data circles
women diversifying data science

Project Kickoff

July 2020

Projects Circle

Mission

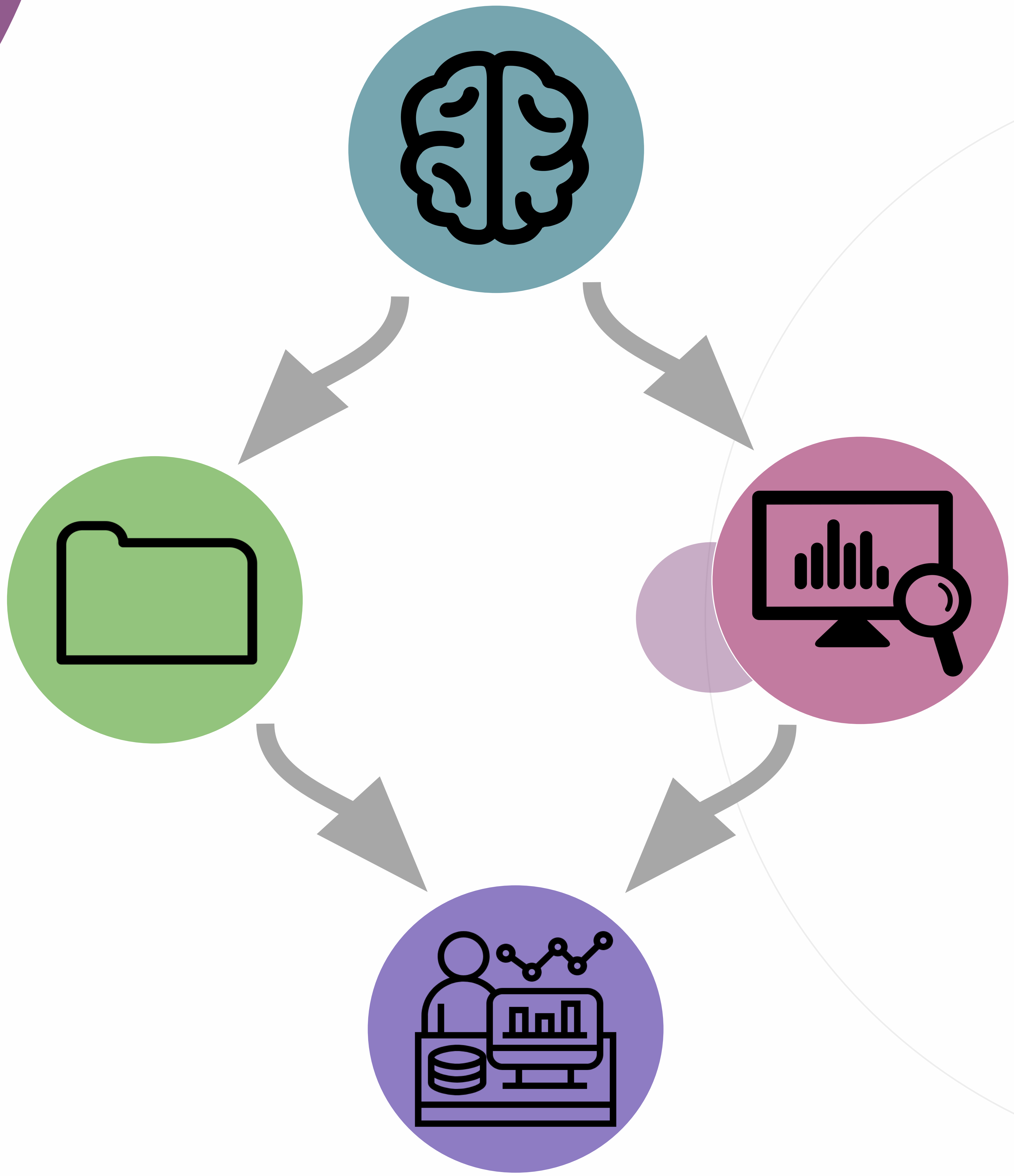
One of the difficulties for newcomers to the field is getting experience with creating end to end projects in a collaborative environment. The goal of this circle is to provide participants with **“real world” experience developing a data science project from conceptualization to execution.** We will have mentors and partners to guide assistants through the whole process and ensure that they get the necessary tools to develop data science projects independently.

