presentation**
(20 min)
- **Demo's**
(30 min)
- **Pro-Tips**
(15 mins)



Matt Dancho

Founder of Business Science, Matt designs and executes educational courses and workshops that deliver immediate value to organizations. His passion is up-leveling future data scientists coming from untraditional backgrounds.



David Curry

Founder of Sure Optimize, David works with businesses to help improve website performance and SEO using data science. His passion is ethical Machine Learning initiatives.

Python & R Series



Learning Labs Pro
Community-Driven Data Science Courses

Matt Dancho \$19/m

- **Lab 33 - Shiny HR Attrition App**
Attrition Clustering w/ **Scikit Learn**



- **Lab 34 - Shiny Ecommerce App**
 - Machine Learning w/ **Scikit Learn**



- **Lab 35 - Shiny Finance App**
Univariate Forecasting w/ **TensorFlow**



- **Lab 36 - Energy Forecasting**
 - Multivariate Forecasting w/ **TensorFlow**



- **Lab 37 - Shiny Twitter App**
 - Text Classification



Learning Labs PRO

1.5-Hour Courses

Recordings, Code, Bonuses, & Slack

- \$39/month for monthly subscription
- \$29/month for annual subscription

Get Labs 2X per Month on Advanced Topics

university.business-science.io

LL PRO Course List

Coming Next! Python & R Series.



Python & R Series, 5-Part Series

- Lab 34 [Part 2]: [Advanced Customer Segmentation & Market Basket Analyzer App \(E-Commerce\)](#)
- Lab 33 [Part 1]: [Employee Segmentation with Python & R \(HR Analytics\)](#)

Shiny API, 5-Part Series

- Lab 32 [Part 5]: [Text Mining Tweets with Twitter & Tidytext](#)
- Lab 31 [Part 4]: [Forecasting Google Analytics with Facebook Prophet & Shiny](#)
- Lab 30 [Part 3]: [Shiny Financial Analysis with Tidyquant API \(Finance\)](#)
- Lab 29 [Part 2]: [Shiny Crude Oil Forecast \(Multivariate ARIMA\) with Quandl API & Fable](#)
- Lab 28 [Part 1]: [Shiny Real Estate App with Zillow API](#)

Marketing Analytics, 4-Part Series

- Lab 27 [Part 4]: [Google Trends Automation with Shiny](#)
- Lab 26 [Part 3]: [Machine Learning for Customer Journey](#)
- Lab 25 [Part 2]: [Marketing Multi-Channel Attribution with ChannelAttribution](#)
- Lab 24 [Part 1]: [A/B Testing for Website Optimization with Infer & Google Optimize](#)

SQL for Data Scientists, 3-Part Series

- Lab 23 [Part 3]: [Google Analytics & BigQuery \(SQL\) - Conversion Funnel Analysis](#)
- Lab 22 [Part 2]: [SQL for Time Series - Mortgage Loan Delinquency](#)
- Lab 21 [Part 1]: [SQL for Data Science - Home Loan Applications & Default](#)

Plus 20 More Labs:

Continuous Learning
Data Science

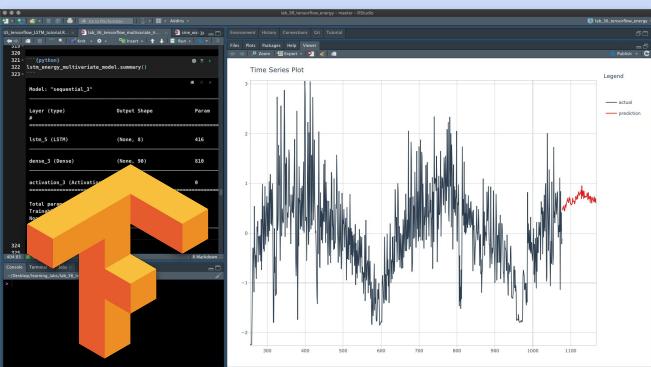


Learning Labs Pro
Community-Driven Data Science Courses

Matt Dancho



Agenda



- **Business Case**
 - Forecasting Applications
 - Shiny App from Lab 35
 - **TensorFlow**
 - Expands on Lab 35 (TF for Finance)
 - Univariate, Single Step
 - Univariate, Multi-Step
 - Multivariate, Single Step
 - Multivariate, Multi-Step
 - **30-Min Demo**
 - 600 Lines of Code
 - Enhanced TF Tutorial w/ Univariate, Multivariate, Single Step, & Multi-Step Explained
[LL PRO]
 - The image contains three logos: a large yellow and orange TensorFlow logo, a Python logo consisting of a yellow snake in a blue circle, and an R logo consisting of a blue letter 'R' with a grey circular arrow around it.
 - **Career Acceleration**
 - Shiny capabilities
 - Build your career



Bonuses

for LL PRO Members Today

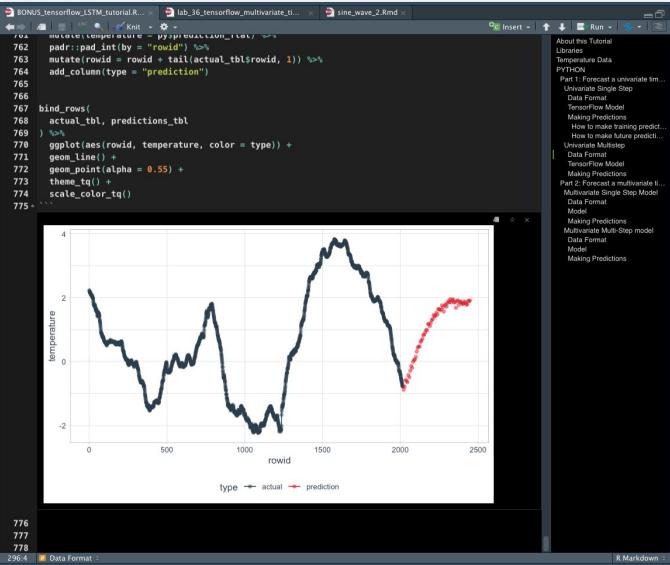
\$29/mo

When you purchase an annual plan



Bonus #1 - Enhanced TensorFlow Training

Uses TensorFlow LSTMs in the 4 Most Common Situations



Bonus #2 - Lab 35 LSTM Explorer

Integrate TensorFlow into your Shiny Apps!



Business Case Study

Forecasting

Demo - Become a LL PRO Member, & get this App!

