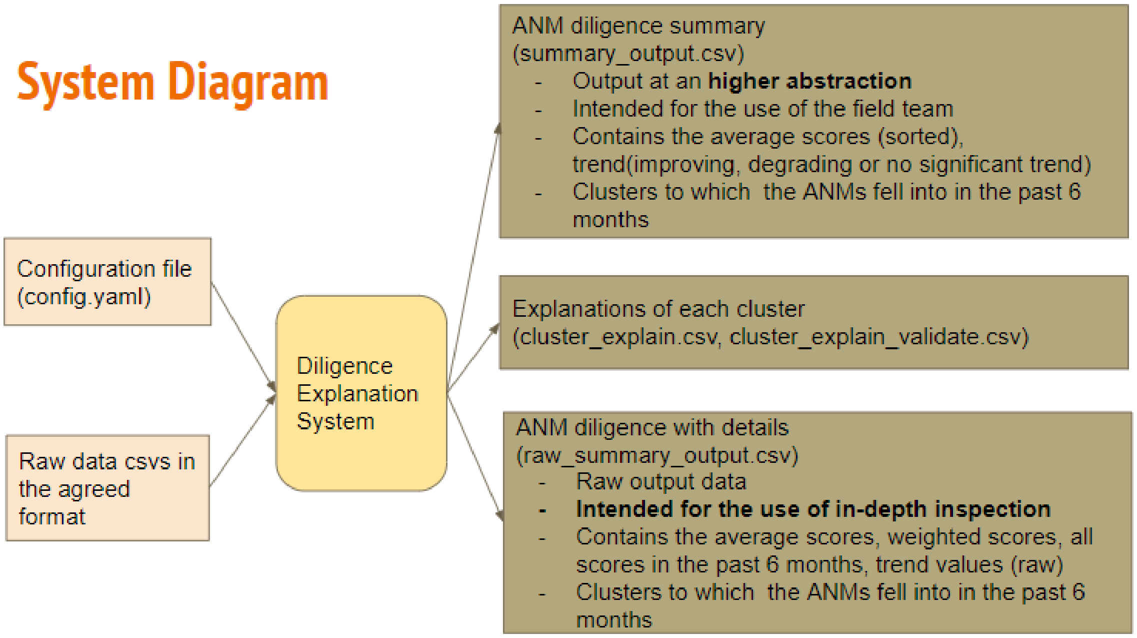
Table of Contents

Measuring and Identifying Non-diligence



System Diagram

Configuration file (config.yaml)

Raw data csvs in the agreed format

Diligence Explanation System

ANM diligence summary (summary_output.csv)

- Output at an **higher abstraction**
- Intended for the use of the field team
- Contains the average scores (sorted), trend(improving, degrading or no significant trend)
- Clusters to which the ANMs fell into in the past 6 months

Explanations of each cluster (cluster_explain.csv, cluster_explain_validate.csv)

ANM diligence with details (raw_summary_output.csv)

- Raw output data
- **Intended for the use of in-depth inspection**
- Contains the average scores, weighted scores, all scores in the past 6 months, trend values (raw)
- Clusters to which the ANMs fell into in the past 6 months

We first describe the ideas in our method and point to the code.

Filtering out ANMs who add noise to the distribution:

We ignore ANMs with less data on KB, such ANM’s data has high variance and they anyways do not participate in the KB system. Keeping the data of these ANMs can mislead the estimation of the distribution and hinder the measurement of diligence.

We filter out the ANMs according to 2 thresholds (both can be configured using the configuration file):

- ANMs who recorded patients below a given threshold in each month
- ANMs who recorded a total number of patients below number of months*given threshold

Only the short term rules will be considered for filtering ANMs.

How to use domain expertise knowledge as the foundation of the non-diligence measuring system?

Our first main idea is to get a probability of non-diligence for each rule, given the data for each rule from an ANM.

We use the 11 rules given by Khushi Baby (as of 15/02/2021).

These rules are of two types:

(1) rules that track various percentages related to health records in a given month

(2) rules that track data discrepancies/contradictions between health records

All rules specify a percentage of something, and we know which extreme (0% or 100%) or a given range corresponds to non-diligence.

