Roosevelt simple. seamless. smart.

Team Roosevelt Innovations Data Science

Provider Anomaly Analytics Toolkit

Project Plan

Fall 2022

Roosevelt Innovations Sponsors

Jacob Ernst

Ayush Singh

Shikha Mohindra

Mukundan Agaram

Michigan State Capstone Team

Justin Doan Carson Honkala Jeffrey Lo Tanawan Premsri Kate Roney Junchi Zhu

Contents

Executive Summary	3
Functional Specifications	4
Design Specifications	5
Overview	5
User Interface – Navigation Menu and Frame	5
User Interface – Outlier Identification View	6
User Interface – Summary View	6
User Interface – Comparison View	8
Technical Specifications	10
Software Architecture	10
Infrastructure Technologies	10
Programming Libraries	11
Risk Analysis	12
Ensuring solution is modular	12
Identifying useful business insights from claims data	12
Schedule	13

Executive Summary

Roosevelt Innovations is a subsidiary of Delta Dental, a leading provider of dental insurance in the United States. While Roosevelt Innovations is solely owned by the Delta Dental of Michigan, they are a software platform provider that also caters to providing solutions to other Delta Dentals across the country. These solutions focus on developing software that allows insurance companies to streamline their operations into a single platform to optimize their business decision in a cost-effective manner. One of the main tasks required is being able to handle a variety of data sources and having the ability to delve into each source comprehensively.

To achieve this, Roosevelt Innovations has created a variety of tools and data visualizations that can view a provider's performance metric. These metrics consider the quality of care, overutilization, and over-treatment aspects that contribute to a risk score. This score aids in determining decisions such as whether an insurance provider's claim should be manually reviewed before its submission. However, these features have been implemented through multiple disparate systems that makes it difficult to use and comprehend. Roosevelt Innovations is looking to curate these systems into one simple repository. In this new system, they want to have access to a summary of these results while still retaining the ability to comprehensively analyze an individual data source.

Roosevelt Innovations has asked us for a solution that allows them to have aspects of their legacy products and to integrate them into web components. These components will be hosted on a web application that can display each provider's key performance indicators. Additionally, they will be developed in a way that will allow reusability of key web components in future software projects and platforms. Each component can be tailored to fit a certain analytical perspective. In doing so, insurance providers can look to monitor and make the same decision with increased clarity and considerably less effort.

Functional Specifications

Roosevelt Solutions provides software platforms for Delta Dental and other insurance companies. They allow these companies to monitor the activities of their dental providers. Business decision makers along with analysts can functionally use these tools, however, it becomes a problem when the amount of data starts to scale. The ability to identify potentially problematic dental providers becomes hard to comprehend and organize as investigators must manually visit multiple different archives to gather all the relevant data. This time-consuming process negatively impacts the efficiency of their workflow.

The overarching goal of our application is to replicate their current visualization systems into a more centralized location. Here, users can expect to have the same functional aspects with an increased ease of use. Our website will provide the necessary tools to guide analysts towards noticing problematic providers without manual instruction. Additionally, this will allow managers and executives to gain a high-level understanding of impactful key metrics related to each provider without requiring insight from their analysts. This will be accomplished by providing multiple views that summarize, compare, and highlight providers of interest from existing data sources. Upper management should expect to quickly identify potential problematic dental providers. From there, managers will be able to alert their analysts to review any questionable activities. This is important because it should allow for an efficient chain of workflow.

By implementing our solution, insurance providers such as Delta Dental can expect to see a variety of benefits especially with regards to their efficiency and costs. They should be able to identify questionable claims such as those who have elevated amounts of waste, on a more productive basis. This will allow companies to redistribute their budget into other departments. There should also be an improvement in the quality of care provided by being able to enact upon problematic providers sooner. These described features will help to highlight key business insights in a central location which will improve how efficient decisions can be made.

Design Specifications

Overview

The Provider Analytics Analysis Toolkit is a set of expandable WebUI components designed to help insurance providers, such as Delta Dental, save money through investigating dental providers making inefficient insurance claims. These WebUI components will be used through a web browser by three main "personas" that work for the insurance provider: technical analysts, middle management, and executives. To facilitate the use cases for each of these personas, there will be three views available for selection via a sidebar menu.