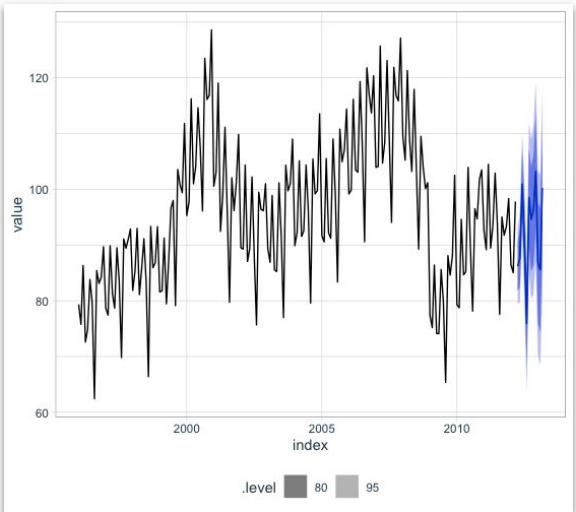


Gold Price
Prediction
Using Deep Learning

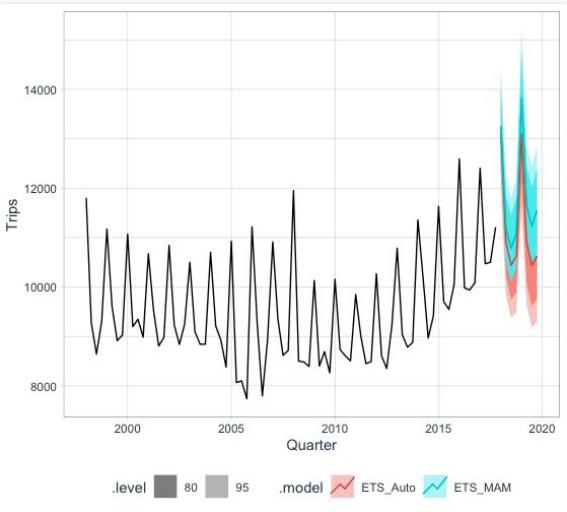
Classical Methods



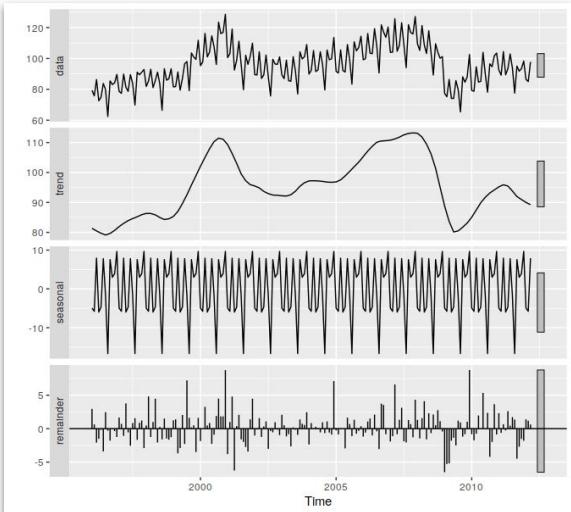
ARIMA



ETS



STL Decomposition

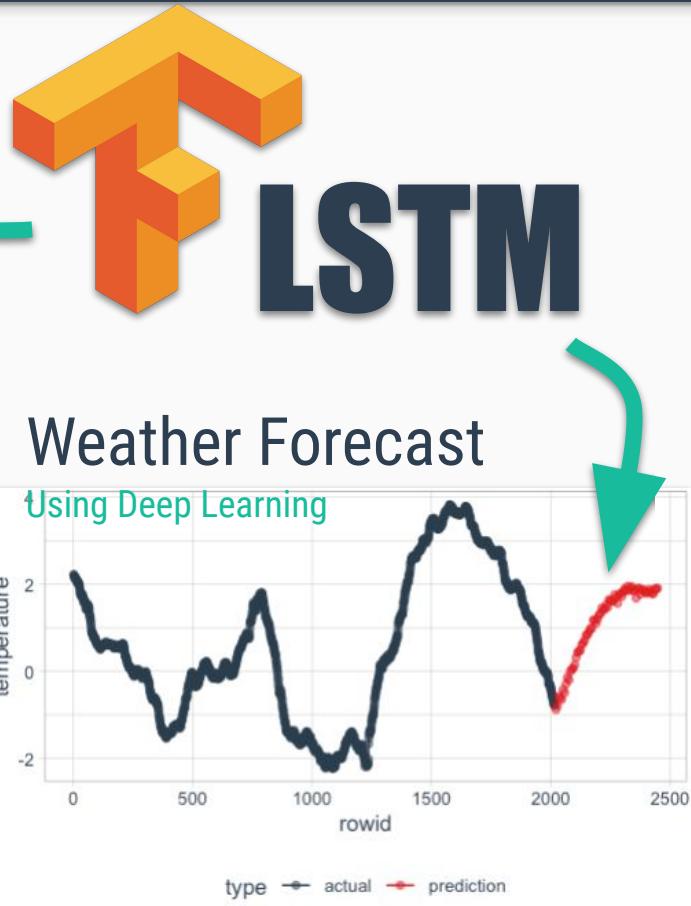
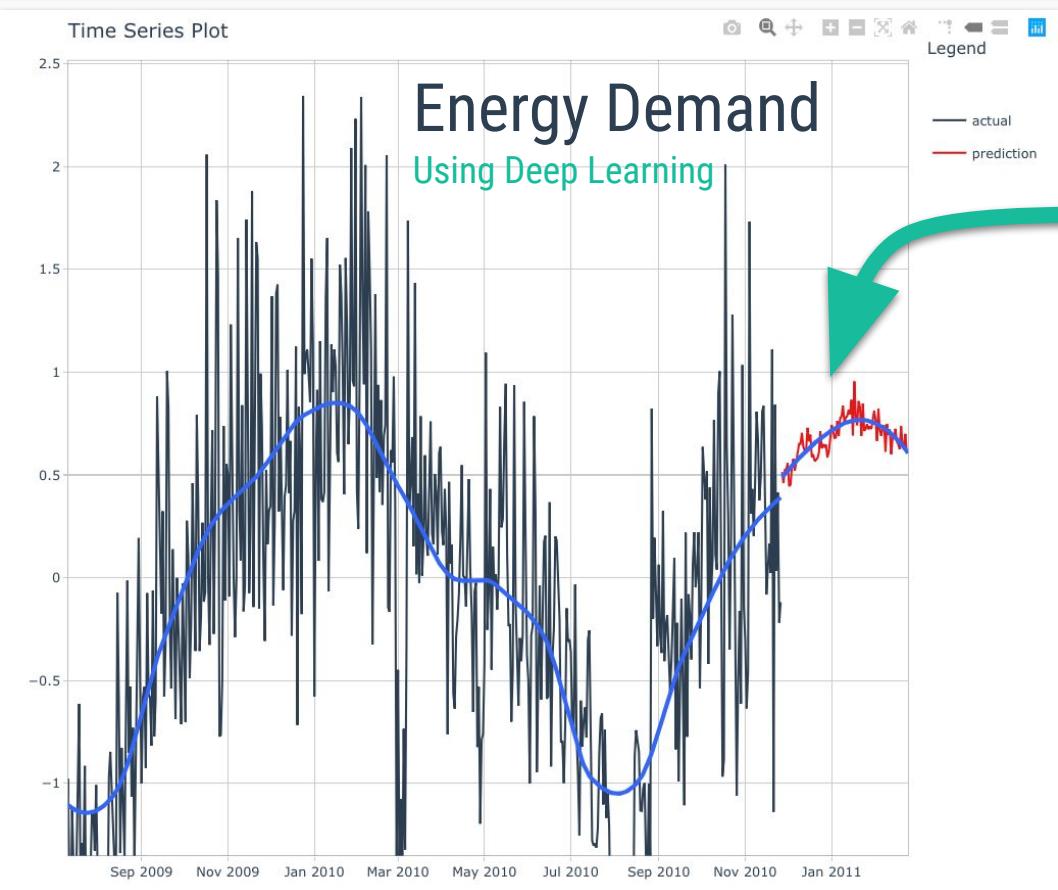


Classical Methods

Are simple.

