```
5
6  # LIBRARIES ----
7  library(tidyverse)
8  library(tidyquant)
9
10 library(DataExplorer)
11 library(correlationfunnel)
12
13 library(recipes)
14
15 library(tidygraph)
16 library(ggraph)
```

Network Analysis

For Customer Community Detection

Difficulty: **Intermediate**

```
28
29 # 2.1 Minimum Payments has NA (missing d
```

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**.

Success Story

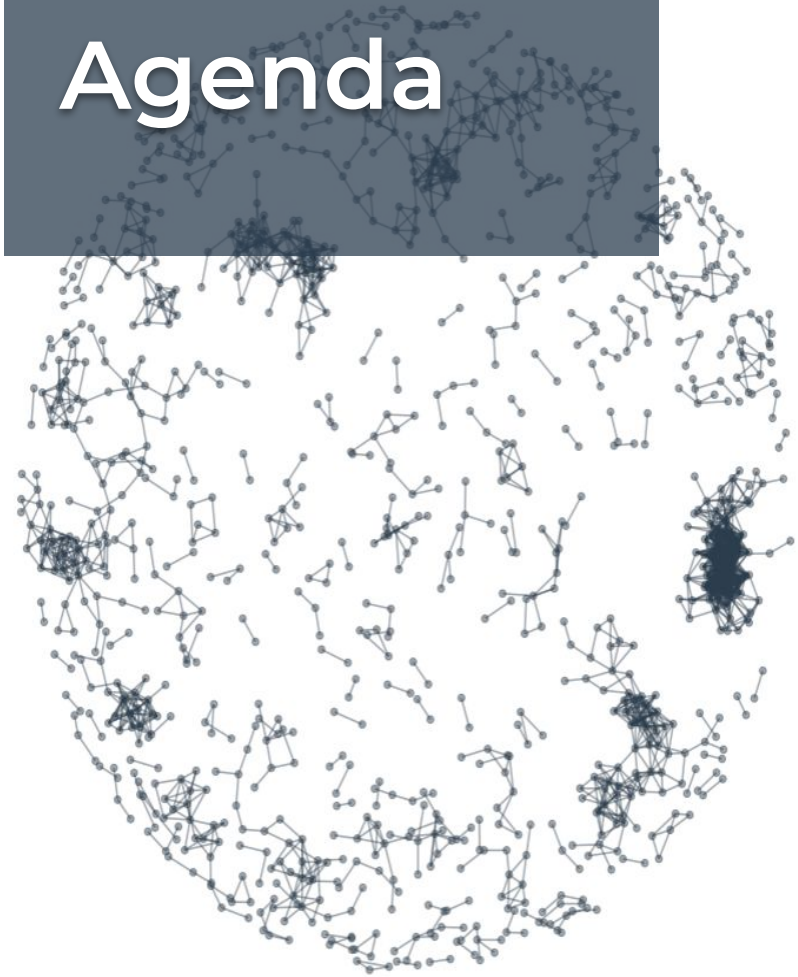
Josh Nelson

- **Friday** - Began Jumpstart
- **Sunday** - Began 101
- **Monday** - **Analyzing Google Analytics API & Plotting**

"I've been [messing] around with Python for 4 months. I learned more in the 2 days I did Jumpstart with R."

#Business Science Success

Agenda



- **Business Case Study**
  - Customer History
- **Network Analysis**
  - 2 Types
  - Key Concepts
- **R Packages**
  - tidygraph
  - ggraph
- **30-Min Demo**
  - Bank Customers
  - Account History
  - **EDA**
  - **Network Analysis**
  - **Machine Learning**
- **Pro-Tips:**
  - Tactics to **Explain** Why Customers belong to Communities

Learning Labs PRO

Every 2-Weeks

1-Hour Course

Recordings + Code + Slack

$19/month

university.business-science.io

Lab 18
Time Series Anomaly Detection with anomalize [HOT - 300+ Data Scientists Attended Live!]

Lab 17
Anomaly Detection with H2O Machine Learning

Lab 16
R's Optimization Toolchain, Part 2 - Nonlinear Programming

Lab 15
R's Optimization Toolchain, Part 1 - Linear Programming

Lab 14
Customer Churn Survival Analysis

Continuous Learning
Jet Fuel for your Brain

Customer Communities

Business Case

Customers Naturally Form Communities

Can Be Profitable to Detect Communities

Customer Communities are Natural Phenomenon

Detection is really important:

1. Avoid Trap of Focusing on EVERYONE
2. Customize Products & Services for Key Groups

Customer History

Descriptive Features

Customers

```
> credit_card_tbl
# A tibble: 8,950 x 18
   CUST_ID BALANCE BALANCE_FREQUEN… PURCHASES ONEOFF_PURCHASES INSTALLMENTS_PU… CASH_ADVANCE PURCHASES_FREQU… ONEOFF_PURCHASE…
   <chr>     <dbl>            <dbl>     <dbl>            <dbl>            <dbl>        <dbl>           <dbl>           <dbl>
 1 C10001     40.9            0.818      95.4                0             95.4            0           0.167               0
 2 C10002   3202.             0.909       0                  0              0           6443.          0                   0
 3 C10003   2495.             1         773.               773.             0              0           1                   1
 4 C10004   1667.             0.636    1499               1499              0            206.          0.0833              0.0833
 5 C10005    818.             1          16                 16              0              0           0.0833              0.0833
 6 C10006   1810.             1        1333.                0           1333.             0           0.667               0
 7 C10007    627.             1        7091.             6403.            688.             0           1                   1
 8 C10008   1824.             1         436.                0            436.             0           1                   0
 9 C10009   1015.             1         861.              661.            200              0           0.333               0.0833
10 C10010    152.             0.545    1282.             1282.              0              0           0.167               0.167
# … with 8,940 more rows, and 9 more variables: PURCHASES_INSTALLMENTS_FREQUENCY <dbl>, CASH_ADVANCE_FREQUENCY <dbl>,
#   CASH_ADVANCE_TRX <dbl>, PURCHASES_TRX <dbl>, CREDIT_LIMIT <dbl>, PAYMENTS <dbl>, MINIMUM_PAYMENTS <dbl>, PRC_FULL_PAYMENT <dbl>,
#   TENURE <dbl>
```

