

# Teaching Digital Health

with KNIME Analytics Platform



# Table of Contents

Introduction.....	3
Hackathon Challenges.....	4
Harness data to forecast disease outbreaks and protect communities .....	4
Revolutionizing medical diagnostics through AI-driven image analysis for timely and precise patient care.....	4
Just KNIME It! Challenges.....	5
CDC Cancer Data .....	5
Obtaining a List of Human Genes via REST .....	5
Patient Network Days.....	6
Rare Blood Types .....	7
Medical Procedure Prices .....	7
Dealing with Diabetes.....	8
Detecting the Presence of Heart Disease.....	8

# Introduction

We have compiled this document to provide some inspiration and starters for how KNIME Analytics Platform can support teaching digital health subjects. If you are interested in a detailed account of how such an integration could look like for a PharmD program: Prof. Dayanjan S. Wijesinghe has published an [overview article](#) of his approach to integrating digital health into the curriculum at Virginia Commonwealth University (VCU) School of Pharmacy. Furthermore, researchers at the University of Southern California and the University of the Pacific have published<sup>1</sup> their experience with integrating Machine Learning through KNIME into the PharmD curriculum.

With this package, you have received **presentations covering various use cases** that you can use in your digital health teaching, in addition to course materials available through the [KNIME Educators Alliance](#):

- *2022-11-29 KNIME Calculators for Personalized Pharmacotherapy.pptx*
  - KNIME in the PharmD program at VCU School of Pharmacy
  - Vancomycin Dosing Calculator
  - [Recording](#)
- *2023-10-25 Getting Your Research Right with Open-Source Visual Programming.pptx*
  - Drug Discovery
  - Literature Research & Topic Modeling
  - [Recording](#)
- *2024-03-06 Heparin Therapeutic Monitoring and Calculation with KNIME.pptx*
  - Heparin Dosing Calculator
  - [Recording](#)
- *2024-03-06 Identification of Adverse Drug Events from FAERS with KNIME.pptx*
  - Working with FAERS in KNIME Analytics Platform

The remainder of this document details **hands-on projects and challenges** that can be run in classes to make the learning more interactive. The first two projects have been proposed at the [Future Health Pioneers](#) hackathon, jointly organized by KNIME and [Virginia Commonwealth University \(VCU\) School of Pharmacy](#) in June 2023. These projects have a bigger scope and are more abstract so you can also assign them as term-long projects. Broad topic areas suggested at the Future Health Pioneers hackathon:

- Prediction of Disease Outbreaks
- Personalized Medicine and Pharmacovigilance
- Medical Image Analysis
- Promoting Diversity and Equity in Healthcare
- Remote Patient Monitoring and Telemedicine

The remaining challenges were originally published as [Just KNIME It!](#) challenges, for which participants in the format had one week to tackle them. This makes them great choices for lab exercises to enhance your learners' KNIME skills with data that they can easily relate to. The bigger picture is often secondary for these challenges since they focus more on technical upskilling. You can find more challenges as well as detailed descriptions of the solutions in the Just KNIME It! [Season 1](#) and [Season 2](#) booklets.

---

<sup>1</sup> <https://doi.org/10.1016/j.ajpe.2024.100696>

# Hackathon Challenges

## Harness data to forecast disease outbreaks and protect communities

### Objective:

- Develop a real-time predictive model/app for disease outbreaks using healthcare data, environmental factors, and social media trends.

### Outcome:

- A predictive tool with real-world applications for healthcare officials and the public. Demonstrate data usage, analysis methods, and implementation potential.

### Putative Data Sources and Ideas:

- *Healthcare Data*: Historical outbreaks, patient records
- *Environmental Data*: Weather, temperature, humidity
- *Social Media Trends*: Data mining for symptom trends
- *Mobility Data*: Human movement patterns

## Revolutionizing medical diagnostics through AI-driven image analysis for timely and precise patient care

### Objective:

- Develop an innovative AI-powered application that accurately analyzes and interprets medical images (such as X-rays, MRIs, CT scans) to assist in the diagnosis and monitoring of diseases.

### Outcome:

- A workflow capable of analyzing medical images with high precision, providing diagnostic suggestions, and integrating patient data for a holistic view.