Designed for problems with strong autocorrelation,
simple trends, & seasonalities.

Energy Prices & Weather Data





What is an LSTM?

An LSTM is a model that can be used to **model sequences**.

Comparison to ARIMA

ARIMA is a time series model that predicts the future using Autocorrelation with the past.

LSTM does the same thing with added benefits.

LSTM Benefits

- LSTM is powerful enough to learn without tweaking parameters (just let it run long enough to build relationships)
- Multivariate regressors can easily be incorporated into LSTM (ARIMA requires modeling errors separately)

Recurrent Neural Networks (RNNs)



Web Traffic Time Series Forecasting
Forecast future traffic to Wikipedia pages

Google 1,095 teams 3 years ago

Overview Data Notebooks Discussion Leaderboard Rules Team My Submissions Late Submission

Overview

Description
This competition focuses on the problem of forecasting the future values of multiple time series, as it has always been one of the most challenging problems in the field. More specifically, we aim the competition at testing state-of-the-art methods designed by the participants, on the problem of forecasting future web traffic for approximately 145,000 Wikipedia articles.

Evaluation
Sequential or temporal observations emerge in many key real-world problems, ranging from biological data, financial markets, weather forecasting, to audio and video processing. The field of time series encapsulates many different problems, ranging from analysis and inference to classification and forecast. What can you do to help predict future views?

Prizes
\$25,000 Prize Money

Timeline


This competition will run as two stages and involves prediction of actual future events. There will be a training stage during which the leaderboard is based on historical data, followed by a stage where participants are scored on real future events.

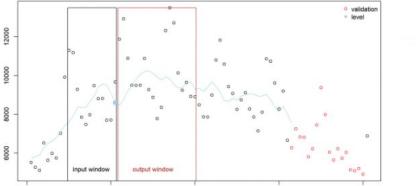
You have complete freedom in how to produce your forecasts: e.g. use of univariate vs multi-variate models, use of metadata (article identifier), hierarchical time series modeling (for different types of

User Data

M4 Forecasting Competition: Introducing a New Hybrid ES-RNN Model

Slawek Smyl, Jai Ranganathan, and Andrea Pasqua June 25, 2018

[Email](#) [Lock](#)



168 213 in Y 1 5 387 SHARES

By Slawek Smyl, Jai Ranganathan, Andrea Pasqua

Uber's business depends on accurate forecasting. For instance, we use forecasting to predict the expected supply of drivers and demands of riders in the 600+ cities we operate in, to identify

Popular Articles

 Meet Michelangelo: Uber's Machine Learning Platform September 5, 2017

 Uber's Big Data Platform: 100+ Petabytes with Minute Latency October 17, 2018

 Introducing Ludwig, a Code-Free Deep Learning Toolbox February 11, 2019

 Why Uber Engineering Switched from Postgres to MySQL

Competition Winners

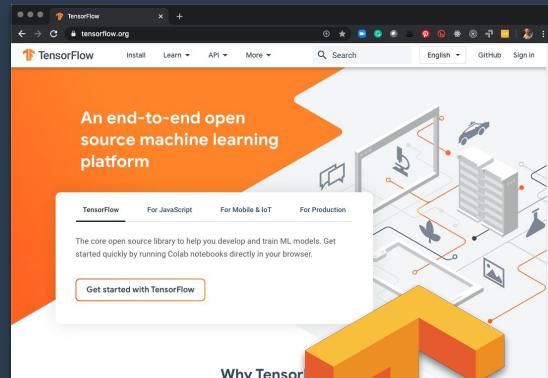
1st Place - Kaggle Web Traffic Time Series Forecasting
1st Place - M4 Forecasting Competition

Source:

1. <https://www.kaggle.com/c/web-traffic-time-series-forecasting/discussion/43795>
2. <https://eng.uber.com/m4-forecasting-competition/>

Workflow

From zero to
Automated LSTM Forecasting





Announcement: Timetk 2.0.0

Mission:

To make it easy to **visualize, wrangle and preprocess time series data** for forecasting and machine learning prediction.

Source:

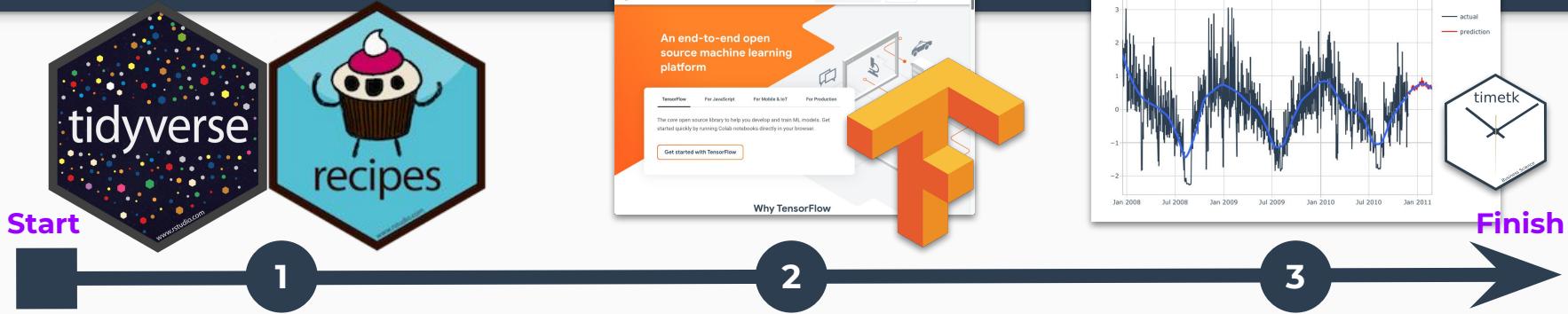
<https://business-science.github.io/timetk/>

**More announcements
coming shortly... Stay tuned!**

A screenshot of a web browser displaying the Timetk 2.0.0 website. The page features a large green banner at the top announcing "Timetk 2.0.0 is on CRAN!" with a small icon of three balloons. Below the banner, there's a navigation bar with tabs for "timetk 2.0.0", "Documentation", "Function Reference", and "News". A "Links" sidebar on the right provides links to CRAN, source code, bug reports, license information, and developer details. The main content area includes sections for "Mission", "Documentation", and "Package Functionality", along with a comparison table for different R packages. The Timetk logo, which is a hexagon with a stylized 'X' and the word "timetk" inside, is prominently displayed.