GitHub Repository

https://github.com/DataCircles/projects_circle

Projects Circle Leadership Team Members

Erika Pelaez, Erin Orbits, Houda Aynaou, Niwako Sugimura, and Sowmya Vasan



data circles

Traffic Collision Data Project

Project introduction

Duration: 4 - 6 weeks

Time commitment: suggest 8 - 10 hours per week (collective)

Current data:

- Traffic collision data from the Seattle Transportation Dept. (SDOT) (csv data)
- Seattle street data from the SDOT (csv data)

Proposed project goals:

1. Identify dangerous locations
2. Identify predictors of traffic collisions, *e.g.*, physical characteristics of the location, road condition, DUI, weather
3. Examine increase or decrease in number of collisions over time
4. Identify predictors of increases or decreases in the rate of collisions
5. Recommend improvements to dangerous locations
6. Plus -- any questions of interest to you!

Recapping the collision data

The SDOT traffic collision data has 40 columns and 200,000+ incidents from 2004-present. The most relevant columns include:

Numeric count variables:

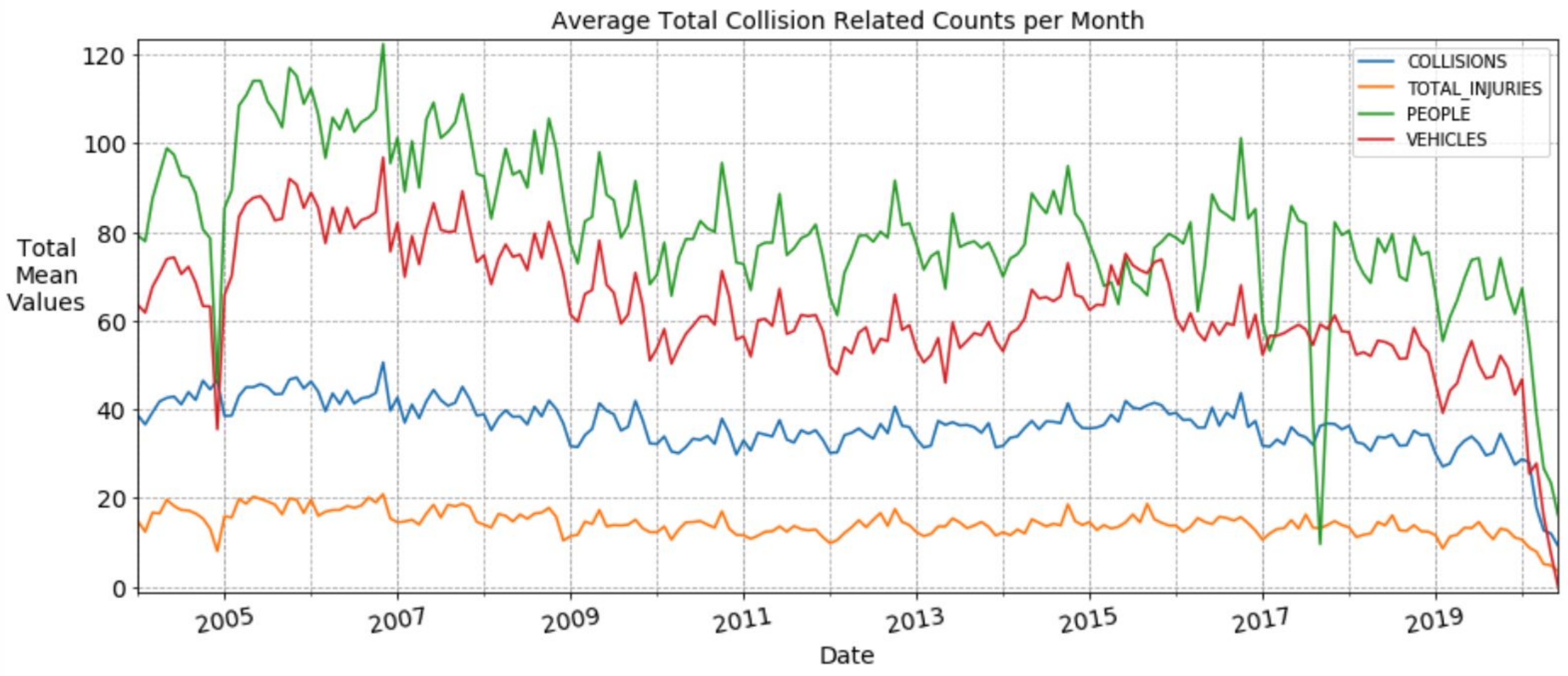
- PERSONCOUNT (# of people involved)
- PEDCOUNT (# of pedestrians)
- PEDCYLCOUNT (# of cyclists)
- VEHCOUNT (# of vehicles)
- INJURIES (# of injuries)
- SERIOUSINJURIES (# of serious injuries)
- FATALITIES (# of deaths)