### Putative Data Sources and Ideas:

- *Medical Imaging Datasets*: Collections of X-rays, MRIs, CT scans, and other imaging modalities
- *Annotated Images*: Images annotated with diagnoses or features, for training machine learning models
- *Patient Data*: Demographics, medical history, lab results for contextual analysis
- *Public Imaging Repositories*: e.g., TCIA, Open-i, MIMIC-CXR for additional data

# Just KNIME It! Challenges

## CDC Cancer Data

**Level:** Easy

**Description:** You received the 2017 cancer data from the CDC for inspection, and your goal is to answer the following questions: (1) What are the top-5 most frequent cancer types occurring in females? (2) What are the top-5 most frequent cancer types occurring in males? (3) Which US state has the highest cancer incidence rate (that is, the highest number of cancer cases normalized by the size of its population)?

**Dataset:** [Cancer and Population Data in the KNIME Hub](#)

**Solution Summary:** To find the top-5 most frequent cancer types occurring in (fe)males in the US in 2017, we preprocessed the data to remove aggregated cancer sites that could lead to wrong counts, grouped the data by cancer type, and pivoted by sex. Next, we sorted the data to find the top-5 most frequent cancer types. As for the highest normalized incidence of cancer, we grouped the CDC data by state, read the states population data, and then joined these two datasets on state. We then normalized the number of cancer cases per state by the corresponding population, and sorted the resulting data to find the state with the largest incidence.

[See our solution in the KNIME Hub](#)

## Obtaining a List of Human Genes via REST

**Level:** Medium

**Description:** You have been working for a Life Sciences company for a month as a data wrangler. Several coworkers from the Biology department would like to obtain a list of human genes related to specific hormones, but they do not know how to use REST services, GET requests, etc. Your task is to use the REST service provided by [MyGene.info](#) to obtain a list of human genes related to a list of hormones provided to you by your coworkers. Next, you should parse the JSON response into a table that is easy to read.

For example, if you use "<http://mygene.info/v3/query?q=summary:>" and append "insulin", then your request would return a JSON structure with 10 hits -- each one of them with the following fields: "\_id", "\_score", "entrezgene", "name", "symbol", and "taxid".

You should then parse this JSON into a table with columns "\_id", "\_score", "entrezgene", "name", "symbol", and "taxid". If the list provided by your coworkers contains more than one hormone, all the parsed information should be aggregated into a single table. Also, sometimes your request may return a response in XML instead of JSON. How could you include a way to also parse XML responses?

Need a tip or two? See our [youtube video on REST API](#).

**Dataset:** [Example of a List of Hormones in the KNIME Hub](#)

**Solution Summary:** To tackle this challenge, we started by reading a list of hormones and then, for each hormone, we formatted and executed a GET request. The user should then inspect the result,

determine whether the responses were in XML or JSON, and then execute a component we created to control the execution of the rest of the workflow. We then implemented solutions to parse responses of both types (XML and JSON).

[See our Solution in the KNIME Hub](#)

## Patient Network Days

**Level:** Hard

**Description:** You work for a hospital and they have data for each time a patient was seen. In this challenge, you will calculate the difference between each time a patient was seen excluding weekends (called "network days"). Once you calculate the network days, calculate the average network days per patient. For the challenge, experiment with the input and output below.

### Input

Patient	Date
Aline	11/01/2022
Aline	12/02/2022
Aline	25/02/2022
Aline	15/04/2022
Victor	05/02/2022
Victor	25/02/2022
Victor	15/03/2022
Victor	30/03/2022

### Output

Patient	Date	Network Days	Mean
Aline	11/01/2022	?	23.333
Aline	12/02/2022	24	23.333
Aline	25/02/2022	10	23.333
Aline	15/04/2022	36	23.333
Victor	05/02/2022	?	13.333
Victor	25/02/2022	15	13.333
Victor	15/03/2022	13	13.333
Victor	30/03/2022	12	13.333

**Note:** if you simply use the Date&Time Difference node, you will mix patient data/dates and will also end up counting weekends. Bonus Challenge: Create a solution without using loops.

**Solution Summary:** First, we changed our string to datetime format so that we can calculate temporal differences. To avoid loops, we then used lag columns to tag when our patients change. Afterwards, we used another lag column to calculate the differences between days. Next, we used a component which we found in the KNIME Forum that allowed us to calculate network days using math formulas. Finally, we calculated averages using a groupby and then joined that with our data.

[See our Solution in the KNIME Hub](#)

## Rare Blood Types

**Level:** Easy

**Description:** You have a dataset containing information on US citizens who donated blood in the last year, including addresses and blood types. The O- blood type, also known as "universal donor", is perhaps the most valuable blood in the world because it can be transfused to nearly any blood type holder. Your goal here is to help a group of researchers find the number of citizens with O- blood type per US state. Unfortunately, the address column comes in a single line, so to extract the state information you will have to perform some data wrangling. They also asked you to create a choropleth map of the US to visualize the results.