Task	timetk	tsibble	feasts	tibbletime
Structure				
Data Structure	tibble (tbl)	tsibble (tbl_ts)	tsibble (tbl_ts)	tibbletime (tbl_time)
Visualization				
Interactive Plots (plotly)	✓	✗	✗	✗
Static Plots (ggplot)	✓	✗	✓	✗

Workflow Step-By-Step



Formatting the data for analysis

TensorFlow
LSTM
Deep Learning

Interactive Plot w/ timetk

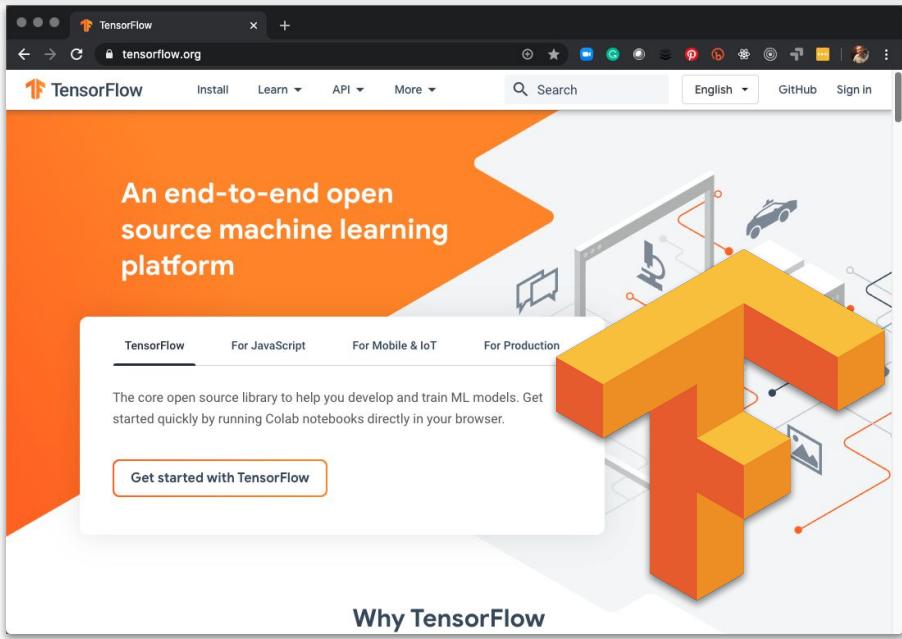


Goals today

Minimal Theory, Maximal Results

4 Types of Problems:

1. Univariate vs Multivariate
 2. Single Step vs Multi-Step
- 

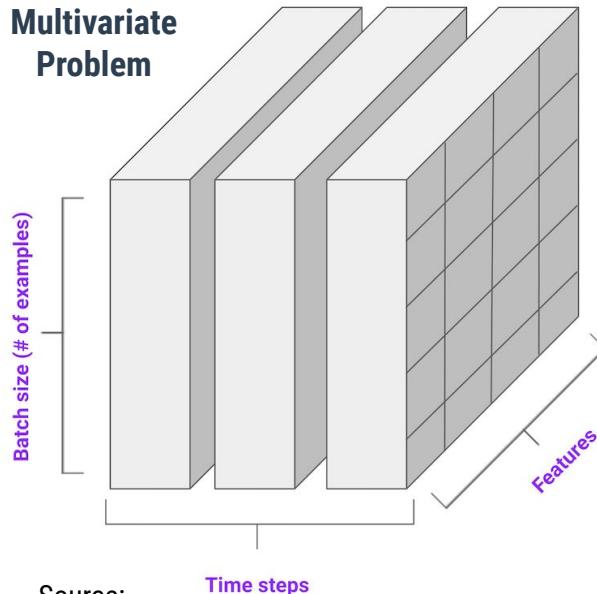


Source: <https://www.tensorflow.org/>

LSTM Implementation - Formatting Predictors



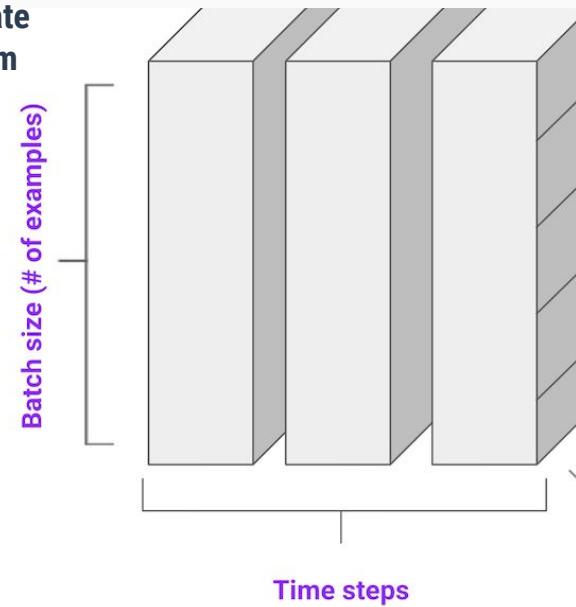
Multivariate
Problem



Source:

https://www.tensorflow.org/tutorials/structured_data/time_series#top_of_page

Univariate
Problem



Step 1 - Prepare the Data
Shaping the problem as a 3D Tensor



LSTM Implementation

```
299 ## LSTM Model  
300  
301 ````{python}  
302 model = Sequential()  
303  
304 model.add(Input(shape=(12,1))) ← Lookback  
305 model.add(LSTM(64, return_sequences=True, activation='tanh'))  
306 model.add(LSTM(12, return_sequences=False, activation='relu'))  
307  
308 model.add(Dense(1)) ← Single Step vs Multi-Step  
309  
310 model.add(Activation(activation='linear'))  
311  
312 model.compile(optimizer="rmsprop", loss="mae")  
313  
314 ````
```

```
316 ```` {python}  
317 model.summary()  
318 ````  
  
Model: "sequential_13"  
  
Layer (type)          Output Shape         Param #  
=====  
lstm_26 (LSTM)        (None, 12, 64)      16896  
  
lstm_27 (LSTM)        (None, 12)           3696  
  
dense_13 (Dense)      (None, 1)            13  
  
activation_13 (Activation) (None, 1)          0  
  
=====  
Total params: 20,605  
Trainable params: 20,605  
Non-trainable params: 0
```

