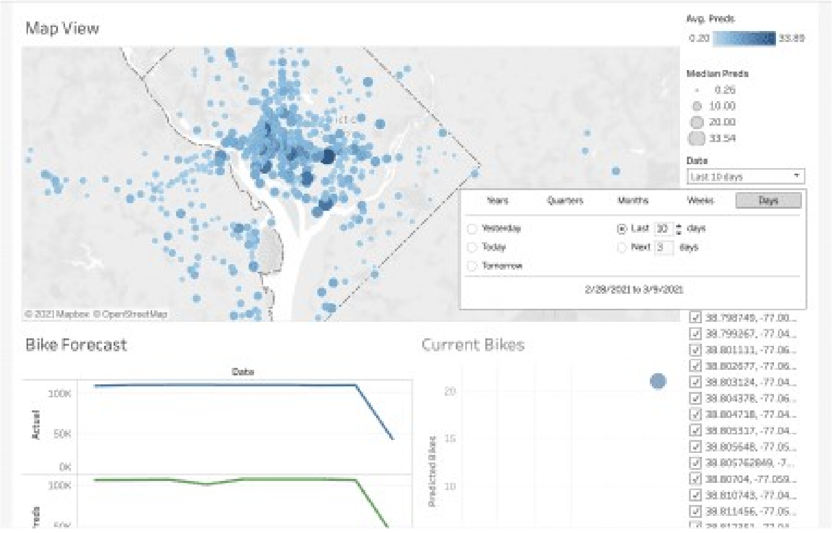


Date: Last 10 days

● Last 10 days

2/28/2021 to 3/9/2021

In recent posts we have explored how organizations can make better decisions by focusing on data science and BI collaboration. In this post, we will look at one type of collaboration accomplished through *data handoffs*, which we define as

> Datasets stored in databases that are created by data scientists and shared with BI analysts

In future posts we will explore additional collaboration techniques that enable real-time interactions between BI tools and data science work.

Why Data Handoffs?

Often data science teams are created to answer particularly hard questions. They work with large messy data, often from unstructured or novel sources, and then apply advanced analytical methods and statistical rigor. As part of this work, data science teams create visualizations, dashboards, and interactive applications to influence decisions. While data scientists can usually accomplish these tasks most effectively using reproducible code, they are typically resource constrained and discover they can't:

- Address every question posed by the data.
- Adapt results to each audience in the organization.
- Satisfy audiences who wish to quickly explore the data themselves.

In these scenarios, data scientists can use data handoffs to address these issues and leverage existing BI capabilities. By exporting the novel data sources, predictions, and calculated features they have created, data scientists can:

- Collaborate with larger BI teams.
- Increase the visibility and re-use of their work.
- Broaden and democratize the access to advanced data.

These benefits create a virtuous cycle. Data scientists can share novel data, and then BI teams can explore that data, identify new problems, and propose solutions that require further validation from the data science team.

How Data Handoffs Work

Data scientists typically begin by building analytic notebooks that do the hard work of cleaning unstructured data, generating calculated features, applying model predictions, or pre-processing big data. These notebooks can conclude by writing final tidy datasets to an analytics data warehouse. The notebooks are then deployed into a production environment where they can monitored and scheduled. An illustrative workflow using data handoffs for the Washington, D.C. bikeshare program is shown below in

Figure 1. The bike-predict GitHub repository contains a more detailed description and all the code behind this workflow.