As a running example we will use two rules:

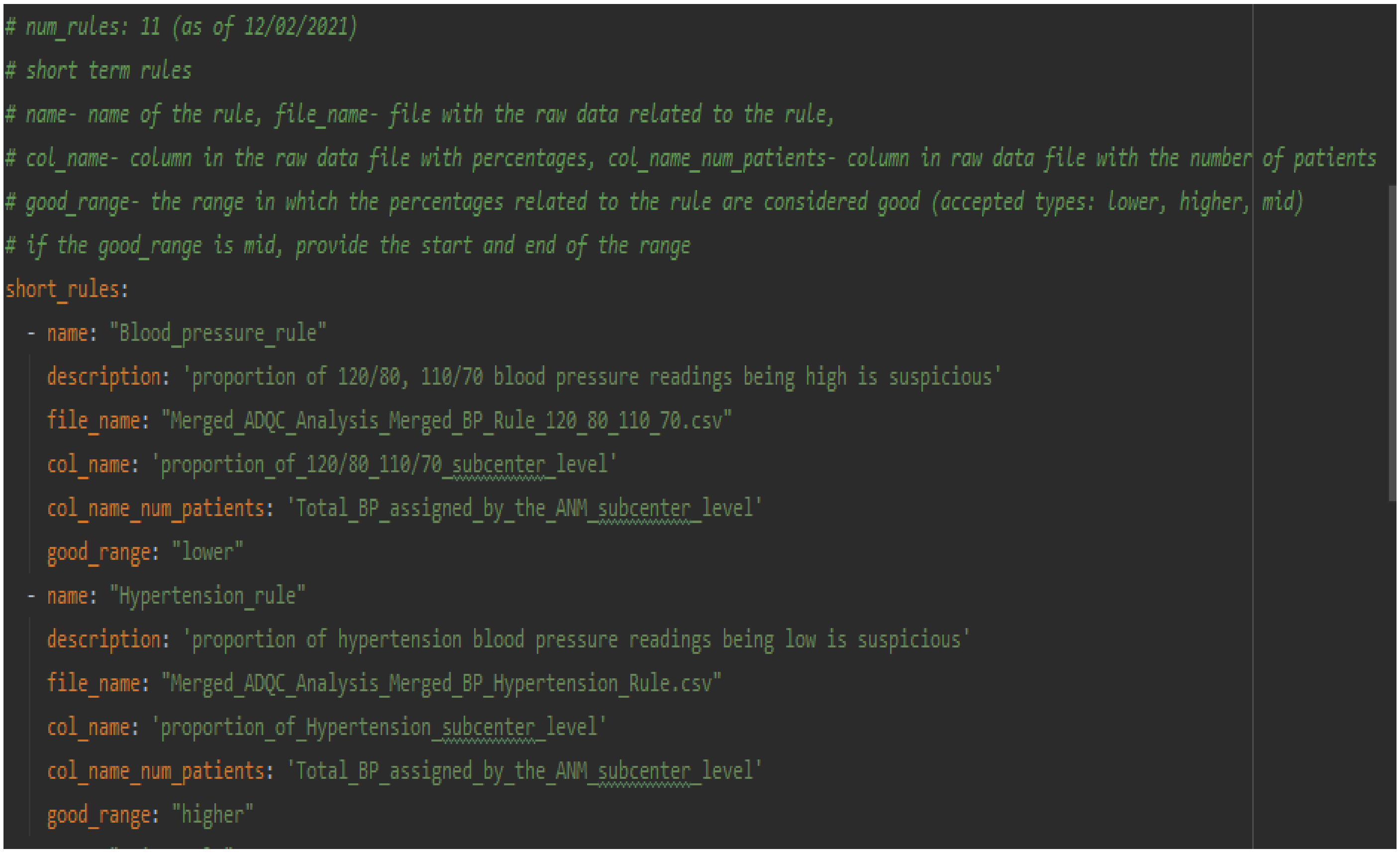
Example of short-term rule: % of Blood Pressure readings that are 120/80 or 110/70. We know that higher percentage corresponds to non-diligence.

Example contradiction rule: Entering “No equipment to test” to one patient and recording Blood pressure readings for another patient in the same camp. We know that towards 0% is diligence.

How to configure the rules?

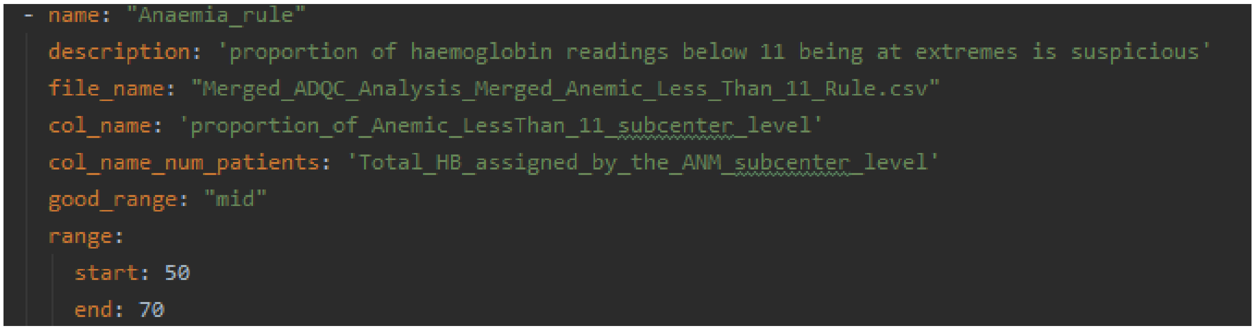
These two types of rules can be defined in the configuration file and the system will automatically handle them. Few examples of how to configure the rules are provided below.