Step 2 - Design a Model

Set of instructions for TensorFlow's Model API



LSTM Implementation

```
```{python}
TAKES SEVERAL MINUTES TO RUN
EVALUATION_INTERVAL = 200
EPOCHS = 5

lstm_energy_multivariate_model.fit(
 train_tf,
 epochs = EPOCHS,
 steps_per_epoch = EVALUATION_INTERVAL,
 validation_data = val_tf, ←
 validation_steps = 5)
```

```
Train for 200 steps, validate for 5 steps
Epoch 1/5
200/200 [=====] - 13s 63ms/step - loss: 0.5699 - val_loss: 0.3733
Epoch 2/5
200/200 [=====] - 11s 57ms/step - loss: 0.5137 - val_loss: 0.3689
Epoch 3/5
200/200 [=====] - 11s 56ms/step - loss: 0.4923 - val_loss: 0.3752
Epoch 4/5
200/200 [=====] - 11s 56ms/step - loss: 0.4742 - val_loss: 0.3851
Epoch 5/5
200/200 [=====] - 12s 58ms/step - loss: 0.4605 - val_loss: 0.3992
<tensorflow.python.keras.callbacks.History object at 0x7fa9a24aba58>
```

## Critical Point:

We're implementing Train/Test Validation this Week

Validation helps calibrate our model to avoid overfitting

## Step 3 - Fit the Model

Takes a few minutes - Optimizes using Gradient Descent



# LSTM Implementation

```
367
368 ````{python}
369 prediction_2d = lstm_energy_multivariate_model.predict(last_data_reshaped)
370 prediction_2d
371 ````

array([[0.5142785 , 0.461255 , 0.5258459 , 0.527794 , 0.5572278 ,
 0.44538248, 0.45278507, 0.543731 , 0.57870924, 0.5208244 ,
 0.5630455 , 0.6252231 , 0.6506658 , 0.65035355, 0.7049812 ,
 0.63702327, 0.61686575, 0.6228745 , 0.7293047 , 0.6636963 ,
 0.6955936 , 0.58696467, 0.5997318 , 0.5677127 , 0.5760865 ,
 0.5807149 , 0.6062061 , 0.7117574 , 0.63016367, 0.64583313,
 0.6376316 , 0.647565 , 0.58625096, 0.61131716, 0.67750454,
 0.7434214 , 0.71600103, 0.6803861 , 0.6819909 , 0.7630124 ,
 0.6893167 , 0.7763725 , 0.7856558 , 0.8358086 , 0.75616163,
 0.77926874, 0.7864046 , 0.81725484, 0.7946158 , 0.86499655,
 0.69225734, 0.9549104 , 0.7815149 , 0.7420964 , 0.85878646,
 0.8303387 , 0.6909549 , 0.84770656, 0.71461713, 0.724964 ,
 0.7475225 , 0.722575 , 0.73581505, 0.7233134 , 0.7677208 ,
 0.81092256, 0.7684323 , 0.66609794, 0.8206383 , 0.7157036 ,
 0.69055116, 0.7317369 , 0.7427145 , 0.6130868 , 0.7484657 ,
 0.6953063 , 0.72071064, 0.6008488 , 0.6830556 , 0.6820278 ,
 0.72441137, 0.6623219 , 0.6506088 , 0.6257362 , 0.73729795,
 0.6495549 , 0.6257185 , 0.69964373, 0.6059621 , 0.6157027]],
 dtype=float32)