Network Analysis Basics

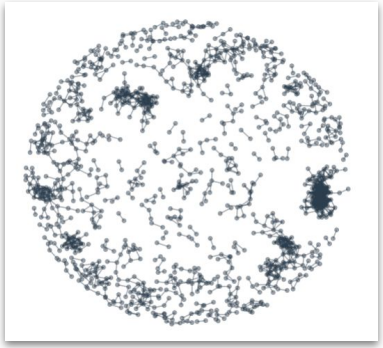
80/20 Concepts

Types of Network Analysis

1

Undirected

Strength of Relationship

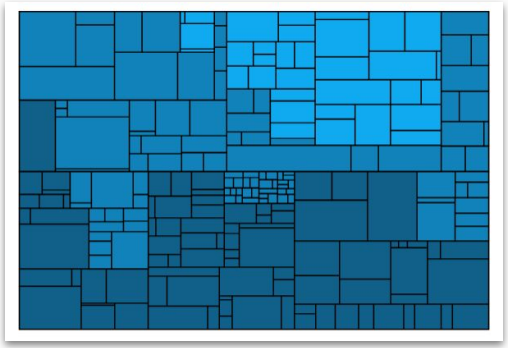


Clustering

2

Directed

Hierarchical Structure



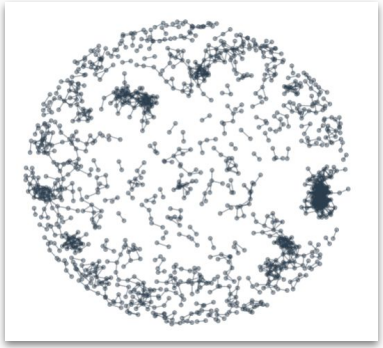
Composition of Groups

Types of Network Analysis

1

Undirected

Strength of Relationship

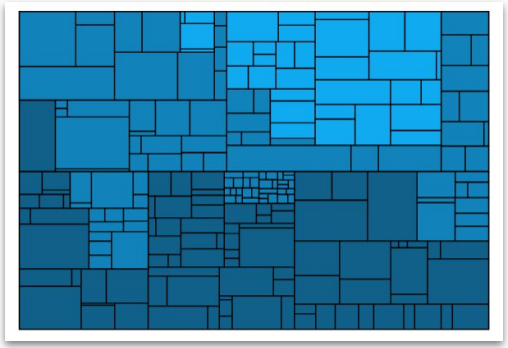


Clustering

2

Directed

Hierarchical Structure



Composition of Groups

Core Concepts

80/20

Adjacency Matrix

Customer Data

```
> credit_card_tbl
# A tibble: 8,950 x 18
   CUST_ID BALANCE BALANCE_FREQUEN~ PURCHASES ONEOFF_PURCHASES INSTALLMENTS_PU~ CASH_ADVANCE PURCHASES_FREQU~ ONEOFF_PURCHASE~
   <chr>     <dbl>            <dbl>     <dbl>           <dbl>            <dbl>        <dbl>           <dbl>           <dbl>
 1 C10001     40.9            0.818      95.4               0             95.4            0           0.167               0
 2 C10002   3202.             0.909       0                 0              0           6443.          0                   0
 3 C10003   2495.             1         773.              773.            0              0           1                   1
 4 C10004   1667.             0.636    1499              1499              0            206.          0.0833              0.0833
 5 C10005    817.             1          16                16              0              0           0.0833              0.0833
 6 C10006   1810.             1        1333.               0           1333.             0           0.667               0
 7 C10007    627.             1        7091.            6403.           688.             0           1                   1
 8 C10008   1824.             1         436.               0            436.             0           1                   0
 9 C10009   1015.             1         861.             661.           200              0           0.333               0.0833
10 C10010    152.             0.545    1282.            1282.             0              0           0.167               0.167
# ... with 8,940 more rows, and 9 more variables: PURCHASES_INSTALLMENTS_FREQUENCY <dbl>, CASH_ADVANCE_FREQUENCY <dbl>,
#   CASH_ADVANCE_TRX <dbl>, PURCHASES_TRX <dbl>, CREDIT_LIMIT <dbl>, PAYMENTS <dbl>, MINIMUM_PAYMENTS <dbl>, PRC_FULL_PAYMENT <dbl>,
#   TENURE <dbl>
```