Short-term rules where lower range of the percentages are good and higher range of the percentages are good are given below respectively.



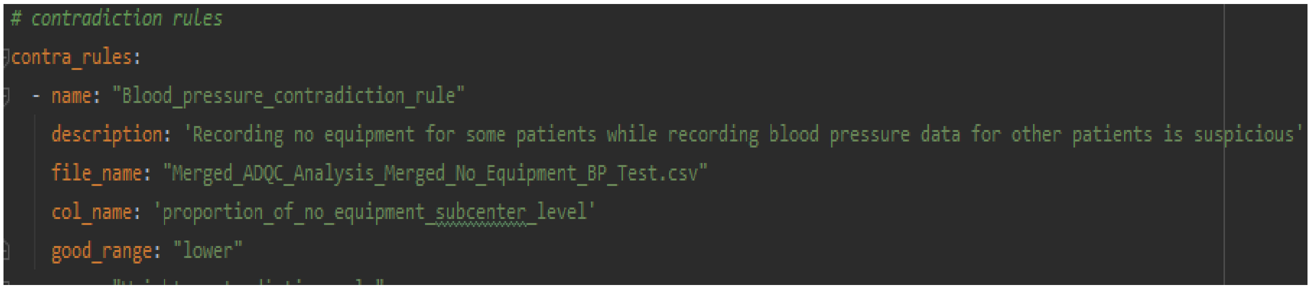
```yaml
# num_rules: 11 (as of 12/02/2021)
# short term rules
# name- name of the rule, file_name- file with the raw data related to the rule,
# col_name- column in the raw data file with percentages, col_name_num_patients- column in raw data file with the number of patients
# good_range- the range in which the percentages related to the rule are considered good (accepted types: lower, higher, mid)
# if the good_range is mid, provide the start and end of the range
short_rules:
  - name: "Blood_pressure_rule"
    description: 'proportion of 120/80, 110/70 blood pressure readings being high is suspicious'
    file_name: "Merged_ADQC_Analysis_Merged_BP_Rule_120_80_110_70.csv"
    col_name: 'proportion_of_120/80_110/70_subcenter_level'
    col_name_num_patients: 'Total_BP_assigned_by_the_ANM_subcenter_level'
    good_range: "lower"
  - name: "Hypertension_rule"
    description: 'proportion of hypertension blood pressure readings being low is suspicious'
    file_name: "Merged_ADQC_Analysis_Merged_BP_Hypertension_Rule.csv"
    col_name: 'proportion_of_Hypertension_subcenter_level'
    col_name_num_patients: 'Total_BP_assigned_by_the_ANM_subcenter_level'
    good_range: "higher"
```

An example of a mid-range rule is given below.



```yaml
  - name: "Anaemia_rule"
    description: 'proportion of haemoglobin readings below 11 being at extremes is suspicious'
    file_name: "Merged_ADQC_Analysis_Merged_Anemic_Less_Than_11_Rule.csv"
    col_name: 'proportion_of_Anemic_LessThan_11_subcenter_level'
    col_name_num_patients: 'Total_HB_assigned_by_the_ANM_subcenter_level'
    good_range: "mid"
    range:
      start: 50
      end: 70
```

An example of a contradiction rule is given below.



```yaml
# contradiction rules
contra_rules:
  - name: "Blood_pressure_contradiction_rule"
    description: 'Recording no equipment for some patients while recording blood pressure data for other patients is suspicious'
    file_name: "Merged_ADQC_Analysis_Merged_No_Equipment_BP_Test.csv"
    col_name: 'proportion_of_no_equipment_subcenter_level'
    good_range: "lower"
```

Calculating non diligence probabilities in each rule:

We describe here how we extract non-diligence probability using the BP rule – rest for other rules are similar or flipped where 0% is non-diligence. In BP rule 100% is non-diligent behavior with probability 1. We look at the data for each ANM for each heath camp to obtain the percentages for any rule. We filter out percentage that are exactly 0% or 100% - as these are for sure not diligent (or dilgent). For the remaining percentage we plot a density distribution using KDE (kded1d library in R). The location of this code is:

Code filename: diligence/rfunc/helpers.r

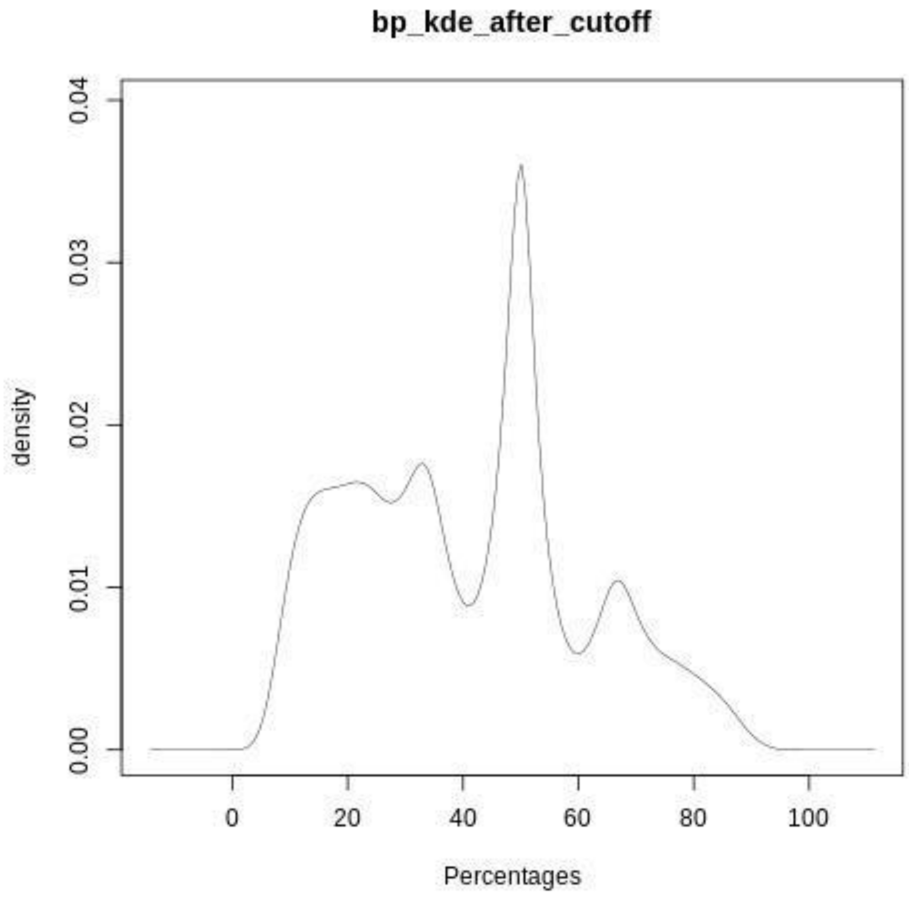
function name: func_get_all_kdes

Then, given the percentage, say x, for a heath camp for an ANM – the probability of non-diligence p is the probability mass between (0,x). Clearly as x increases, the probability of non-diligence is increasing and is exactly 1 when percentage is 100. The location of this code is:

Code filename: diligence/rfunc/helpers.r

function name: func_get_prob_mass_trans

See figure below for example for the BP rule:



The behaviors of ANM over time:

We compute the vector of non-diligence probabilities (short-term and contradiction rules, 11 in total) for an ANM at multiple time points that are separated by one month. Then we get the L2 norm (simple norm score) of these vectors. Then we get the last 6 months scores of each ANM and calculate the average norm score and a weighted norm score giving more weight to the recent months. Higher scores correspond to less diligent ANMs. We identify the trend from the difference between the average norm score and the weighted norm score. Positive trend values correspond to improving ANMs.

Code filename: diligence/scores.py

Class name: Scores

Prediction

The non-diligence score of each ANM will be calculated using past 6 months non-diligence scores. A simple linear regressor model is used here, due to the amount of data available.

ANMs with the highest and lowest predicted non diligence scores are almost same as the ANMs with the highest and lowest average non diligence scores of the past 6 months.

Code filename: diligence/predictor.py

Class name: Predictor

Clustering:

We perform kmeans clustering on the percentage vectors. We observe a clear separation when using 4 clusters. This parameter also can be configured using the configuration file.

Re-clustering is not done every month and the period of re-clustering is configurable in the config file.

Each cluster represents the behavior w.r.t the provided rules.