372
373 ````{python}
374 prediction_2d.shape
375 ````

(1, 90)
```

## 3D Arrays

LSTM's work with 3D sequences  
(cubes)

## 2D Arrays

LSTM's return predictions as 2D arrays

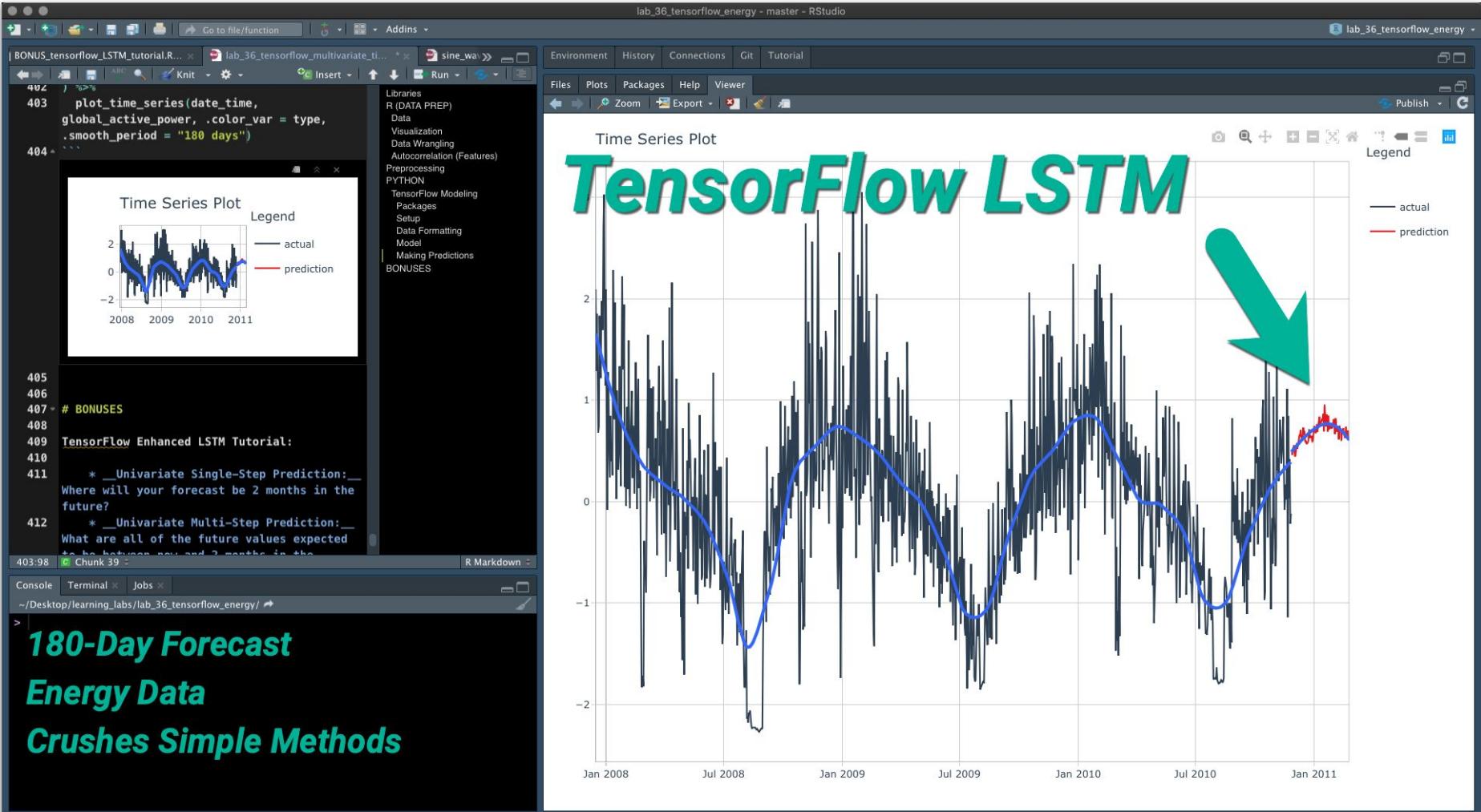
## Step 4 - Predict

Reshaping the data becomes very critical w/ multivariate

# Automate with Shiny



# 30-Min Demo



# Career Acceleration

And how Shiny Apps help

# Automate with Shiny



# Career Tip #1: Give Businesses Apps



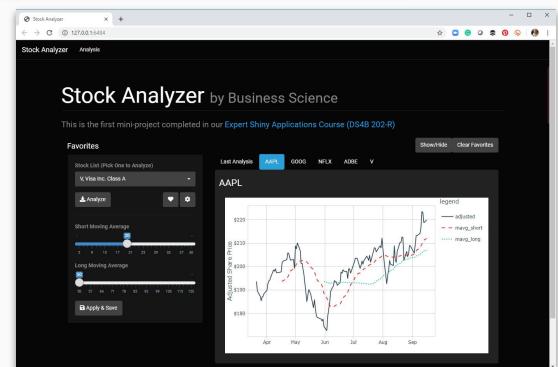
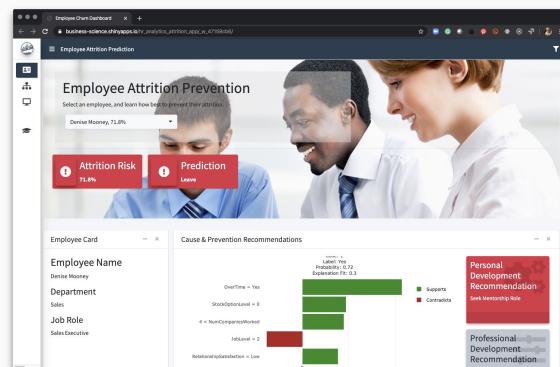
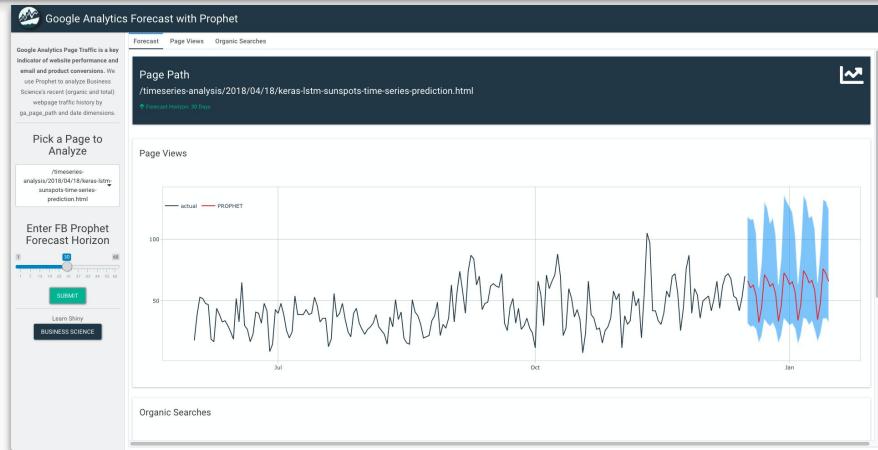
Apps are what businesses need

Businesses can't scale reports

Businesses have cloud & servers  
(AWS / Azure)

Businesses need your analysis in the right form

You can solve these problems with shiny apps



Apps solve these challenges

# Career Tip #2: Say no to reports



# No

arXiv:1811.08963v1 [cs.LG] 21 Nov 2018

## Multivariate Forecasting of Crude Oil Spot Prices using Neural Networks

Ganapathy S. Natarajan

School of Mechanical, Industrial, and Manufacturing Engineering  
Oregon State University  
Corvallis, OR, USA  
Email: gana.natarajan@oregonstate.edu

Aishwarya Ashok

Department of Computer Science and Engineering  
University of Texas at Arlington  
Arlington, TX, USA  
Email: aishwarya.ashok@mavs.uta.edu

**Abstract**—Crude oil is a major component in most advanced economies of the world. Accurately predicting and understanding the behavior of crude oil prices is important for oil economists, analysts, forecasters, and traders, to name a few. The price of crude oil has declined in the past decade and is seeing a phase of stability; but will this stability last? This work is an empirical study on how multivariate analysis may be employed to predict crude oil spot prices using neural networks. The concept of using neural networks showed promising potential. A very few neural network models have been able to perform on par with ARIMA models - the state-of-the-art model in time-series forecasting. Advanced neural network models using larger datasets may be used in the future to extend this proof-of-concept to a full scale framework.

**Keywords:**crude oil; multivariate forecasting; neural networks; ARIMA; regression

### I. INTRODUCTION

Crude oil spot prices saw a tremendous up-tick in the first decade of the 21<sup>st</sup> century. Since 2014, crude oil prices have fallen and may have stabilized now. However, there has always been a constant interest in accurately predicting crude oil prices; given that crude oil drives a major portion of the economy. Economists, scientists, data analysts, and traders are all interested in models that give them the best accuracy. In the last decade, advances in machine learning has enabled everyday data analysts to use techniques like neural networks; case-in-point is Google's TensorFlow™.

Crude oil prices have mostly been forecasted using time-series methods. There is some work in dealing with crude oil price forecasting as an econometric problem; however, there is very limited work in using multivariate techniques that move away from traditional regression modeling. This paper combines the lack of multivariate forecasting and advances in machine learning to provide a proof-of-concept for using Neural Networks in multivariate forecasting of crude oil prices. The results presented in this study are mostly empirical and lays the foundation for more in-depth studies in this direction.

### II. RELATED WORK

We submitted a literature review paper to a conference [1]. This work looked at the different types of models and methods used in forecasting oil prices in general. Time series