**Dataset:** [Synthetic Blood Type Data in the KNIME Hub](#)

**Solution Summary:** After reading the dataset in rare blood types, we first extracted the state from the address line. We then removed the unnecessary address column and renamed the name and the state column for better readability. We then filtered the data to only keep samples where the blood type equals "O-" and grouped the data by state. Lastly, to visualize the data in a world map we used the *Choropleth Map* component.

[See our Solution in the KNIME Hub](#)

## Medical Procedure Prices

**Level:** Hard

**Description:** In America the prices of medical procedures can vary greatly, so savvy Americans tend to shop around for a good deal. In this challenge you will take the role of a data journalist trying to investigate price differences among medical procedures. More specifically, you want to find the top 5 procedures that show the largest variety in terms of pricing from hospital to hospital (in statistics, you would call this high standard deviation). The data you have at hand for this investigation is not uniform and requires inspection in order to be properly read and processed. For simplicity, compare all average charges for the 25 most common outpatient procedures performed by hospitals from the *Kaiser Foundation Hospitals* and *Sutter Hospitals* only. For *Kaiser Foundation*, the relevant data is in Excel files with "...Common25..." in their names; for *Sutter*, the relevant Excel files contain "...CDM\_All..." in their names. Beware of hidden sheets.

**Data:** [Link for Medical Procedures Dataset](#)

**Solution Summary:** We started our solution by filtering the spreadsheets in order to just consider those that concern Kaiser Foundation and Sutter hospitals. After extracting hospital names, we read their corresponding Excel sheets and processed the codes and prices of their medical procedures. Finally, we computed a few statistics for the price distribution of each procedure and picked the top-5 ones with the largest standard deviation. They were *laparoscopic cholecystectomy; excision, breast lesion; hernia repair, inguinal, 5 years and older; carpal tunnel surgery; and arthroscopy, knee, with meniscectomy (medial or lateral)*.

[See our Solution in the KNIME Hub](#)

## Dealing with Diabetes

**Level:** Easy or Medium

**Description:** In this challenge you will take the role of a clinician and check if machine learning can help you predict diabetes. You should create a solution that beats a baseline accuracy of 65%, and also works very well for both classes (having diabetes vs not having diabetes). We got an accuracy of 77% with a minimal workflow. If you'd like to take this challenge from easy to medium, try implementing:

- [sampling techniques](#)
- [feature importance calculation](#)

**Dataset:** [Diabetes Data in the KNIME Hub](#)

**Solution Summary:** Our minimal workflow that beats the baseline accuracy trains a Random Forest classifier with 70% of the original, annotated dataset, and assesses its quality with the remaining 30% of the data. The data is split using stratified sampling due to its heavy class imbalance.

[See our Solution in the KNIME Hub](#)

## Detecting the Presence of Heart Disease

**Level:** Medium

**Description:** You work as a data scientist for a healthcare company attempting to create a predictor for the presence of heart disease in patients. Currently, you are experimenting with 11 different features (potential heart disease indicators) and the XGBoost classification model, and you noticed that its performance can change quite a bit depending on how it is tuned. In this challenge, you will implement hyperparameter tuning to find the best values for XGBoost's *Number of Boosting Rounds*, *Max Tree Depth*, and *learning rate* hyperparameters. Use metric F-Measure as the objective function for tuning.

**Dataset:** [Heart Disease Data in the KNIME Hub](#)

**Solution Summary:** To solve this challenge, we create a workflow segment with integration deployment nodes that is responsible for training and assessing the quality of a XGBoost classifier. This workflow segment is given as input to a component that performs hyperparameter optimization, along with information on what should be optimized. The best F-measure values (0.87 on average) are obtained using *Number of Boosting Rounds* = 50, *Max Tree Depth* = 6, *learning rate* = 0.1.

[See our Solution in the KNIME Hub](#)



KNIME AG  
Talacker 50  
8001 Zurich, Switzerland

[www.knime.com](https://www.knime.com)  
[info@knime.com](mailto:info@knime.com)

The KNIME® trademark and logo and OPEN FOR INNOVATION® trademark are used by KNIME AG under license from KNIME GmbH, and are registered in the United States. KNIME® is also registered in Germany