Code filename: diligenceprediction/clustering.py

Class name: Clustering

Cluster explanations

Cluster explanations are generated automatically using a sparse hyperplane tree and the no diligence probabilities of cluster centers.

Code filename: diligenceprediction/explain.py

Class name: Explain

Rules with low standard deviation is not considered for cluster explanations, since they hardly affect clusters' behavior. The threshold of the standard deviation is configurable using the config.yaml file.

The cluster explanations are provided in 4 abstraction levels.

level 0 : outputs whether the cluster is generally good or bad

level 1: whether the cluster is good/bad for important rules (important rules selected automatically*)

level 2: whether the cluster is good/bad for average rules

level 3: full description

*Important rules are selected using the weights of the hyperplane tree. Whether each rule is good or bad in each cluster is considered by comparing the cluster centers against the cluster center means.

Hyperplane tree

In the sparse hyperplane tree we work on a top down approach to build the boundaries between clusters, using k-means clustering as the reference. This will allow us to create axis aligned boundaries as well as oblique boundaries, which will provide a better explanation.

Steps of the full hyperplane tree:

1. Use k-means clustering and find all cluster centers.
2. Find all hyperplanes, all hyperplanes between all pairs of centers c_i, c_j
3. Find relevant hyperplanes:

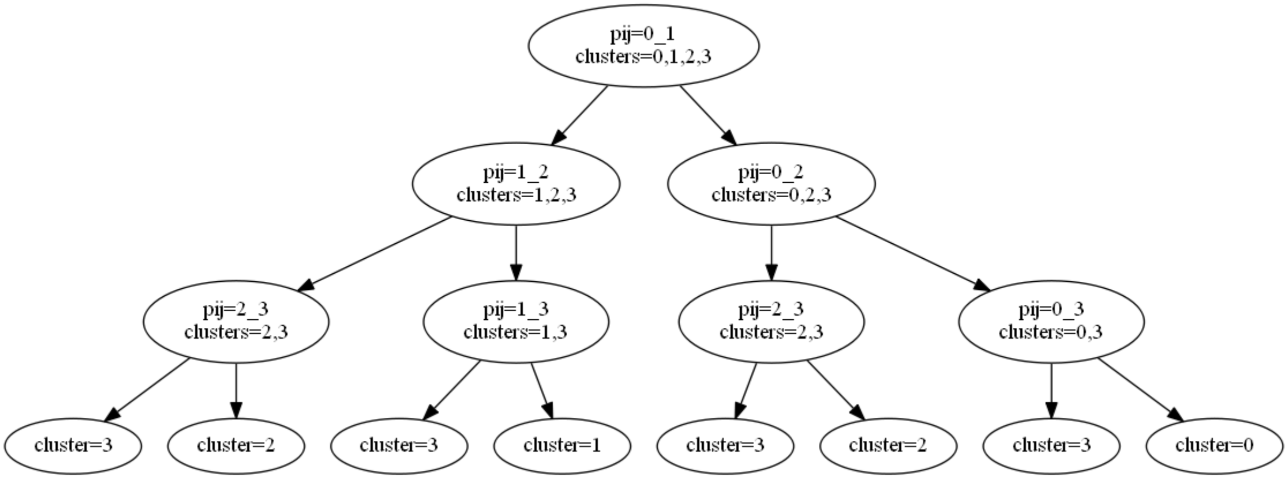
For any center pair c_i, c_j, let the hyperplane between them be P_{i,j}. By definition, every point on P_{i,j} is equidistant from c_i and c_j. P_{i,j} is only relevant if it actually separates the cluster with center c_i and cluster with center c_j, meaning P_{i,j} is the boundary between these two clusters at least for some part of the clusters. We find if a hyperplane is relevant as follows:

- For being relevant, there must be one point on P_{i,j} that is closest to c_i and c_j and not to any other center.

- The above can be solved using an optimization. Find x with the constraints that P_{i,j}(x) = 0 (meaning x is on the hyperplane) and ||x - c_k|| > ||x - c_i|| and ||x - c_k|| > ||x - c_j|| for all k different from i,j. Note the norm constraints are actually linear and so is P_{i,j}(x) = 0.

Find hyperplanes (among relevant hyperplanes) that cause most distinction, meaning those that divide the space so that most clusters lie on one side or another. For this, we need an ability to find which all clusters does a hyperplane intersect. This is also doable as an optimization. If a hyperplane intersects a cluster c_k then there is a point on the hyperplane that is closest to c_k compared to any other center. Also, it is possible to find which cluster centers lie on one side of P_{i,j} and which lie on the other side. Using these, we can find which clusters lie on one side and which lie on the other side completely (meaning without intersecting).