models were predominantly used. Time series models use just the oil price over time to predict future prices. ARIMA group of methods were the most commonly used time series forecasting methods [2]–[5] either for model building or as a benchmark or for a part of the model building. Artificial intelligence and machine learning based models [6]–[10] including neural network models [11] were also used to just predict the time series or components of the time series.

The second type of approach to forecasting oil prices was econometric models. Econometric models usually perform linear regression based analysis and take into account economic factors. Some of the common independent variables used were income growth, price increase, and population growth [12], gas tax exemption [13], OECD stocks, and OPEC spare capacity [14]. OECD is the Organization for Economic Co-operation and Development - a group of developed economies in the world, and OPEC is the Organization of the Petroleum Exporting Countries. Some of the variables used in econometric studies historically, will be used in this study too, viz. CPI, population, petroleum stocks, to name a few.

The third commonly used methods are multivariate models, which are of interest in this paper. Multivariate models use multiple variables akin to econometric models, but do not always assume a linear relationship or use just regression. Some of the commonly used multivariate methods include multivariate time series models [15], neural networks [16], [17], and multi-criteria decision models [18]. Reference [16] used neural networks to predict commercial oil price using country of origin, sulfur content, and density, and [17] used wavelet neural network to predict monthly crude oil price using just the inventory levels - not truly a multivariate methodology.

Recent works have continued with the trend of using different methods but still mostly perform time-series forecasting. Although the recent works use hybrid models [19], ensemble neural network models [20], gray wave model [21], and wavelet neural network models [22], they still are time-series forecasting models. Reference [23] used network science and predicted parts of a time series to deal with specific non-linear patterns.

The common pattern in all the related work is that most of

# Career Tip #3: Say YES to apps.



# Yes

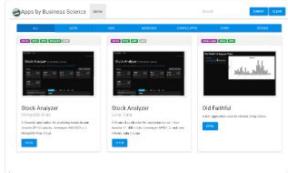
Apps by Business Science +

apps.business-science.io

Apps by Business Science App Gallery Search Apps Submit Clear

ALL BUSINESS FINANCE HUMAN RESOURCES MARKETING SALES

**Business** Shiny AWS

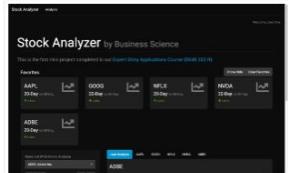


**Application Library**  
with Full-Text Search

A meta-app for hosting multiple business applications for an organization. Includes Full-Text Search and advanced filtering capability. Hosted on AWS.

Launch App Build It in DS4B 202A-R

Finance Shiny AWS MongoDB Auth



**Stock Analyzer** by Business Science

This is our first project completed in our App4Data Applications Course (DS4B 202A)

Features:

- API: SP 500 stocks
- EC2: 23 Feb - 2021
- DB: 00000
- ML: 23 Feb
- H2O: 23 Feb

R: 23 Feb

**R Packages:** tidyquant, plotly, mongolite

Launch App Build It in DS4B 202A-R

Human Resources Shiny H2O Bootstrap 4



**HR Employee Attrition Prevention**  
Bootstrap 4

Apply H2O machine learning to predict and explain employee churn. Identify which departments, job roles, job levels, and people have the highest expected attrition risk.

**R Packages:** H2O, LIME

Launch App Coming Soon

[apps.business-science.io](https://apps.business-science.io)

# 4-Course R-Track System



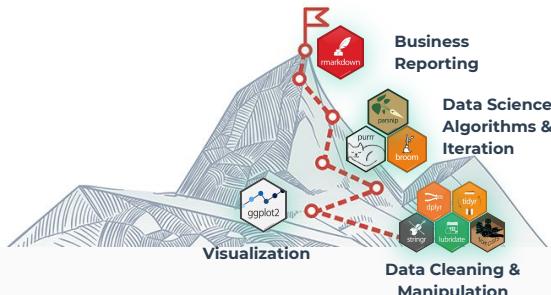
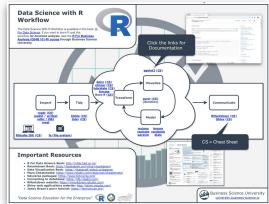
## Business Analysis with R (DS4B 101-R)

## Data Science For Business with R (DS4B 201-R)

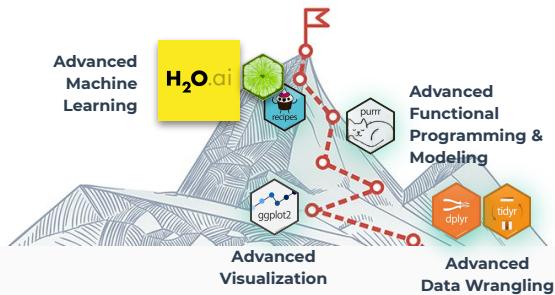
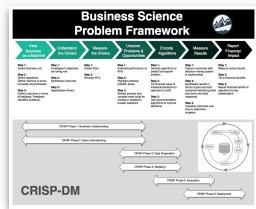
## Web Apps & Shiny Developer (DS4B 102-R + DS4B 202A-R)

### Project-Based Courses with Business Application

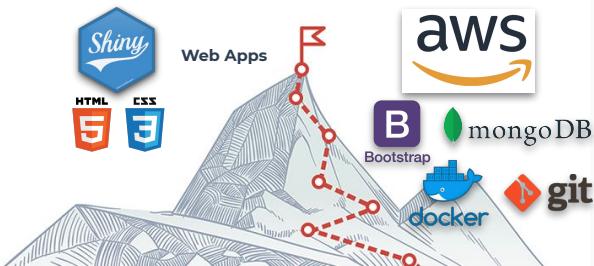
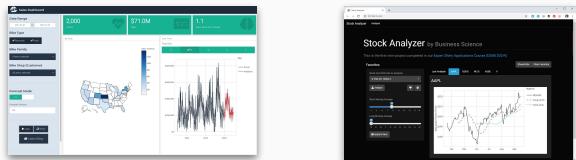