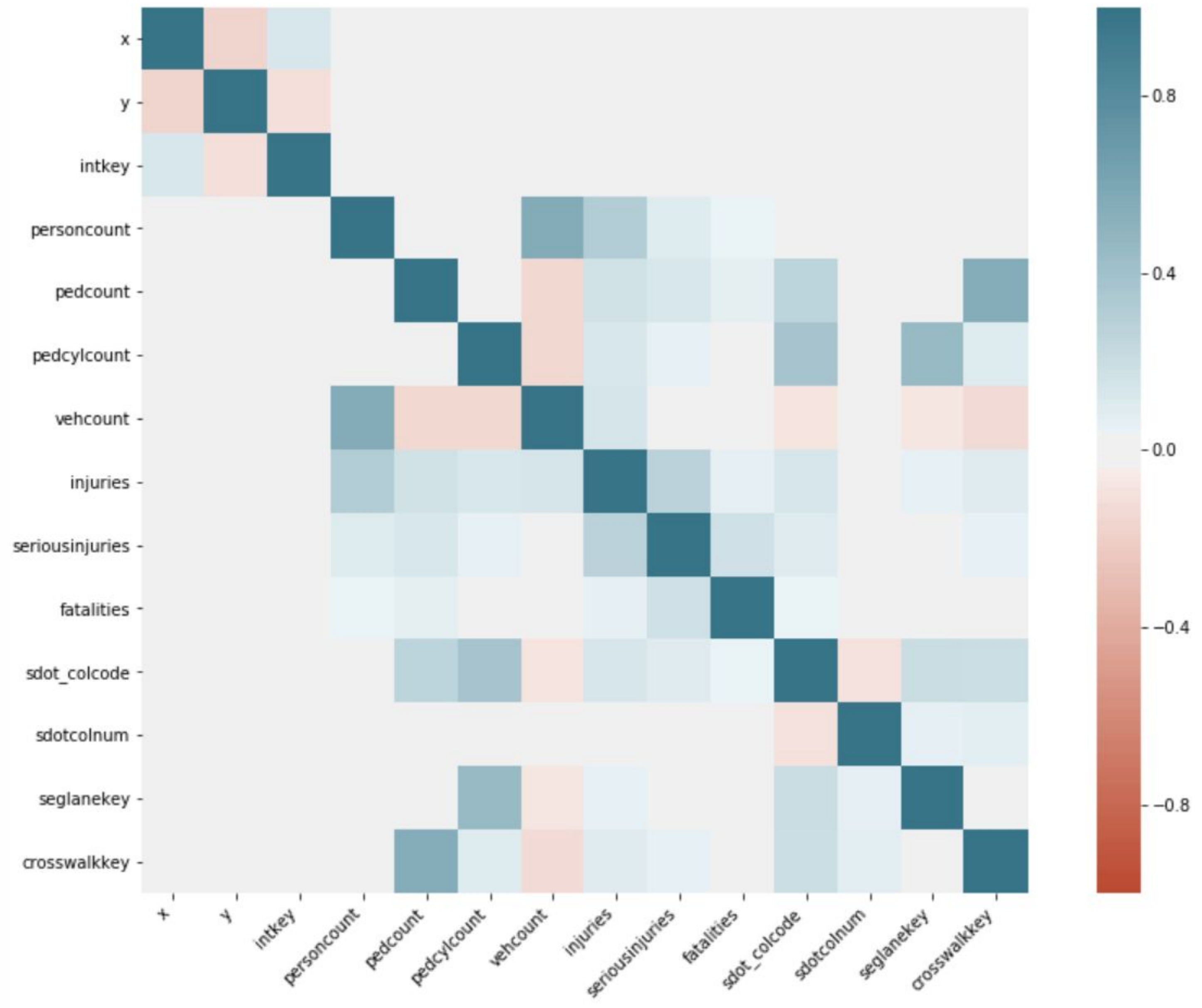
Human factors:

- INATTENTIONIND (whether driver was inattentive)
- UNDERINFL (whether driver was under the influence)
- PEDROWNOTGRNT (whether pedestrian had right of way)
- SPEEDING (whether speeding was a factor in the collision)
- ST_COLCODE (collision type label)

Characteristics of the site and conditions:

- ADDRTYPE (address type)
- LOCATION (text description with street names)
- X & Y (GPS location)
- CROSSWALKKEY (crosswalk label)
- JUNCTIONTYPE (roadway junction type)
- ROADCOND (road condition)
- LIGHTCOND (light condition)
- WEATHER (text description of the weather conditions)