Adjacency Matrix

```
# A tibble: 8,950 x 8,951
   CUST_ID C10001 C10002 C10003 C10004 C10005 C10006 C10007 C10008 C10009 C10010  C10011 C10012 C10013 C10014 C10015
   <chr>     <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>   <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
 1 C10001  1      0.0842 -0.171  0.125  0.807  0.386 -0.193  0.0384 0.279 -0.0580  0.166   0.608  0.0828 0.00733 0.511
 2 C10002  0.0842 1      -0.276  0.249  0.207 -0.363 -0.448 -0.469  0.0102 0.135  -0.500  -0.127 -0.720 -0.509  0.484
 3 C10003 -0.171 -0.276  1       0.0913 0.0823 -0.156  0.292 -0.0332 0.266  0.226  -0.120   0.353  0.109  0.0579 -0.0656
 4 C10004  0.125  0.249  0.0913  1      0.0419 -0.386  0.222 -0.496  0.464  0.860  -0.516   0.463  -0.196 -0.103  0.288
 5 C10005  0.807  0.207  0.0823  0.0419 1       0.269 -0.223 -0.0651 0.434 -0.168  -0.00797 0.565  -0.0794 -0.115  0.792
 6 C10006  0.386 -0.363 -0.156  -0.386  0.269  1      -0.273  0.696  0.185 -0.390   0.857  -0.0811  0.466  0.591  0.149
 7 C10007 -0.193 -0.448  0.292   0.222 -0.223 -0.273  1       -0.0878 0.214  0.433  -0.146   0.372  0.529  0.285 -0.473
 8 C10008  0.0384 -0.469 -0.0332 -0.496 -0.0651  0.696 -0.0878  1      0.125 -0.440   0.938  -0.259  0.685  0.548 -0.184
 9 C10009  0.279  0.0102 0.266   0.464  0.434   0.185  0.214  0.125  1       0.508  0.0476  0.302  0.151  0.465  0.413
10 C10010 -0.0580  0.135  0.226   0.860 -0.168  -0.390  0.433 -0.440  0.508  1      -0.481   0.303 -0.175  0.126 -0.0478
# ... with 8,940 more rows, and 8,935 more variables: C10016 <dbl>, C10017 <dbl>, C10018 <dbl>, C10019 <dbl>, C10020 <dbl>,
#   C10021 <dbl>, C10022 <dbl>, C10023 <dbl>, C10024 <dbl>, C10025 <dbl>, C10026 <dbl>, C10027 <dbl>, C10028 <dbl>, C10029 <dbl>,
#   C10030 <dbl>, C10031 <dbl>, C10032 <dbl>, C10033 <dbl>, C10034 <dbl>, C10035 <dbl>, C10036 <dbl>, C10037 <dbl>, C10038 <dbl>,
#   C10039 <dbl>, C10040 <dbl>, C10041 <dbl>, C10043 <dbl>, C10044 <dbl>, C10045 <dbl>, C10046 <dbl>, C10047 <dbl>, C10048 <dbl>,
#   C10049 <dbl>, C10050 <dbl>, C10051 <dbl>, C10052 <dbl>, C10053 <dbl>, C10054 <dbl>, C10055 <dbl>, C10056 <dbl>, C10057 <dbl>,
#   C10058 <dbl>, C10059 <dbl>, C10060 <dbl>, C10061 <dbl>, C10062 <dbl>, C10063 <dbl>, C10064 <dbl>, C10065 <dbl>, C10067 <dbl>,
#   C10068 <dbl>, C10069 <dbl>, C10070 <dbl>, C10071 <dbl>, C10072 <dbl>, C10073 <dbl>, C10074 <dbl>, C10075 <dbl>, C10077 <dbl>,
#   C10078 <dbl>, C10079 <dbl>, C10080 <dbl>, C10081 <dbl>, C10082 <dbl>, C10083 <dbl>, C10084 <dbl>, C10085 <dbl>, C10086 <dbl>,
#   C10087 <dbl>, C10088 <dbl>, C10089 <dbl>, C10090 <dbl>, C10092 <dbl>, C10093 <dbl>, C10094 <dbl>, C10095 <dbl>, C10096 <dbl>,
#   C10097 <dbl>, C10098 <dbl>, C10099 <dbl>, C10100 <dbl>, C10101 <dbl>, C10102 <dbl>, C10103 <dbl>, C10104 <dbl>, C10105 <dbl>,
#   C10106 <dbl>, C10107 <dbl>, C10108 <dbl>, C10109 <dbl>, C10110 <dbl>, C10111 <dbl>, C10112 <dbl>, C10113 <dbl>, C10114 <dbl>,
#   C10115 <dbl>, C10116 <dbl>, C10117 <dbl>, C10118 <dbl>, C10119 <dbl>, ...
```

Key Concept

N x N matrix of **Relationship Strength** (Measure)

Nodes & Edges

Nodes

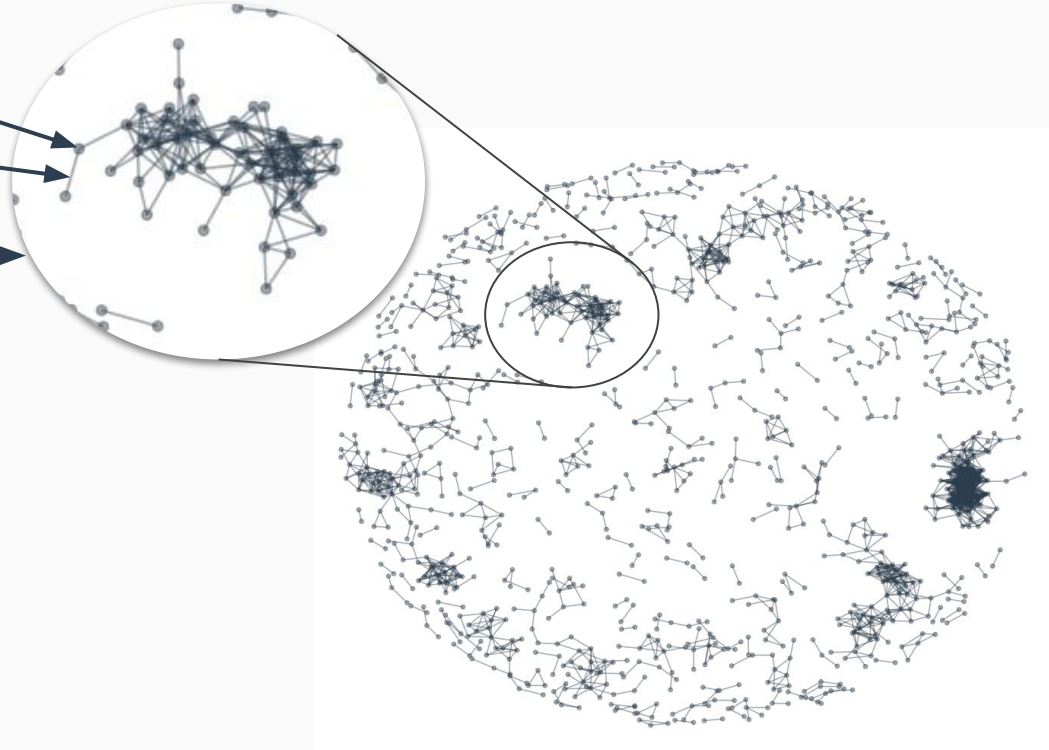
- Customers

Edges

- Relationship Strength

Clusters (Groups)

- Densely Connected Web



Key Concept

Groups (Clusters) have more edges connecting more nodes at a given relationship **threshold**.