Data Science Foundations  
**7 Weeks**



Machine Learning & Business Consulting  
**10 Weeks**



Web Application Development  
**12 Weeks**

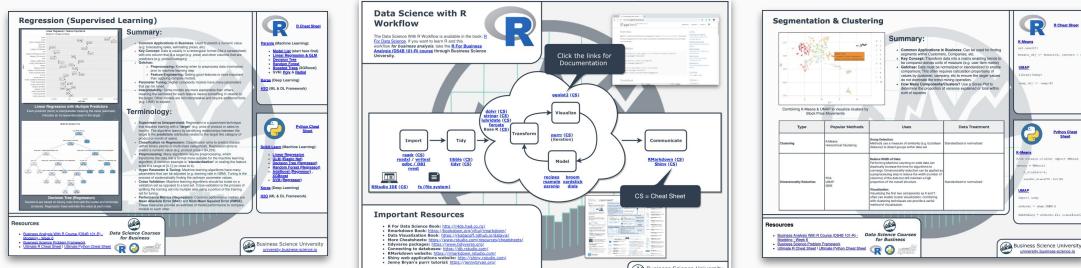


# Key Benefits

- Fundamentals - Weeks 1-5 (25 hours of Video Lessons)
  - Data Manipulation (dplyr)
  - Time series (lubridate)
  - Text (stringr)
  - Categorical (forcats)
  - Visualization (ggplot2)
  - Programming & Iteration (purrr)
  - 3 Challenges
- **Machine Learning - Week 6 (8 hours of Video Lessons)**
  - Clustering (3 hours)
  - Regression (5 hours)
  - 2 Challenges
- Learn Business Reporting - Week 7
  - RMarkdown & plotly
  - 2 Project Reports:
    1. Product Pricing Algo
    2. Customer Segmentation

# Business Analysis with R (DS4B 101-R)

Data Science Foundations  
**7 Weeks**



# Key Benefits

## End-to-End Churn Project

Understanding the Problem & Preparing Data - Weeks 1-4

- Project Setup & Framework
- Business Understanding / Sizing Problem
- Tidy Evaluation - rlang
- EDA - Exploring Data -GGally, skimr
- Data Preparation - recipes
- Correlation Analysis
- 3 Challenges

## Machine Learning - Weeks 5, 6, 7

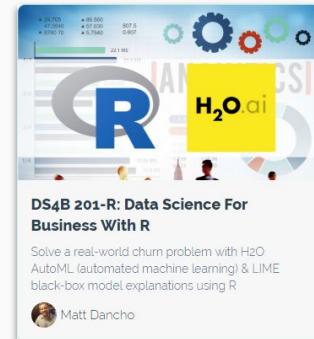
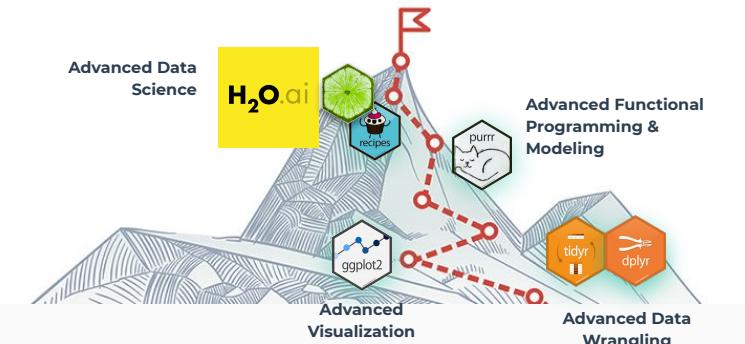
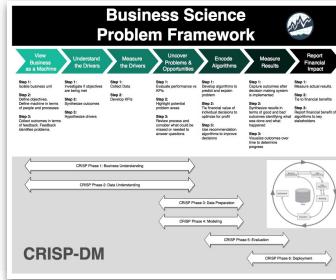
- H2O AutoML - Modeling Churn
- ML Performance
- LIME Feature Explanation

## Return-On-Investment - Weeks 7, 8, 9

- Expected Value Framework
- Threshold Optimization
- Sensitivity Analysis
- Recommendation Algorithm

# Data Science For Business (DS4B 201-R)

Machine Learning & Business Consulting  
**10 Weeks**



# Key Benefits

## Learn Shiny & Flexdashboard

- Build Applications
- Learn Reactive Programming
- Integrate Machine Learning

## App #1: Predictive Pricing App

- Model Product Portfolio
- XGBoost Pricing Prediction
- Generate new products instantly

## App #2: Sales Dashboard with Demand Forecasting

- Model Demand History
- Segment Forecasts by Product & Customer
- XGBoost Time Series Forecast
- Generate new forecasts instantly

# Shiny Apps for Business (DS4B 102-R)



Web Application Development  
**4 Weeks**



**DS4B 102-R: Shiny Web Applications  
For Business (Level 1)**

Build a predictive web application using Shiny, Flexdashboard, and XGBoost.

Matt Dancho

# Key Benefits

Frontend + Backend + Production Deployment

## Frontend for Shiny

- Bootstrap

## Backend for Shiny

- MongoDB Atlas Cloud
- Dynamic UI
- User Authentication & Security

## Production Deployment

- AWS
- EC2 Server
- SSL & HTTPS Encryption

# Shiny Apps for Business (DS4B 202A-R)



Web Application Development  
**6 Weeks**



# 15% OFF PROMO Code: **learninglabs**

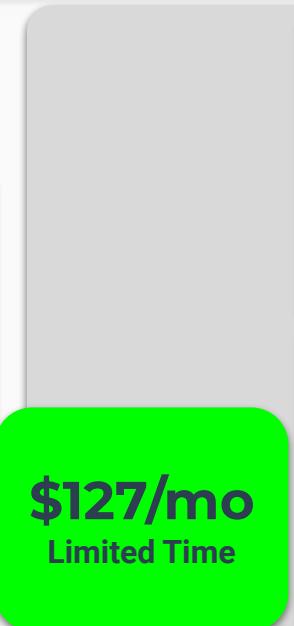


## R-TRACK BUNDLE

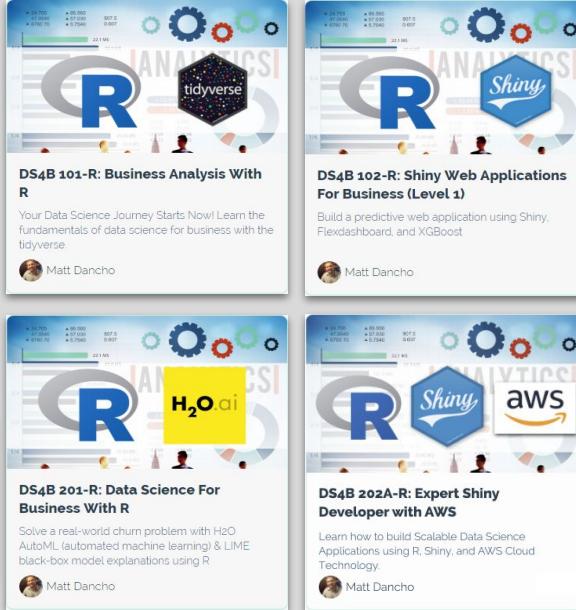
**4-Course Bundle - Machine Learning + Expert Web Applications (R-Track)**

Go from Beginner to Expert Data Scientist & Shiny Developer in Under 6-Months

4 Course Bundle ~~\$1,500~~



<input type="radio"/>	<b>Paid Course</b> 15% COUPON DISCOUNT	\$1,596 \$2,356.60
<input checked="" type="radio"/>	<b>12 Low Monthly Payments</b> 15% COUPON DISCOUNT 12X Payment Plan	12 payments of \$149/m 12 payments of \$126.65/m



**DS4B 101-R: Business Analysis With R**  
Your Data Science Journey Starts Now! Learn the fundamentals of data science for business with the tidyverse.

**DS4B 102-R: Shiny Web Applications For Business (Level 1)**  
Build a predictive web application using Shiny, Flexdashboard, and XGBoost.

**DS4B 201-R: Data Science For Business With R**  
Solve a real-world churn problem with H2O AutoML (automated machine learning) & LIME black-box model explanations using R.

**DS4B 202A-R: Expert Shiny Developer with AWS**  
Learn how to build Scalable Data Science Applications using R, Shiny, and AWS Cloud Technology.

# Career acceleration awaits

[university.business-science.io](http://university.business-science.io)