At a high level, the views can be described as follows:

- Outlier Identification View Creates a table that shows information about "high-risk" providers. In other words, this view automatically identifies providers who may be engaging in problematic behavior during the claims process. Problematic behavior generally refers to claims that result in additional expenses for the insurance company, such as frequently performing procedures that are unnecessary. This view will be used primarily by executives and middle management to quickly identify providers who may need to be investigated.
- Summary View Summarizes the claims activities of a specific dental provider through key
 performance indicators (KPIs). This view will be used primarily by middle management and
 analysts to get a high-level understanding of the behavior of a particular dental provider.
- Comparison View Uses data visualizations to compare a few different providers at a time using specified KPIs. This view will be used primarily by analysts who want to dive deeper into how specific KPIs for a provider compare to the entire dataset.

In addition to the three different views, the data is broken down into three different categories of indicators. Each view will allow the user to select a particular category of performance indicator to use. A basic description of these categories are:

- 1. Utilization Volume of procedures or types of treatments that are done in the office.
- 2. Unbundling Scenario Services that should be charged together but are broken down which results in additional fees
- Collective Care (Quality of Care) Unnecessary procedures that could have been prevented if a good standard of care was provided.

User Interface – Navigation Menu and Frame

Visually, the WebUI is built to be stylistically consistent with other tools created by Roosevelt Innovations. A dark blue sidebar on the left side of the screen controls navigation between each of the dental provider visualization pages. These navigation controls allow the user to select which of the three persona views they would like to use the WebUI as. This sidebar, the top bar that contains the Roosevelt logo, and the name of the project always remains on the screen which frames the rest of the content inside. This can be seen in *Figure 1* below.

Roosevelt Innovations Data Science - 5

Commented [SL1]: What does wasteful mean? Malicious?

Commented [RK2R1]: Changed.

User Interface – Outlier Identification View

This view will be primarily used by executives and middle management to quickly identify potentially problematic dental provider. In this view, the user will be able to filter the data with a variety of conditions, such as region, time, location, and the threshold of risk. Once these filters have been applied, the table below will automatically display information about providers that meet the specified risk threshold. This can be seen visually in *Figure 1* below.



Figure 1: A table of identified high-risk providers in the Outlier Identification View

User Interface – Summary View

This view may be used by all three personas, however, its primary target is middle management and technical analysts. As seen in *Figure 2* below, this view contains a search menu at the top of the page to help users find a particular provider using filter conditions such as location, name, and region. The user may fill in as many filters as they would like. If the user clicks search without any active filters the search results will contain every provider in the database.

Once the search button is selected every provider that fits the filter criteria will be displayed in the search result area. The select button then allows the user to choose which provider they would like to see the key performance metrics for.

Following the selection of the desired provider, a variety of KPI boxes will populate with information from the data. Each of these KPI boxes can be customized to contain any metric, which is done by using a dropdown menu at the top of the box. In the bottom right of each box there is also text that indicates how a particular provider ranks in a KPI for their region. This can be seen visually in *Figure 3* below.

OS€V e. seamless	/elt 5. smart.						
×							So assame.
	Derwiden Companien				Sector		
\sim	Provider Comparison						
<u>_</u>	Prevention Filter						
U	State		City.		Presiden	hovide ID:	
ome	bil.		Select/Dr		Salart Provider	falet (travider II)	
-			Destroy		peeu rome	Select of the lo	
TΔ	State	City		Provider		Provider ID	PEER_EBOUR
-		Loreing		Hertness-Smitham		1712596	5
		Loving		"Davis, Wessent and Mannerstill		1125040	8
parison	*	Loreing		"Jaskobki, Gaylond and O'Hara"		1125974	5
~	*	Battle Greek		Konze U.C.		112560	5
<u>-</u> 1		Detroit		"Nation, Kawiney and Partiel"		1138964	
-		First		"Culturedal, Pranecial and Maure"		1127875	4
		harmington		"Nators, Showan and Braker"		1101313	1
nmary	*	7109		"Merzy, Skiles and Cormier"		1130011	4
	-	P1-4		Marken and Cours			
E1	Select				Add To Wetch List		
Watch List	Hartmann-Smitham, Hi, Lansing				Hunze LLC, HI, Battle Creek		
	MPL Filter						,
	Data Calogory:				KPb		
	Claim Mainipulation				Select KPI		
	Auto seleci KPla				Clear KPIs		

