The modern business world is captivated by data science and analytics. It seems like in almost every enterprise there is talk of needing to bulk up the data science and analytics functions to solve all conceivable business problems and chart the company towards endless success. In more cash-rich operations, management often seeks to go even beyond analytics and towards vaunted ‘AI’ solutions, generally caught up in Silicon Valley hype and expecting the state of AI to be decades (or even centuries) beyond its actual current state. Whether their company is large or small, countless contemporary managers are trying to win the data science and analytics race, but often come up frustrated when their expectations don’t meet reality.

**4 Common Reasons Why Data Science and Analytics Initiatives Are Slow To Gain Momentum**

There are many reasons why this may be the case in any particular organization, though companies which share the struggle typically exhibit at least one of the following traits:

1. **Fractured data.** How profitable is a certain SKU? It depends on who you ask. Did our marketing campaign generate sufficient ROI? Perhaps according to one analysis, but not another. Many managers can relate to this exceedingly-common experience. Different departments and teams manage their own sources of data, and their staff have their own custom methods of parsing and interpreting that data, which consequently leads to different answers when senior management asks questions to gauge the health of the business or viability of a project. In short, there is no ‘single source of truth.’ This is often related to another common corporate issue:

1. **Siloed departments.** When a company is small, typically a startup on through the first several years, it tends to operate with a relatively-flat and collaborative organizational structure. This is partly because smaller organizations often don’t have the money to build large teams of varied functions, but also because small and growing businesses tend to be more agile; they move faster than larger organizations because they have to in order to stay solvent and capitalize on their momentum. This is how a small company outmaneuvers larger competitors so that it, too, can eventually scale and become a big company. As a company grows and additional layers of management are added, new departments created, and ownership / senior management becomes increasingly-separated from daily operations, bureaucratic structures tend to crop up, leading to silos. Departments in large organizations may rarely speak to one another or know what is happening in other parts of the company. This means that not only may departments maintain disparate sources of data; it also often means their analyses and uses of the data may be operating at cross-purposes without them even realizing. But even if there is a single repository of data and the various departments are coordinating their use, many companies find themselves running into yet another issue:

1. **Biased or flawed analysis.** As much as we may like to think business data is unbiased and objective, factors such as corporate politics, lack of business acumen, or simple human error can result in skewed interpretations of that data. While data itself is non-sentient and unbiased, it doesn’t actually convey ‘information’ until interpreted. What does the data mean? The job of the analyst is to answer that question and tell a story which makes the data practical to management. While human involvement in this process is necessary and vital, the more you depend on it, the greater likelihood of flawed results. In some organizations, this leads to yet another problem:

1. **Mistrust in the data.** This often happens in more traditional businesses or older organizations outside the tech/e-commerce sectors, where senior management may have instituted some form of analytics or data science initiative because they heard about it on CNBC, Fox Business, Harvard Business Review, etc., but the initiative never really received sufficient attention, funding, or buy-in from the bulk of the staff. Institutional memory has a certain way of determining ‘how we do things around here,’ and when the data disagrees with the existing paradigm, the data is assumed to be wrong. One of the best contemporary examples of this phenomenon is the baseball analytics field pioneered at the Oakland A’s and documented in Michael Lewis’ book, *Moneyball*.

Sometimes data is rightly mistrusted because of the aforementioned problems in how it is collected and presented, but this issue can occur even in organizations with good data practices. It is a dangerous attitude for management to have, because even if the data has been properly collected and structured, various departments are collaborating, and the analysts and data scientists have done a good job of avoiding biases and other skews, reliable analysis can still be discarded due to a cultural mismatch. When this happens, management may pay lip service to analytics and data science, but will continue marching onward based on their personal preferences.

**What are some quick-win strategies to address these common problems?**

So what exactly is the manager to do in today’s environment? Most sensible, informed business people have by now realized that data science and analytics are going to be critical components of business success from here on out, but often the aforementioned problems run deep and present enormous roadblocks. And what about startups or small businesses that don’t have these same problems, but also doesn’t have the budget to build a full-scale analytics or data science team?