4. The above construct a binary tree with possible clusters - there is an overlap among possible clusters on left and right, so we choose hyperplanes that cause the most distinction meaning least intersection with other clusters. But, we also aim to get good balance on the left and right. This is done by maximizing the number of clusters on left and right.



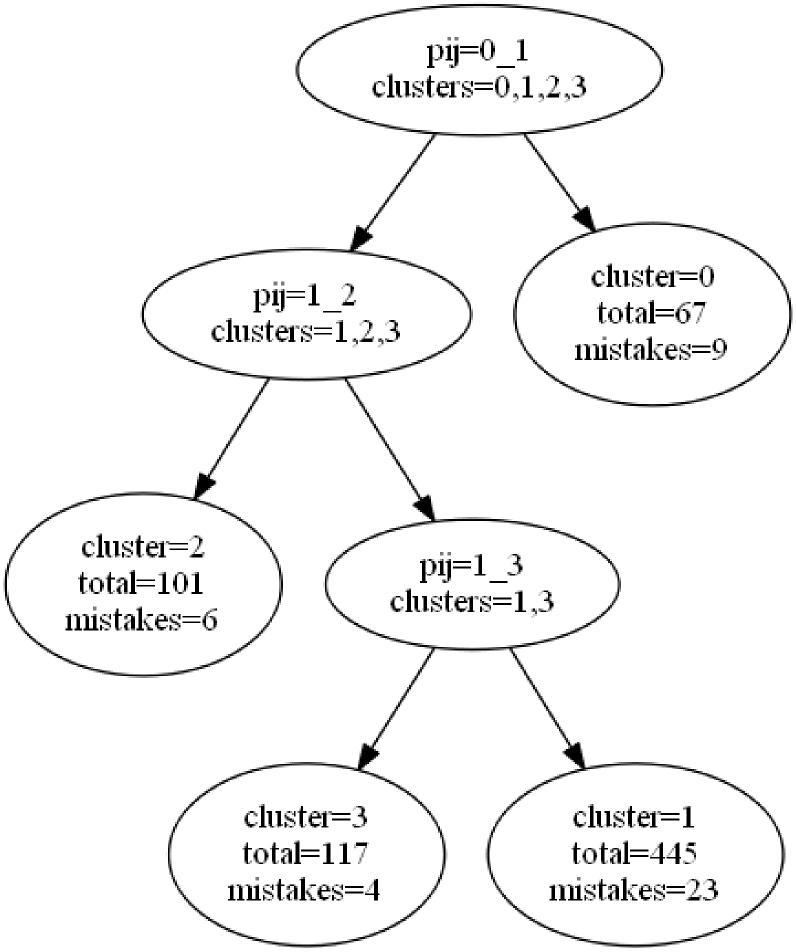
Sparse hyperplane tree:

Additionally, to creating a hyperplane tree, we can observe that: not all the hyperplanes in the full hyperplane tree will be needed to separate the data in the training distribution. Also, not all the dimensions of a hyperplane are important for explanations.

Hence, for dimensionality reduction we use L1 norm. For removing less important hyperplanes, we prune the tree if the proportion of data in each node is below a threshold.