Pruning & Threshold

Pruning

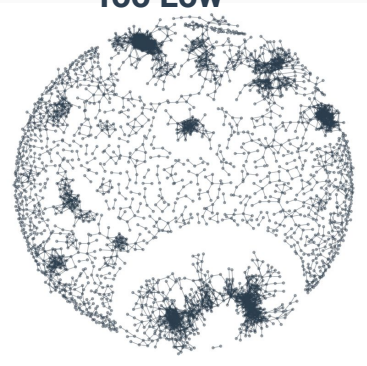
- Filtering to reduce to the **most “influential” nodes**
- We use a **threshold** to find an optimal visualization that explains the groups

Threshold = 0.9999

Too High

Threshold = 0.99

Too Low



Threshold = 0.996

Just Right!

Key Concept

Data mining is subjective. Where do we cut off?

Network Analysis

Software

tidygraph

Tidy Network Data

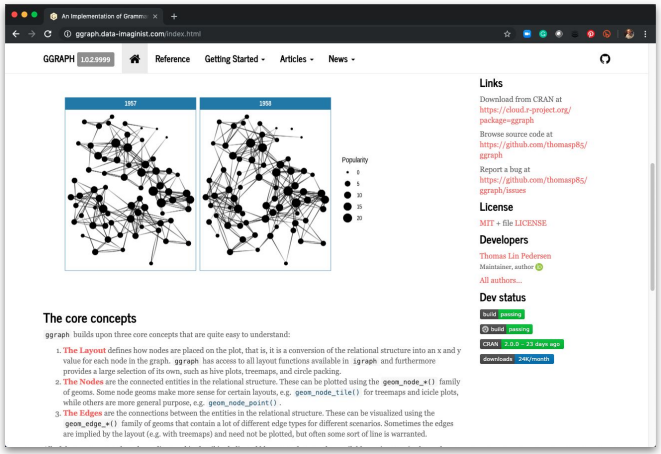
- Combines Node and Edge Data inside 1 tidygraph object
- Makes it super easy to work with network data
- Can activate() inner node and edge tbls to manipulate them
- Can apply special network analysis functions like group_components() & centrality_degree()

```
# A tbl_graph: 1125 nodes and 2156 edges
#
# An undirected simple graph with 247 components
#
# Node Data: 1,125 x 1 (active)
  name
  <chr>
1 C10278
2 C16180
3 C17657
4 C14958
5 C11181
6 C12206
# … with 1,119 more rows
#
# Edge Data: 2,156 x 3
   from    to weight
  <int> <int>  <dbl>
1     1   711  0.996
2     2   712  0.997
3     3   712  0.998
# … with 2,153 more rows
```

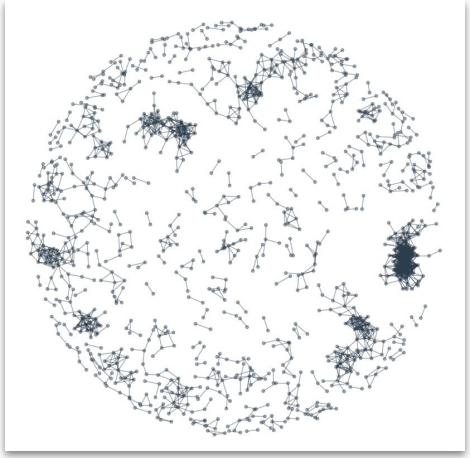
https://github.com/thomasp85/tidygraph

ggraph

Visualization Package for tidygraph data




```
152 # 7.0 NETWORK VISUALIZATION -----
153
154 customer_correlation_matrix %>%
155
156     prep_corr_matrix_for_tbl_graph(edge_limit = 0.996) %>%
157
158     as_tbl_graph(directed = FALSE) %>%
159
160     ggraph(layout = "kk") +
161     geom_edge_link(alpha = 0.5, color = palette_light()["blue"]) +
162     geom_node_point(alpha = 0.5, color = palette_light()["blue"]) +
163     theme_graph(background = "white")
164
```

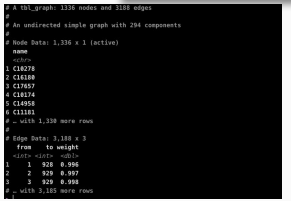


https://ggraph.data-imaginist.com/

Customer Segmentation Workflow

Step-By-Step

CUST_ID	C10001	C10002	C10003	C10004	C10005
<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1 C10001	1	0.0842	-0.171	0.125	0.807
2 C10002	0.0842	1	-0.276	0.249	0.207
3 C10003	-0.171	-0.276	1	0.0913	0.0823
4 C10004	0.125	0.249	0.0913	1	0.0419
5 C10005	0.807	0.207	0.0823	0.0419	1



Start

1

Data Clean & Transform

Exploratory Data Analysis

2

Adjacency Matrix, tidygraph, & ggraph Visualizations

Develop Segments

3

H2O & LIME

Predict & Explain Customer Segments

Finish

30-Min Demo

Analyze Customer Networks

Secret Tactics for

Network Analysis

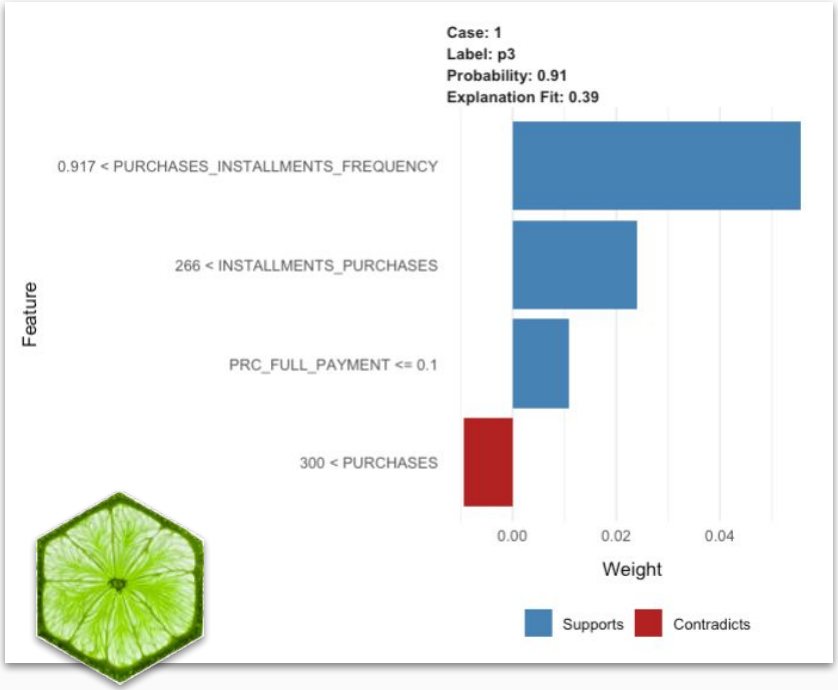
Use these tips to
increase your customer segmentation explainability