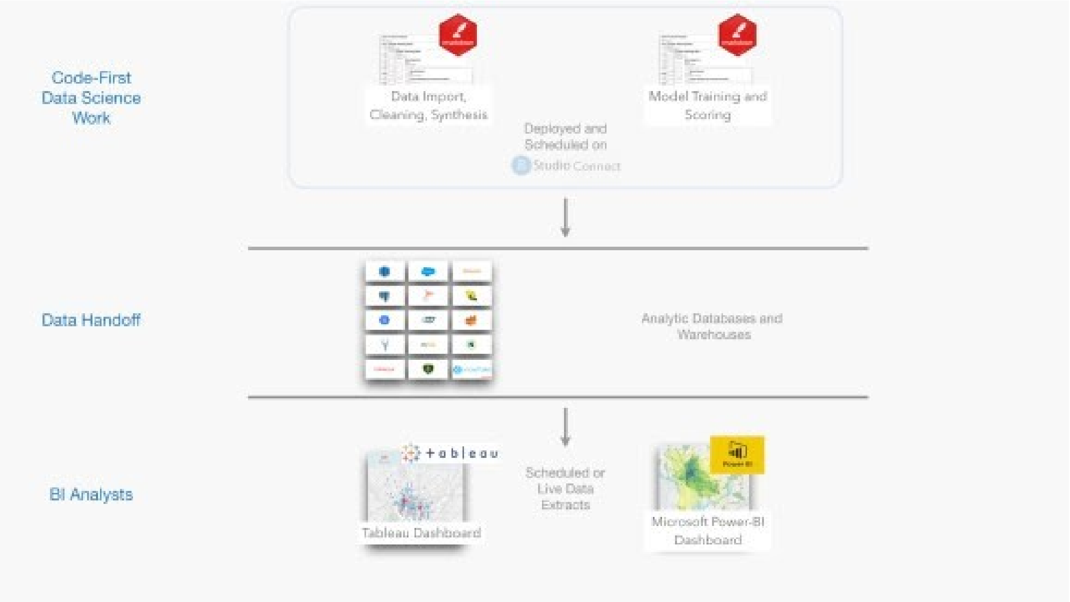
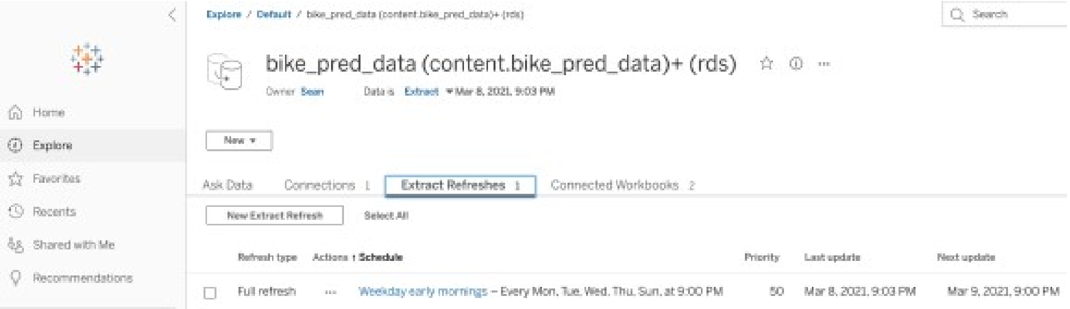

Code-First Data Science Work
Data Import, Cleaning, Synthesis
Model Training and Scoring
Deployed and Scheduled on
RStudio Connect
Data Handoff
Analytic Databases and Warehouses
BI Analysts
tableau
Tableau Dashboard
Scheduled or Live Data Extracts
Power BI
Microsoft Power-BI Dashboard

Figure 1: Illustration of the use of handoff data for forecasting Washington, D.C. bikeshare

The R Markdown notebook in this example uses dbplyr to query data from a database, and then uses a trained xgboost model to create a forecast. The resulting forecast is written back to the database. This document is then deployed and scheduled on RStudio Connect, which also supports scheduling Jupyter Notebooks. While this example focuses on creating batch model predictions, other common tasks could include data wrangling in Spark, accessing novel data sources such as web scraping, feature generation, or advanced data verification.

Data scientists are able to create analytic pipelines that generate rich and tidy data. This activity often complements the existing work of data engineering teams responsible for wrangling data across the entire organization.

Once the data is written to the database it becomes easily accessible to the BI team. BI tools such as Tableau or Power BI have robust support for querying relational databases that contain tidy data. BI analysts can access the tidy data and conduct exploratory analysis or generate visualizations and dashboards that can be broadly consumed by stakeholders. BI admins can set up data extracts to occur on a scheduled basis, ensuring that newly processed data from the data science team is automatically available to BI users. Dashboards and workbooks that are tied to these data extracts can register updates automatically and take advantage of the latest BI features such as dynamic parameters.



Explore / Default / bike_pred_data (content.bike_pred_data)+ (rds)

bike_pred_data (content.bike_pred_data)+ (rds)

Owner Sean Data is Extract ▾ Mar 8, 2021, 9:03 PM

	Refresh type	Actions	↑ Schedule	Priority	Last update	Next update
□	Full refresh	···	Weekday early mornings – Every Mon, Tue, Wed, Thu, Sun, at 9:00 PM	50	Mar 8, 2021, 9:03 PM	Mar 9, 2021, 9:00 PM

Figure 2: A scheduled Tableau data extractions

The Pros and Cons of Data Handoffs

The above example demonstrates the flexibility of data handoffs. However, like any technology, data handoffs have their advantages and disadvantages as shown in the Table 1 below.

Pros	Cons

	Pros	**Cons**
Data Handoff Approach	• The data hand-off technique allows ready access from tools in both the data science and BI stack. R, Python, Tableau, and Power BI all support reading and writing to databases which means setup is easy and ongoing maintenance is limited. • Data handoffs cleanly isolate interactions between tools, allowing developers to collaborate quickly while still making it easy to troubleshoot errors.	• Data handoffs rely on batch schedules which aren't ideal for data workflows that require near real-time data updates. • The flow of data in this approach is uni-directional starting with the data science team and flowing into the BI tool. Bi-directional data flows would require additional techniques.

Table 1: Summary of some of the pros and cons of data handoffs.