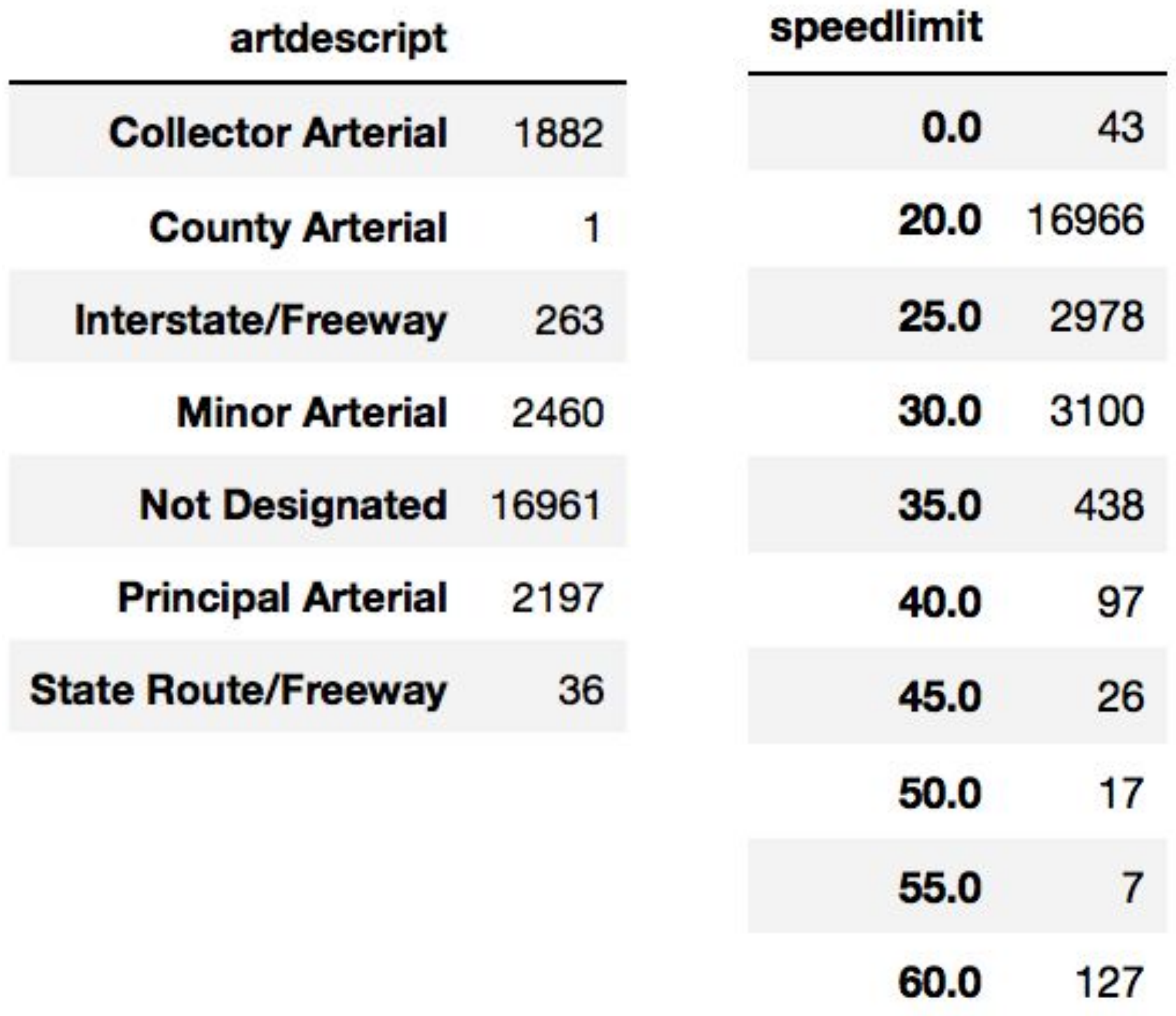
Recapping the street data

The Seattle street data has 38 columns and almost 24,000 rows describing the locations and attributes of streets within the Seattle city limits.

The variables include:

- Arterial Classification
- Street Names
- Block Number
- Direction
- One-way
- Surface Width
- Surface Type
- Pavement Condition
- Speed Limit
- Percent Slope

Example Summary Tables: road segment data grouped by the type of arterial and the speed limit



artdescript	
Collector Arterial	1882
County Arterial	1
Interstate/Freeway	263
Minor Arterial	2460
Not Designated	16961
Principal Arterial	2197
State Route/Freeway	36

speedlimit	
0.0	43
20.0	16966
25.0	2978
30.0	3100
35.0	438
40.0	97
45.0	26
50.0	17
55.0	7
60.0	127

Fun fact

Did you know...