Pro Tip

Use H2O & LIME to EXPLAIN WHY

```
> h2o.predict(h2o_model, newdata = as.h2o(credit_card_group_tbl)) %>%
+   as_tibble()
  |======================================================================
  |======================================================================
# A tibble: 1,125 x 7
   predict      p1        p2        p3      p4        p5   Other
   <fct>     <dbl>     <dbl>     <dbl>   <dbl>     <dbl>   <dbl>
 1 Other   0       0.0000704 0.0228    0       0         0.977
 2 Other   0.0232  0.0000717 0.0000553 0       0.00376   0.973
 3 Other   0       0.0000737 0.0238    0       0.000107  0.976
 4 Other   0.00643 0.0000724 0.0000558 0.00343 0.000105  0.990
 5 Other   0       0.0000720 0.0000555 0       0.000104  1.000
 6 3       0       0.0000704 0.909     0       0.000102  0.0909
 7 3       0       0.0000761 0.995     0       0.000110  0.00491
 8 1       0.984   0.0000735 0.0000567 0.00349 0.00010
 9 Other   0.195   0.0000602 0.0000464 0.00285 0.00008
10 Other   0       0.0000737 0.0000568 0       0
# … with 1,115 more rows
```

H2O.ai

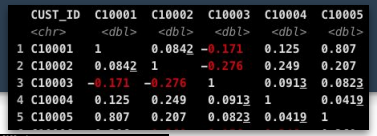


Data Science Transformation

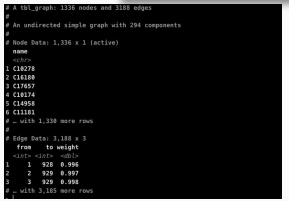
Skills that are needed to do what we just did

Customer Segmentation Workflow

Step-By-Step



	CUST_ID	C10001	C10002	C10003	C10004	C10005
	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	C10001	1	0.0842	-0.171	0.125	0.807
2	C10002	0.0842	1	-0.276	0.249	0.207
3	C10003	-0.171	-0.276	1	0.0913	0.0823
4	C10004	0.125	0.249	0.0913	1	0.0419
5	C10005	0.807	0.207	0.0823	0.0419	1



Start

1

2

3

Finish

Data Clean & Transform

Exploratory Data Analysis

101 & 201

Adjacency Matrix, tidygraph, & ggraph Visualizations

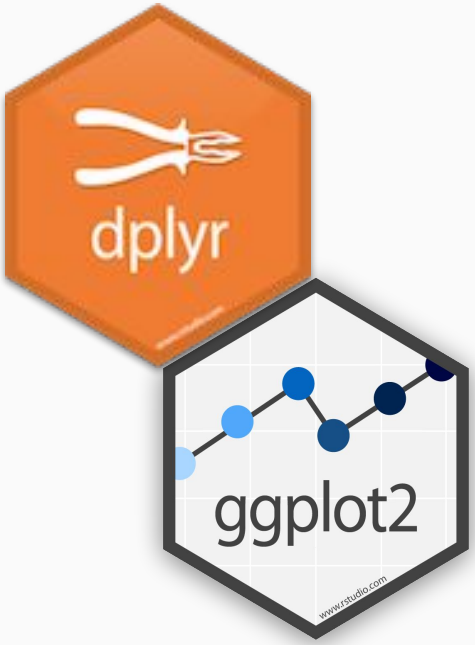
Lab 19

H2O & LIME

Predict & Explain Customer Segments

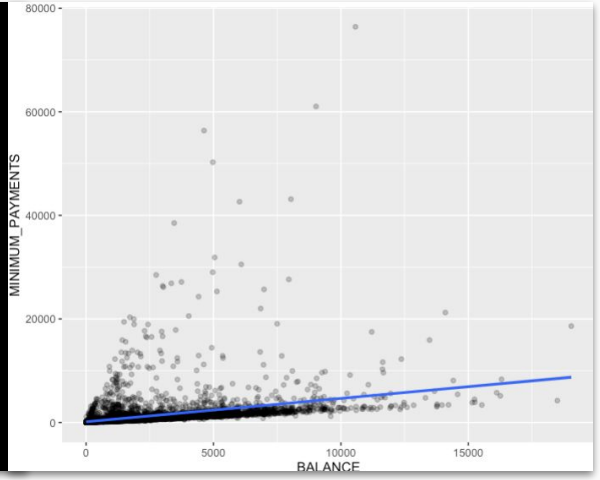
201

dplyr, ggplot2



101 & 201

```
30  # 2.1 Minimum Payments has NA (missing data)
31
32  credit_card_tbl %>%
33      pull(MINIMUM_PAYMENTS) %>%
34      quantile(na.rm = TRUE)
35
36  credit_card_no_missing_tbl <- credit_card_tbl %>%
37      select_if(is.numeric) %>%
38      filter(!is.na(MINIMUM_PAYMENTS)) %>%
39      filter(!is.na(CREDIT_LIMIT))
40
41  credit_card_no_missing_tbl %>%
42      binarize() %>%
43      correlate(target = MINIMUM_PAYMENTS__825.49646275_Inf) %>%
44      plot_correlation_funnel()
45
46  credit_card_tbl %>%
47      ggplot(aes(BALANCE, MINIMUM_PAYMENTS)) +
48      geom_point(alpha = 0.25) +
49      geom_smooth(method = "lm")
50
```