Thankfully, the general trend of markets is deflationary; stuff gets cheaper over time as it becomes more available, and this includes analytics and data science tools just as much as consumer goods. Consequently, open source tools which solve some of these issues — such as R, Python, and RemixAutoML — are beginning to appear, allowing even entrepreneurs and small enterprises access to capabilities which even just a short number of years ago would be hard to come by for the Fortune 500.

**Can you provide specifics? How can SMB’s and startups tap into this?**

Let’s start with one example based off an Amazon model. If you’re looking for a company which leads in leveraging machine learning and AI to optimize sales and profits, look no further than Amazon. From AI-enabled products like Alexa, to AI-based product recommenders on Amazon’s website, to AWS machine learning software, Amazon has no shortage of uses for machine learning and AI. Even Jeff Bezos said in his [2016 Letter to Amazon Shareholders](https://blog.aboutamazon.com/company-news/2016-letter-to-shareholders/),

*These big trends are not that hard to spot (they get talked and written about a lot), but they can be strangely hard for large organizations to embrace. We’re in the middle of an obvious one right now: machine learning and artificial intelligence.*

*At Amazon, we’ve been engaged in the practical application of machine learning for many years now. Some of this work is highly visible: our autonomous Prime Air delivery drones; the Amazon Go convenience store that uses machine vision to eliminate checkout lines; and Alexa,1 our cloud-based AI assistant. (We still struggle to keep Echo in stock, despite our best efforts. A high-quality problem, but a problem. We’re working on it.)*

*But much of what we do with machine learning happens beneath the surface. Machine learning drives our algorithms for demand forecasting, product search ranking, product and deals recommendations, merchandising placements, fraud detection, translations, and much more. Though less visible, much of the impact of machine learning will be of this type – quietly but meaningfully improving core operations.*

Superficial advice like “look at how successful Amazon is, so just do what they do” won’t help you. It’s almost impossible to try. Amazon’s R&D budget alone is $22 billion, dwarfing even the total annual revenues of most large companies. In all likelihood, your company’s R&D budget is $0. And since many companies don’t understand the significant investment required to be a leader in machine learning and AI (like Amazon), odds are you need a more pragmatic approach that can be done on a shoestring budget. Then, if you can pick up some quick-win ROI, management may see justification for further investment in machine learning and AI initiatives.

Example of Amazon’s “Frequently Bought Together” Algorithm

This is where you can use an open-source, automated machine learning tool (such as RemixAutoML) which allows small-to-medium sized businesses and startups to build Amazon-style ‘frequently bought together’ product recommendations with just a single line of code and very little data. Your organization may have a messy data warehouse, but the only data points needed are:

* invoice or transaction data, such as invoice or sales order numbers
* a list of item or SKU numbers purchased on each invoice or sales order

This data can easily be extracted from either your Point-Of-Sale system or your e-commerce platform. Using that data, an analytics professional can create a machine learning model with a single line of code in RemixAutoML capable of competing with even the largest big box retailers, thus lifting conversion rates, average order values, and repurchase rates while increasing market share.

Tools like RemixAutoML help overcome hurdles such as fractured data (since only few data points are required), mistrust of data and siloed objectives (as everyone can utilize and see immediate ROI), and biased analysis (as the tool uses laws of probability and machine learning to reduce bias).

As a quick example, consider this online retail dataset from an e-commerce company in the UK:

Online Retail Data Set from a UK eCommerce Company | Source: UCI Machine Learning Repository

Again, the only two data points needed are invoice number (called InvoiceNo in this data set) and item number (called StockCode in the data set).

Running the following R code using RemixAutoML yields the end product below: a table of products frequently bought together based on highest statistical significance. This would equip the sales organization to leverage the table whenever it tries to upsell a customer with Product B given that Product A has been added to their shopping cart. This upsell could be facilitated by adding ‘frequently bought together’ recommenders on the website, in a personalized sales email, or at the Point-Of-Sale in a brick and mortar store. For more technical users, details are provided in the R code comments.

Sample Output from RemixAutoML’s AutoMarketBasketModel

The output mirrors Amazon’s ‘frequently bought together’ algorithm. The column called **StockCode\_LHS** means “StockCode Left-Hand Side” and is the product the customer has added to their cart. The column called **StockCode\_RHS** means “StockCode Right-Hand Side” and is the product that is most ‘frequently bought together’ with **StockCode\_LHS** at the highest statistically significant level.