Code filename: diligenceprediction/tree.py

Class name: hpTree2



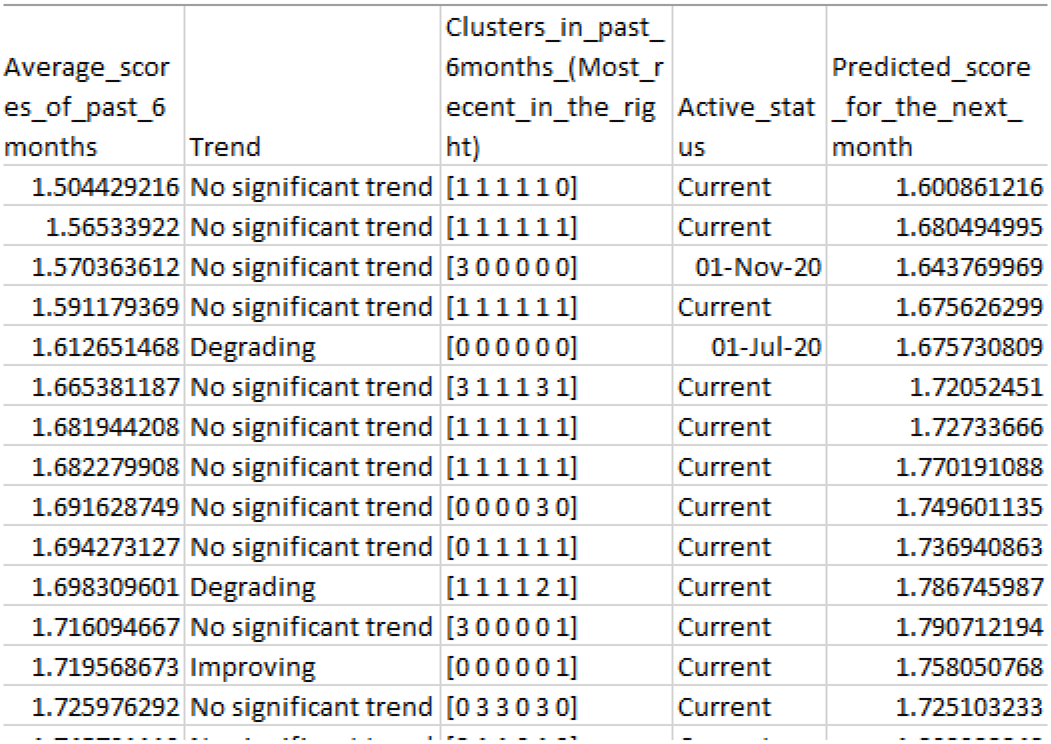
Outputs

We output 3 main files.

1. **ANM diligence summary**

   (outputs/summary_output.csv)

   - Output at a higher abstraction
   - Intended for the use of the field team
   - Contains the average scores (sorted), trend(improving, degrading or no significant trend)
   - Clusters to which the ANMs fell into in the past 6 months, Active status of the ANM (most recent month that the ANM worked)
   - Predicted score for the next month