Figure 2: Dental Provider search results are of the Provider Comparison View



Figure 3: KPIs being displayed in the Provider Summary View

User Interface – Comparison View

The comparison view is designed for middle management and analysts to comprehensively understand the differences between selected providers. This is done by providing data visualizations such as radar and bar charts to illustrate the trends over specific time periods.

Users can select up to five providers to compare at a given time using the search and selection menu tools at the top of the page. The search menu will function identically to the one described and shown previously in the *User Interface - Summary View* section and *Figure 2* above but will support multiple providers.

After the providers have been selected, the user can then select up to five KPI categories to be put on display. These categories can be searched and selected by using the dropdown menus immediately above the "Selected KPI" box. Once selected, visualizations below will be updated to include this information. This selection menu along with the accommodating visualizations can be seen in Figure 4 below.

For each of the plots in the comparison view, every provider will be assigned distinctive colors to distinguish the level of their KPIs from others. To efficiently represent multiple providers at a given time, we will be using a radar and bar chart. This radar visual will offer comparison of multiple KPI types while

the bar chart will offer the capability to compare a singular KPI category with its respective average. Users will be given the ability to display specific values from each bar by hovering over it. Both visualizations are featured in Figure 4 below.



Figure 4: Visualizations being displayed In the Provider Comparison View

Technical Specifications

Software Architecture

All functionality of the webapp will be implemented into one Python server using the Streamlit framework. The server will be hosted inside a docker container to allow for easy deployment of the application. All graphs and UI elements will be generated using this Python framework, data for the graphs will be fetched from snowflake at runtime using the pandas library and will dynamically update any necessary visuals. Potentially problematic dental providers will be identified using a machine learning model, which will be trained offline using the PyTorch and SciKit Learn libraries with data provided to us. The models will be trained using supervised learning models including neural network models, and linear models

". The model will deliver a risk score from each dental provider, which will then be stored back into the Snowflake database. A visual representation of this architecture can be seen in *Figure 5* below.



Figure 5: A diagram describing the software architecture of our website

Infrastructure Technologies **Python**

Python is a high-level programming language that consists of powerful libraries and high-level data structures for the development of both frontend and backend applications. Furthermore, Python is optimized for workloads with heavy computations such as machine learning and big data analysis. This

language is used as the main component of the project for web application development, data analysis, and machine learning.

Docker

Docker is open-source software developed for efficient building, running, and management of servers. This server will be created within an isolated environment known as a container. In this project, Docker will be used to host and deploy our Python server.

Snowflake

Snowflake is a cloud database that provides the ability to store and query data securely. It offers a powerful cluster for massive data computations. It supports connection from several programming languages such as Python and Node.js. It will serve as the main database for our web application.

Programming Libraries Streamlit

Python is a high-level programming language that consists of powerful libraries and high-level data structures for the development of both frontend and backend applications. Furthermore, Python is optimized for workloads with heavy computations such as machine learning and big data analysis. This language is used as the main component of the project for web application development, data analysis, and machine learning.

Pandas

Pandas is a Python library developed for the purpose of managing and analyzing data. It provides several APIs to do this and gives us the capability such as reading and writing from various types of sources. It will be used as the main library for retrieving and processing data from Snowflake.

Numpy

NumPy is a Python library used to process multidimensional data types. It contains various functions and mathematical methods to process data efficiently. This will be used as a tool for efficiently managing large backend data and establishing a well composed metadata.

Scikit-learn

Skikit-learn is a robust Python library used for machine learning. It provides multiple methods to implement simple machine learning models. This will be used to implement a machine learning approach for our project.