Traf-O-Data was a business partnership between Bill Gates, Paul Allen and Paul Gilbert that existed in the 1970s.

The objective was to read the raw data from roadway traffic counters and create reports for traffic engineers. This software used CP/M as an operating system.

The company had only modest success but the experience was instrumental in the creation of Microsoft Corporation a few years later.

[from Wikipedia]

Project Goals

Two proposed tracks

1. Visualization Emphasis Track:

   Input: geolocation data on collisions, road conditions, weather conditions.

   Output 1: Generate maps with geopandas, folium, pandas.

   Output 2: Generate dashboard with Tableau, Google data studio

2. ML Emphasis Track:

   Input: geolocation data on collisions, road conditions, weather conditions

   Output 1. Classify and label collisions based on degree of danger, decide how to classify what constitutes a dangerous or not dangerous intersection

   Output 2. Predict the danger level of each intersection

   Output 3. Recommend changes to improve dangerous intersections based on similarity metrics

Other Track Suggestions? We want to focus the projects on your areas of interest, so please feel free to suggest areas of focus or final deliverables.

Challenges we foresee

Data quality: Some of the concerns for this data include the data collection technique (see the metadata page), understanding all the column names, and sanity check gotchas.

Lack of data: You will likely have to look for external sources of data. Although there are several relevant datasets in https://data.seattle.gov, the quality and completeness of the data varies.

Technical expertise: Data engineering (merging datasets, cleaning, reformating, & standardizing); GIS analysis; ML techniques; visualization tools (Tableau, Google Data Studio)

What is at the end of the tunnel?

Presentation with demo
At the end of the 6 weeks we will have a project fair where each team will have time to do a presentation of the project and findings they did. For the people who complete a data product they can also showcase their app with a demo.

Think bigger!
For the ones who are courageous, they can submit their work for a Data Science conference, there are different modalities but paper, posters and short talks are a good way to showcase your work. You can also write a blogpost and become a DS celebrity… The sky is the limit!

Mechanics of the Meetups

Weekly standups:

- Lessons of the week -- Share something interesting you learned with all teams!
- Status of your project -- What you were able to achieve in the week.
- Blockers -- How can we help you to advance your project
- Planning goals for the next week

Collaboration tools:

- Github
- Project Management Software: Trello, Asana, Jira…
- Slack channels #projects

Ad Hoc mini-workshops:

Possible topics: Agile, Github, geopandas, Tableau…
We want to be flexible and supportive, so let us know what is most useful.

Mechanics of Project Teamwork

Project Workflow (the logistics of work)
Data Science projects involve thousands of small decisions, tasks, goals, milestones, and deliverables. When you work by yourself, you know what decisions you've made, which tasks are done, and how you measure success. When you work on a team, you need to have a common way of doing things (aka standards), keep everyone in the loop, and everyone needs to know what work quality is expected.

Therefore, it's essential for the team to talk about:

(1) what tasks need to be done;
(2) how the work will get done;
(3) who will do the work; and
(4) when the work will be complete.

This administrative stuff is easy to put off, but if it's not done in the beginning, you will have an unhappy team.

Project Structure (the delivered work)
There are many examples of how to structure data science projects. Here's a blog article that outlines the general project folder structure for a data science project with the author's GitHub project template.
An example of a machine learning project structure is included in a later slide.
Here is an example of a data visualization project template on GitHub.

Example Project Workflows

Example high-level structures for organizing data science projects

Team Data Science Process (TDSP) structure for the development of data science projects describes four stages that projects typically execute, often iteratively [from Microsoft]:

1. Business Understanding
2. Data Acquisition and Understanding
3. Modeling
4. Deployment

Knowledge discovery in databases (KDD) process is commonly defined with the stages [from Wikipedia]:

1. Selection
2. Pre-processing
3. Transformation
4. Data mining
5. Interpretation/evaluation

Cross-industry standard process for data mining (CRISP-DM) is similar, but defines six phases [from Wikipedia]:

1. Business understanding
2. Data understanding
3. Data preparation
4. Modeling
5. Evaluation
6. Deployment

Example Machine Learning Data Science Project Template

```
├── .gitignore               <- Files that should be ignored by git. Add seperate .gitignore files in sub folders if needed
├── conda env.yml            <- Conda environment definition for ensuring consistent setup across environments
├── LICENSE
├── README.md                <- The top-level README for developers using this project.
├── requirements.txt         <- Requirements file for reproducing the analysis environment, e.g. generated with `pip freeze > requirements.txt`
├── setup.py                 <- Metadata about your project for easy distribution.
│
├── data
│   ├── processed            <- The final, canonical data sets for modeling.
│   ├── raw                  <- The original, immutable data dump.
│   ├── temp                 <- Temporary files.
│   └── training             <- Files relating to the training process
│
├── docs                     <- Documentation
│   ├── processdocumentation.md         <- Standard template for documenting process and decisions.
│   └── writeup              <- Sphinx project for project writeup including auto generated API.
│      ├── conf.py           <- Sphinx configuration file.
│      ├── make.bat          <- For generating documentation (Windows)
│      └── Makefikle         <- For generating documentation (make)
│
├── notebooks                <- Notebooks for analysis and testing
│   ├── eda                  <- Notebooks for EDA
│   │   └── example.ipynb    <- Example python notebook
│   ├── modelling            <- Notebooks for modelling
│   └── preprocessing        <- Notebooks for Preprocessing
│
├── scripts                  <- Standalone scripts
│   ├── deploy               <- MLOps scripts for deployment (WIP)
│   │   └── score.py         <- Scoring script
│   ├── train                <- MLOps scripts for training
│   │   ├── submit-train.py  <- Script for submitting a training run to your chosen service
│   │   └── train.py         <- Example training script using the iris dataset
│   ├── example.py           <- Example script
│
├── src                      <- Code for use in this project.
│   └── examplepackage       <- Example python package - place shared code in such a package
│       ├──   init   .py     <- Python package initialisation
│       ├── examplemodule.py <- Example module with functions and naming / commenting best practices
│       ├── features.py      <- Feature engineering functionality
│       ├── io.py            <- IO functionality
│       └── pipeline.py      <- Pipeline functionality
│
└── tests                    <- Test cases (named after module)
    ├── test notebook.py     <- Example testing that Jupyter notebooks run without errors
    ├── examplepackage       <- examplepackage tests
        ├── examplemodule    <- examplemodule tests (1 file per method tested)
        ├── features         <- features tests
```

Initial Project Tasks & Milestones

Week 0 (aka Project Kickoff)

Tasks

1. Introduce yourselves to your teammates
2. Document the business problem(s) and scope of the project
3. Decide team logistics:
    a. how to communicate, e.g. email, Slack
    b. how to track team progress, e.g. Trello, GitHub Project Board, Google Doc
    c. how to work collaboratively, e.g. GitHub repository, Google Drive
    d. how to divide up the work, e.g. assign team roles, work independently then compare

Milestone

1. Deliver the project charter to document the business problem and scope of the project

Week 1

Tasks

1. Produce a Jupyter notebook with a data report to document:
    a. the structure and statistics of the raw data
    b. how the data can help answer the business problem(s)
    c. what additional data is needed to answer the business problem(s)
2. Locate the additional data