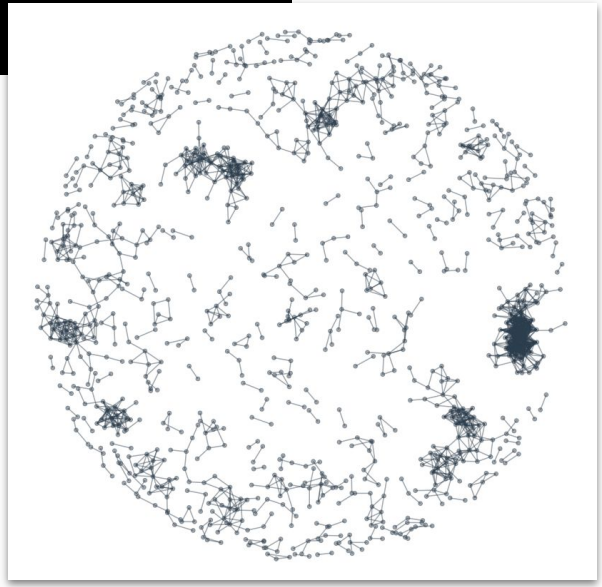
Adjacency Matrix, tidygraph, & ggraph

CUST_ID	C10001	C10002	C10003	C10004	C10005
<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1 C10001	1	0.0842	-0.171	0.125	0.807
2 C10002	0.0842	1	-0.276	0.249	0.207
3 C10003	-0.171	-0.276	1	0.0913	0.0823
4 C10004	0.125	0.249	0.0913	1	0.0419
5 C10005	0.807	0.207	0.0823	0.0419	1

```
# A tbl_graph: 1336 nodes and 3188 edges
#
# An undirected simple graph with 294 components
#
# Node Data: 1,336 x 1 (active)
  name
  <chr>
1 C10278
2 C16180
3 C17657
4 C10174
5 C14958
6 C11181
# … with 1,330 more rows
#
# Edge Data: 3,188 x 3
   from    to weight
  <int> <int>  <dbl>
1     1   928  0.996
2     2   929  0.997
3     3   929  0.998
# … with 3,185 more rows
```

```
153 # 7.0 NETWORK VISUALIZATION ----
154
155 customer_correlation_matrix %>%
156
157     prep_corr_matrix_for_tbl_graph(edge_limit = 0.996) %>%
158
159     as_tbl_graph(directed = FALSE) %>%
160
161     ggraph(layout = "kk") +
162     geom_edge_link(alpha = 0.5, color = palette_light()["blue"]) +
163     geom_node_point(alpha = 0.5, color = palette_light()["blue"]) +
164     theme_graph(background = "white")
165
166
167
```

Lab 19

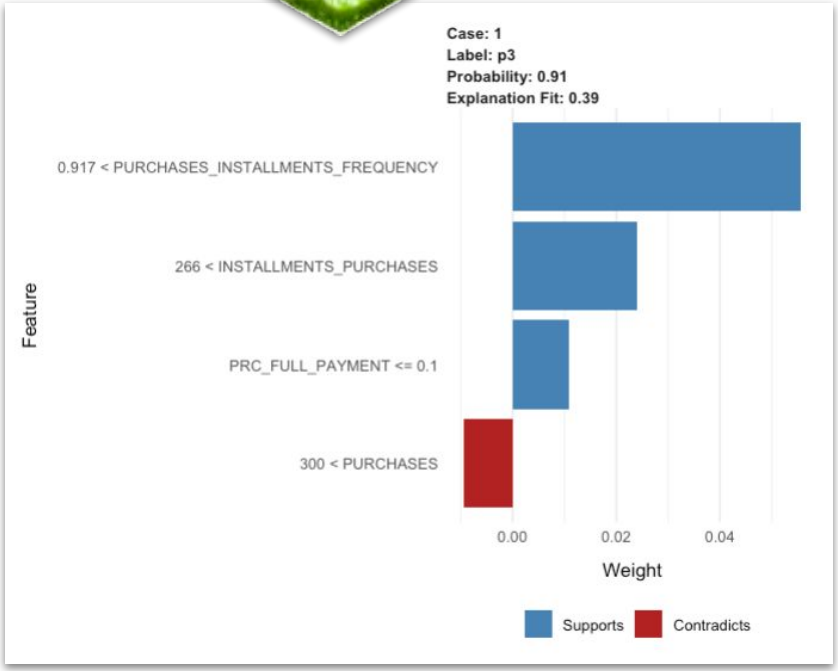


ggplot2 & purrr

H2O.ai

201

```
> h2o.predict(h2o_model, newdata = as.h2o(credit_card_group_tbl)) %>%
+   as_tibble()
  |======================================================================
  |======================================================================
# A tibble: 1,125 x 7
   predict      p1        p2        p3      p4        p5   Other
   <fct>     <dbl>     <dbl>     <dbl>   <dbl>     <dbl>   <dbl>
 1 Other   0       0.0000704 0.0228    0       0         0.977
 2 Other   0.0232  0.0000717 0.0000553 0       0.00376   0.973
 3 Other   0       0.0000737 0.0238    0       0.000107  0.976
 4 Other   0.00643 0.0000724 0.0000558 0.00343 0.000105  0.990
 5 Other   0       0.0000720 0.0000555 0       0.000104  1.000
 6 3       0       0.0000704 0.909     0       0.000102  0.0909
 7 3       0       0.0000761 0.995     0       0.000110  0.00491
 8 1       0.984   0.0000735 0.0000567 0.00349 0.000106  0.0127
 9 Other   0.195   0.0000602 0.0000464 0.00285 0.0000870 0.802
10 Other   0       0.0000737 0.0000568 0       0         1.000
# … with 1,115 more rows
```



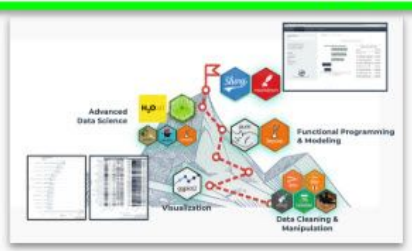
Business Science University

Our program that will TRANSFORM YOU in weeks, not years.