PyTorch

PyTorch is a deep learning framework used for the creation of machine learning models. It provides effective tools used for the implementation of complex machine learning models from scratch. Some approaches include various loss functions and optimizers. This will be used in the event that we require more complex machine learning tools.

Risk Analysis

Ensuring solution is modular

Description: The features built into our solution need to be reusable in other software contexts in the future. For example, data read-in should be agnostic to a particular source.

Mitigation: While planning the design of our features, we will compartmentalize aspects of our product with due diligence in order to ensure this reusability element.

Identifying useful business insights from claims data

Description: Each view can only display a limited number of key performance metrics on the screen at a time. There are hundreds of metrics to choose from, and the displayed metrics must be useful for the specified person to make a business decision

Mitigation: In our weekly meetings with sponsors, we will consult with them to discuss and confirm which metrics would help them the most in making business decisions, and our solution will be updated based on their requirements.

Finding an effective method to determine high risk providers

Description: It is difficult for humans to manually identify potentially problematic providers without knowledge of the key problematic metrics, especially with large datasets like the one provided to us.

Mitigation: We decided to implement both supervised and unsupervised machine learning models to automatically identify potentially problematic providers based on similarity to already known problematic dental providers.

Match website UI to existing Roosevelt products

Description: All existing Roosevelt websites are built with the Angular framework, and this is one of the first websites Roosevelt will use with Streamlit as their frontend technology. Therefore, there might be difficulties matching the appearance and feel of our website to existing Roosevelt websites, which may provide an unpleasant experience for older users.

Mitigation: Research what's possible with Streamlit and consult with sponsors based on this information. We will then come to an agreement on how to best match our UI to the existing Roosevelt platform.

Schedule

Week 1 (8/29 - 9/4):

- Initial meeting with team
- Contact project sponsor
- Install IDEs and expected packages/languages

Week 2 (9/5 - 9/11):

- First meeting with Roosevelt Team
- First triage meeting
- Research necessary technologies

Week 3 (9/12 - 9/18):

- Status report presentation
- Create rough draft of project plan
- Create dashboard prototype
- Learn necessary technologies

Week 4 (9/19 – 9/25):

- Finish project plan document
- Project plan presentation
- Finish restructuring project
- Implement base pages for each view
- Create filter box component

Week 5 (9/26 - 10/2):

- Initial machine learning model for classifying and clustering
- Set up Python Docker container
- Create empty pages and buttons to front-end
- Implement sidebar functionality

Week 6 (10/2 - 10/8):

- Evaluation and analysis of machine learning models
 - Tuning parameters of machine learning models
- Finish implementing the bar chart with average line
- Implement summary view boxes
- Work on alpha presentation

Week 7 (10/9 – 10/15):

- Alpha presentation
- Finish implementing outlier view
- Implement comparison of two or more providers with bar graph
- Start working on Shopping Cart view

Roosevelt Innovations Data Science - 13

Commented [SL3]: Update schedule

Commented [RK4R3]: Updated.

• Tune machine learning models for more accurate results

Week 8 (10/16 - 10/22):

- Finish implementing the radar chart
- Updating front-end look with Roosevelt images and resources

Week 9 (10/23 - 10/29):

- Fall Break (buffer week for any behind tasks)
- Add csv export to Watch List

Week 10 (10/30 - 11/5):

- Populate the outlier table with data from machine learning model
- Consult with sponsor to fine tune website or add functionality
- Add pdf export to Watch List

Week 11 (11/6 - 11/12):

- Work on beta presentation
- Add/change functionality from discussion on Week 10
- Start planning video presentations
- Add descriptions for graphs

Week 12 (11/13 - 11/19):

- Beta Presentation
- Integration with Snowflake
- Optimize code

Week 13 (11/20 - 11/26):

- Project Video filming
- Styling changes
- Clean up code and refactoring

Week 14 (11/27 – 12/3):

- Start design day presentation
- Finish up Video filming
- Documentation
- Finalize deliverables

Week 15 (12/4 - 12/10):

- Design Day presentations and Project Videos
- All Deliverables