Milestones

1. Deliver the data report
2. Deliver progress report at standup meeting

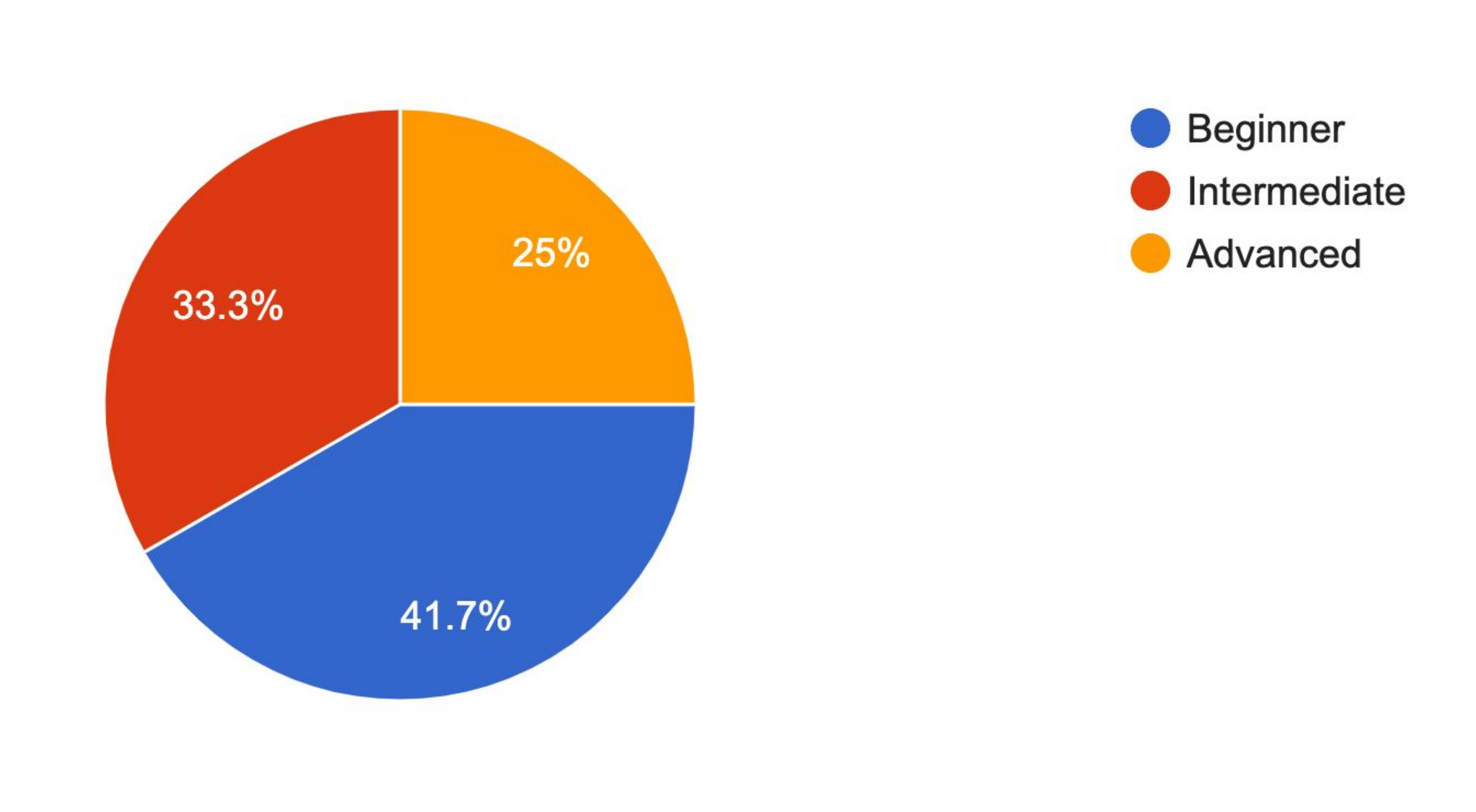
data circles

Questions?

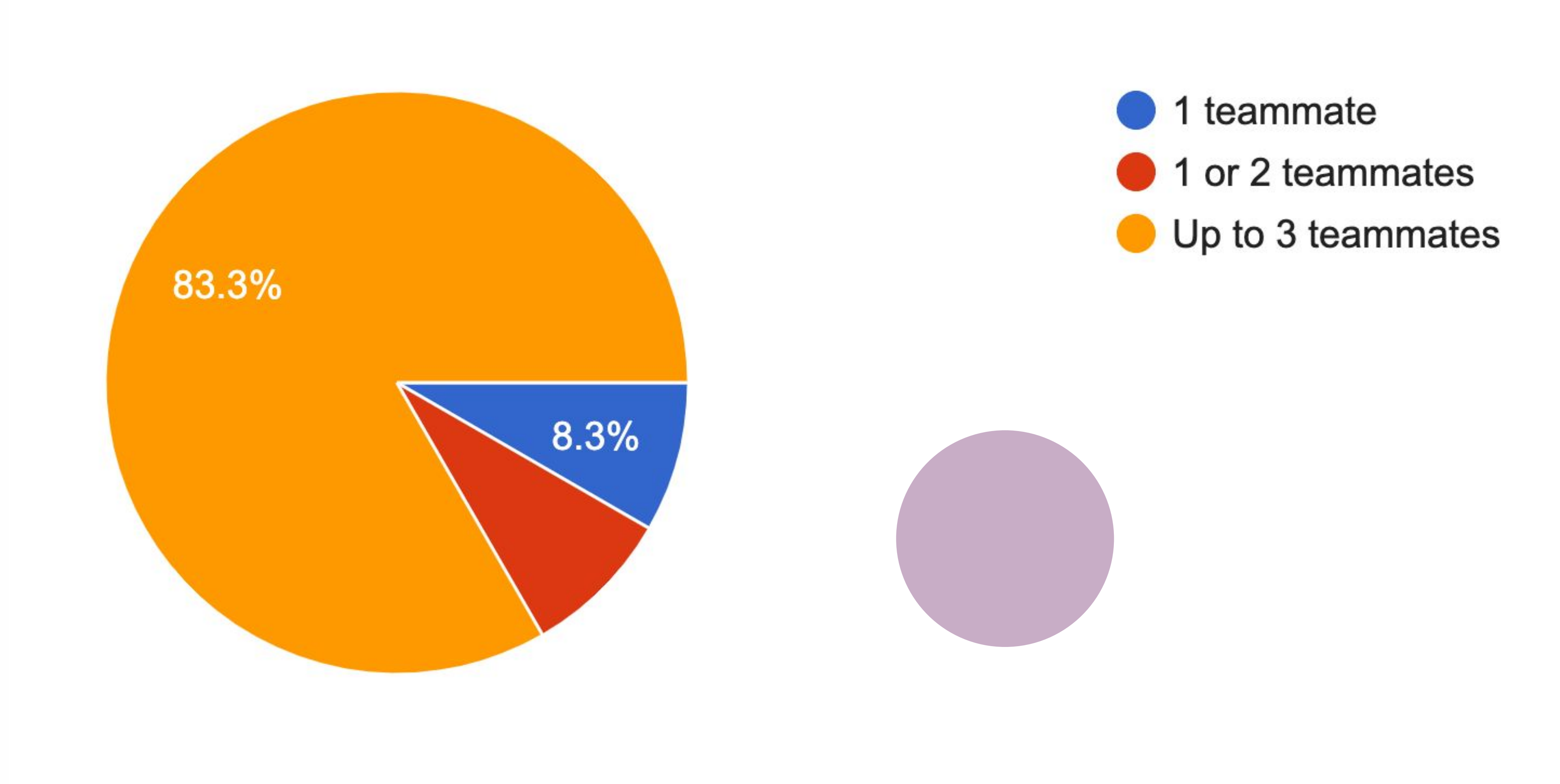
If not we can move on to team formation...