The program that will deliver YOUR Transformation

R-TRACK

LEARNING LABS PRO



Do Business Projects
Climb the Hill

Build Production-Ready
Web Apps

Complete 1-Hour Courses
Continuous Education

Start

1

2

3

Finish

Analysis Courses
101 & 201

App Development Courses
102

Learning Labs PRO

Everything is **Taken Care of** For You in Our Platform

3-Course R-Track System

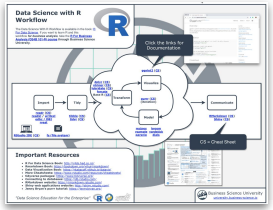
Business Analysis with R
(DS4B 101-R)

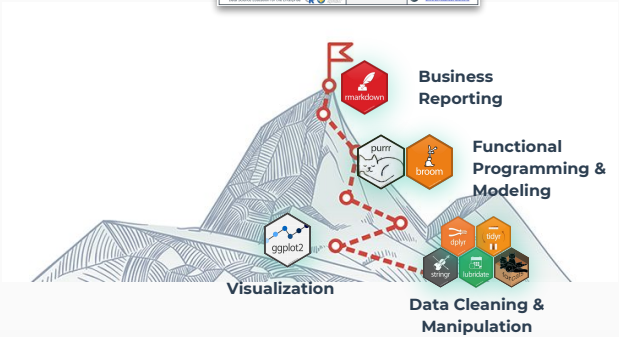
Data Science For Business with R
(DS4B 201-R)

R Shiny Web Apps For Business
(DS4B 102-R)

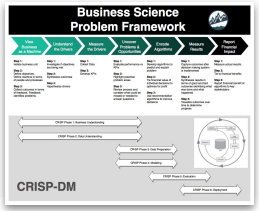
Project-Based Courses with Business Application

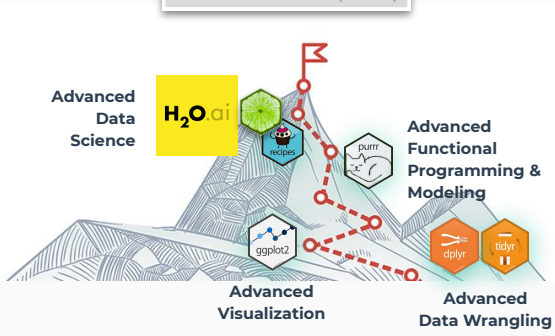
Data Science Foundations
7 Weeks



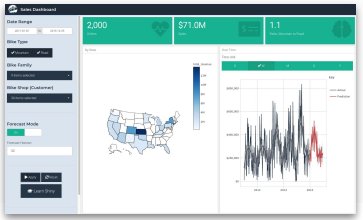


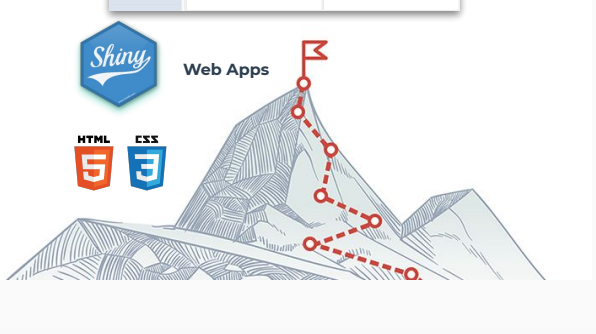
Machine Learning & Business Consulting
10 Weeks





Web Application Development
4 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

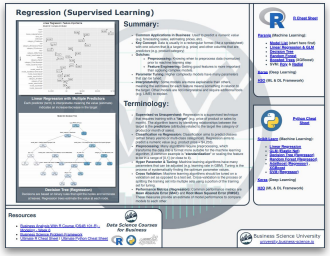
- **Machine Learning - Week 6 (8 hours of Video Lessons)**
  - **Clustering (3 hours)**
  - **Regression (5 hours)**
  - **2 Challenges**

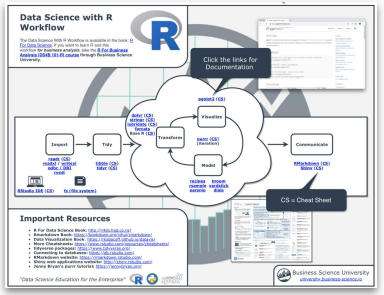
- Learn Business Reporting - Week 7
  - RMarkdown & plotly
  - 2 Project Reports:
    1. Product Pricing Algo
    2. Customer Segmentation

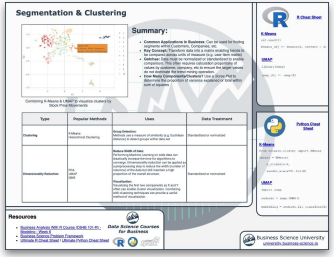
Business Analysis with R (DS4B 101-R)