**Support** means the percent of total transactions in which StockCode\_LHS and StockCode\_RHS appear together. **Confidence** means the conditional probability that StockCode\_RHS is purchased given that StockCode\_LHS has been added to the cart. The columns called **Lift, Chi\_SQ,** and **P\_Value** are all statistical significance metrics of the relationship between StockCode\_LHS and StockCode\_RHS. **RuleRank** is the ranking system that RemixAutoML uses to rank the market basket affinities for you.

*Nick Gausling is a businessman and investor who has worked across multiple industries with companies both small and large. You can connect with Nick at* [*https://www.linkedin.com/in/nick-gausling/*](https://www.linkedin.com/in/nick-gausling/) *or* [*www.nickgausling.com*](http://www.nickgausling.com/)

*Douglas Pestana is a data scientist with 10+ years experience in data science and machine learning at Fortune 500 and Fortune 1000 companies. He is one of the co-founders of Remix Institute.*

library(data.table)

library(dplyr)

library(magrittr)

library(RemixAutoML)

# IMPORT ONLINE RETAIL DATA SET THEN CLEAN DATA SET-------

# Original Source: UCI Machine Learning Repository - <https://archive.ics.uci.edu/ml/datasets/online+retail>

# download file from Remix Insitute Box account

online\_retail\_data = data.table::fread("<https://remixinstitute.box.com/shared/static/v2c7mkkqm9eswyqbzkg5tbqqzfa1885v.csv>", header = T, stringsAsFactors = FALSE)

# create a flag for cancelled invoices

online\_retail\_data$CancelledInvoiceFlag = ifelse(substring(online\_retail\_data$InvoiceNo, 1, 1) == 'C', 1, 0)

# create a flag for negative quantities

online\_retail\_data$NegativeQuantityFlag = ifelse(online\_retail\_data$Quantity < 0, 1, 0)

# remove cancelled invoices and negative quantitites

online\_retail\_data\_clean = online\_retail\_data %>% dplyr::filter(., CancelledInvoiceFlag != 1) %>%

dplyr::filter(., NegativeQuantityFlag != 1)

# PREP DATA SET FOR MODELING -------------

# for market basket analysis models, you'll need data grouped by invoice (InvoiceNo) and item number (StockCode). Then you can sum up the units sold.

online\_retail\_data\_for\_model = online\_retail\_data\_clean %>% dplyr::group\_by(., InvoiceNo, StockCode) %>%

dplyr::summarise(., Quantity = sum(Quantity, na.rm = TRUE) )

# RUN AUTOMATED MARKET BASKET ANALYSIS (PRODUCT RECOMMENDER) IN RemixAutoML -----------

# the AutoMarketBasketModel from RemixAutoML automatically converts your data,

# runs the market basket model algorithm, and adds Chi-Square statistics for significance

market\_basket\_model = RemixAutoML::AutoMarketBasketModel(

data = online\_retail\_data\_for\_model,

OrderIDColumnName = "InvoiceNo",

ItemIDColumnName = "StockCode"

)

# add product Description

# left-hand side products

StockCode\_LHS\_description = online\_retail\_data\_clean %>% dplyr::select(., StockCode, Description) %>%

dplyr::rename(., StockCode\_LHS = StockCode,

Description\_LHS = Description

) %>%

dplyr::distinct(., StockCode\_LHS, .keep\_all = TRUE)

# right-hand side products

StockCode\_RHS\_description = online\_retail\_data\_clean %>% dplyr::select(., StockCode, Description) %>%

dplyr::rename(., StockCode\_RHS = StockCode,

Description\_RHS = Description

)%>%

dplyr::distinct(., StockCode\_RHS, .keep\_all = TRUE)

# merge

market\_basket\_model\_final = merge(market\_basket\_model, StockCode\_RHS\_description, by = 'StockCode\_RHS', all.x = TRUE)

market\_basket\_model\_final = merge(market\_basket\_model\_final, StockCode\_LHS\_description, by = 'StockCode\_LHS', all.x = TRUE)

# re-sort by StockCode\_LHS and RuleRank

market\_basket\_model\_final = market\_basket\_model\_final[order(StockCode\_LHS, RuleRank),]

# view results

View(market\_basket\_model\_final)