Team formation

Level of experience

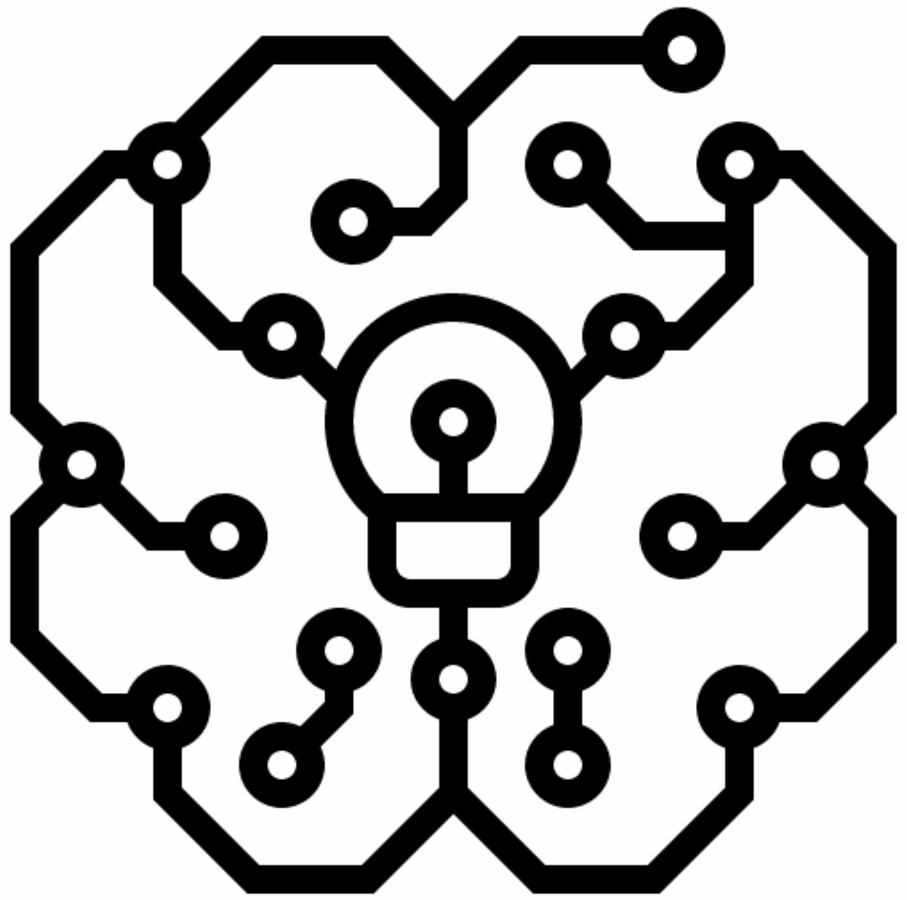


Team size



Round of introductions

Quick poll on tracks



data circles

Let’s form teams!

THANKS

THANKS