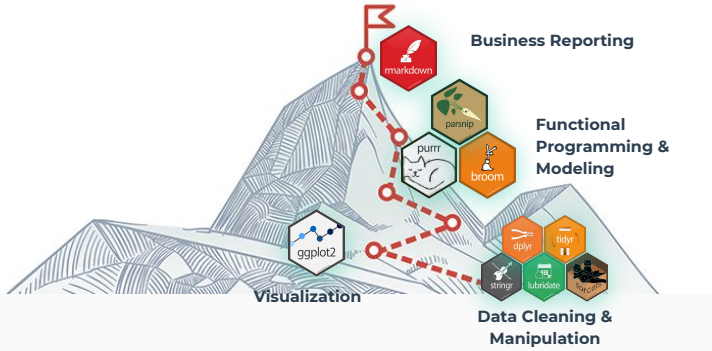
Data Science Foundations

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**

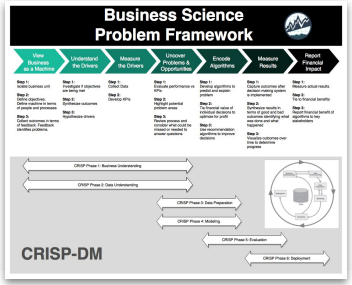
Return-On-Investment - Weeks 7, 8, 9

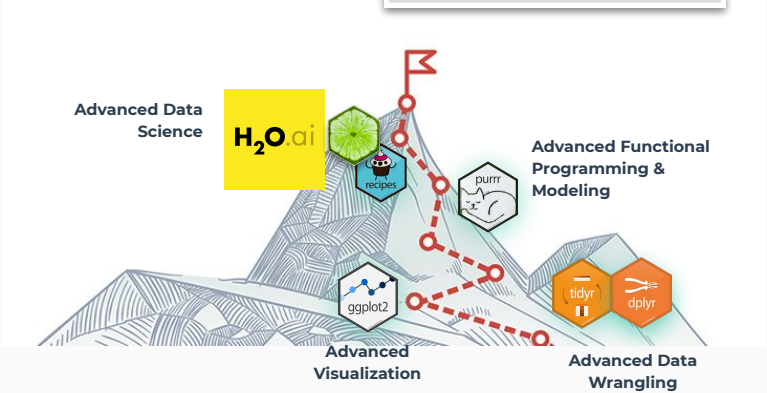
- **Expected Value Framework**
- **Threshold Optimization**
- **Sensitivity Analysis**
- **Recommendation Algorithm**

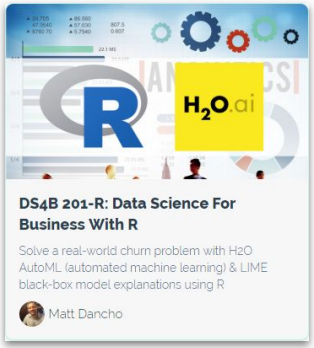
Data Science For Business (DS4B 201-R)

Machine Learning & Business Consulting

10 Weeks







Key Benefits

Learn Shiny & Flexdashboard
- Build Applications
- Learn Reactive Programming
- Integrate Machine Learning

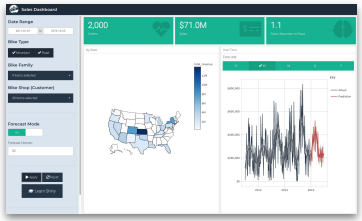
App #1: Predictive Pricing App
- Model Product Portfolio
- XGBoost Pricing Prediction
- Generate new products instantly

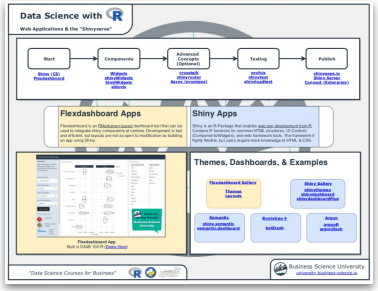
App #2: Sales Dashboard with Demand Forecasting
- Model Demand History
- Segment Forecasts by Product & Customer
- XGBoost Time Series Forecast
- Generate new forecasts instantly

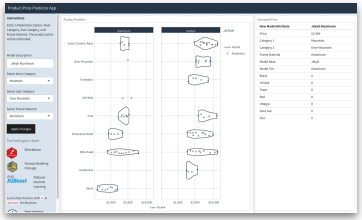
Shiny Apps for Business (DS4B 102-R)

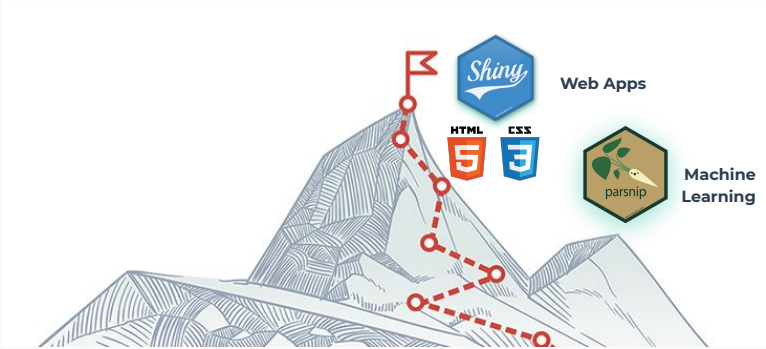
Web Application Development

4 Weeks









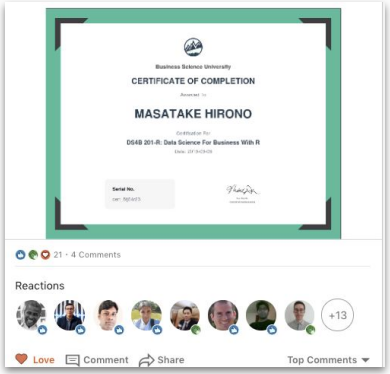
Success Story

Masatake Hirono

- Took DS4B 201-R
- Completed the 10-Week Course
- **Landed a Job at one of the most Prestigious Management Consulting Firms**

"This course showed me how to place data analytics in real business settings."

#Business Science Success



15% OFF PROMO Code: learninglabs

R-TRACK BUNDLE

Bundle - DS For Business + Web Apps (Level 1): R-Track - Courses 101, 102,

3 Course Bundle

0% COMPLETE

DS4B 101-R: Business Analysis With R

Your Data Science Journey Starts Now! Learn the fundamentals of data science for business with the tidyverse.

Matt Dancho

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

Matt Dancho

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

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

Matt Dancho

Get started now!

◉	**Paid Course** 15% COUPON DISCOUNT	~~$1,149~~ *$976.65* Enroll
○	**3 Monthly Payments** 15% COUPON DISCOUNT 3X Monthly	~~3 payments of $449/m~~ *3 payments of $381.65/m*
○	**6 Low Monthly Payments** 15% COUPON DISCOUNT 6X Payment Plan	~~6 payments of $234/m~~ *6 payments of $198.90/m*
○	**12 Low Monthly Payments** 15% COUPON DISCOUNT 12X Plan	~~12 payments of $125/m~~ *12 payments of $106.25/m*

Begin Learning Today

university.business-science.io