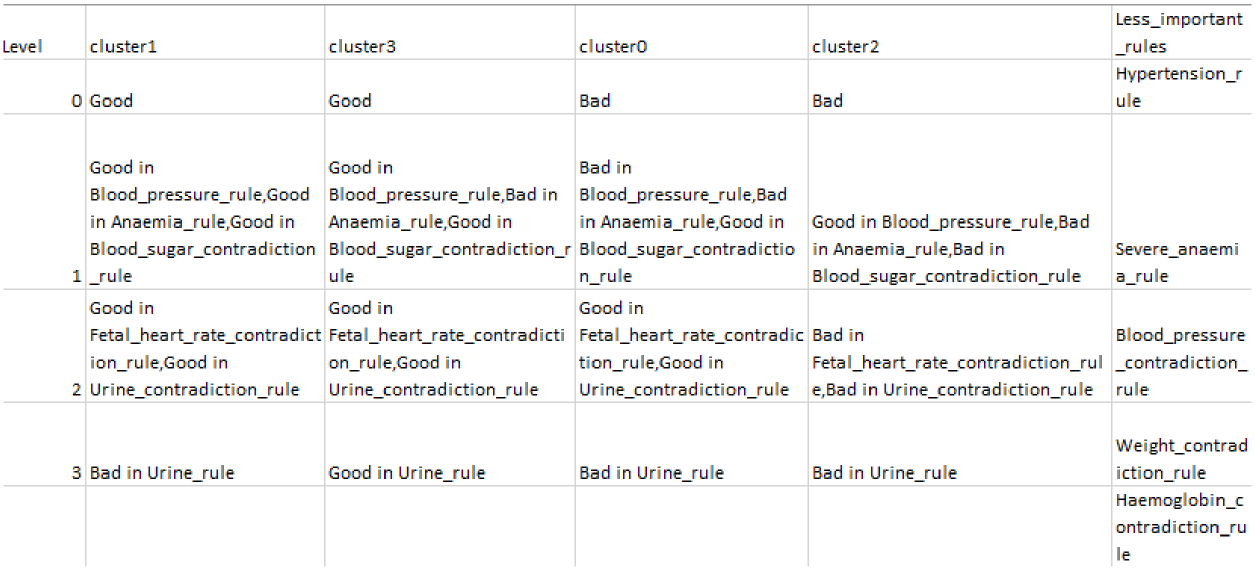


Average_scores_of_past_6 months	Trend	Clusters_in_past_6months_(Most_recent_in_the_right)	Active_status	Predicted_score_for_the_next_month
1.504429216	No significant trend	[1 1 1 1 1 0]	Current	1.600861216
1.56533922	No significant trend	[1 1 1 1 1 1]	Current	1.680494995
1.570363612	No significant trend	[3 0 0 0 0 0]	01-Nov-20	1.643769969
1.591179369	No significant trend	[1 1 1 1 1 1]	Current	1.675626299
1.612651468	Degrading	[0 0 0 0 0 0]	01-Jul-20	1.675730809
1.665381187	No significant trend	[3 1 1 1 3 1]	Current	1.72052451
1.681944208	No significant trend	[1 1 1 1 1 1]	Current	1.72733666
1.682279908	No significant trend	[1 1 1 1 1 1]	Current	1.770191088
1.691628749	No significant trend	[0 0 0 0 3 0]	Current	1.749601135
1.694273127	No significant trend	[0 1 1 1 1 1]	Current	1.736940863
1.698309601	Degrading	[1 1 1 1 2 1]	Current	1.786745987
1.716094667	No significant trend	[3 0 0 0 0 1]	Current	1.790712194
1.719568673	Improving	[0 0 0 0 0 1]	Current	1.758050768
1.725976292	No significant trend	[0 3 3 0 3 0]	Current	1.725103233

2. Explanations of each cluster

(outputs/ cluster_explain.csv, cluster_explain_validate.csv)

- In 4 granularity levels
- Less important rules are provided in the right side

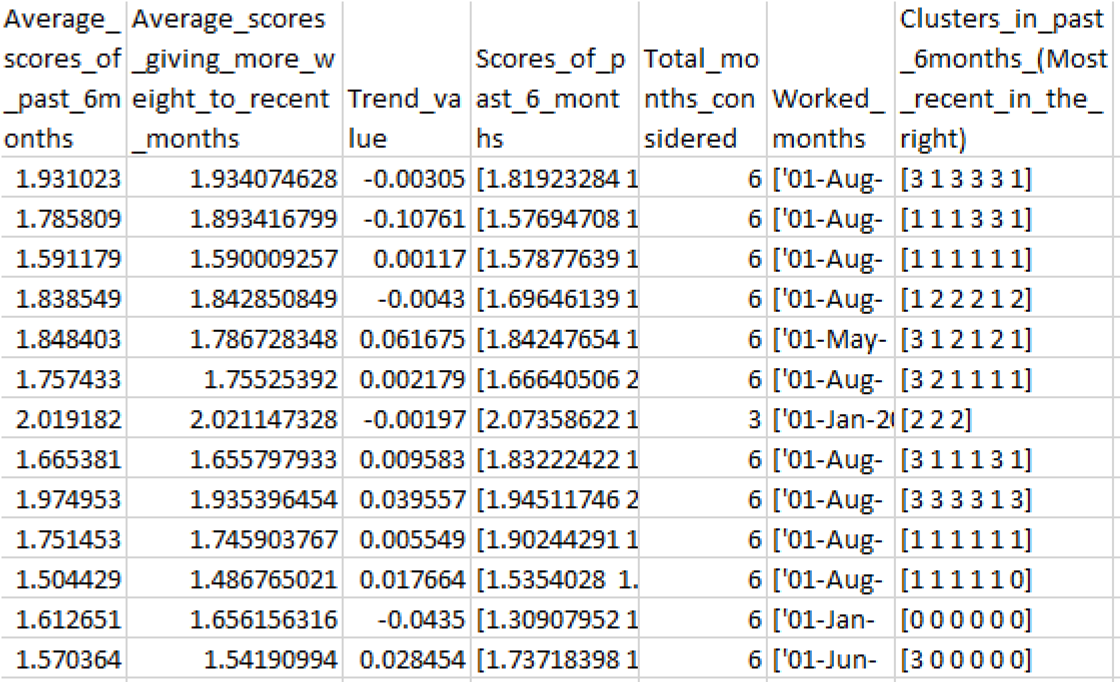


Level	cluster1	cluster3	cluster0	cluster2	Less_important_rules
0	Good	Good	Bad	Bad	Hypertension_rule
1	Good in Blood_pressure_rule,Good in Anaemia_rule,Good in Blood_sugar_contradiction_rule	Good in Blood_pressure_rule,Bad in Anaemia_rule,Good in Blood_sugar_contradiction_rule	Bad in Blood_pressure_rule,Bad in Anaemia_rule,Good in Blood_sugar_contradiction_rule	Good in Blood_pressure_rule,Bad in Anaemia_rule,Bad in Blood_sugar_contradiction_rule	Severe_anaemia_rule
2	Good in Fetal_heart_rate_contradiction_rule,Good in Urine_contradiction_rule	Good in Fetal_heart_rate_contradiction_rule,Good in Urine_contradiction_rule	Good in Fetal_heart_rate_contradiction_rule,Good in Urine_contradiction_rule	Bad in Fetal_heart_rate_contradiction_rule,Bad in Urine_contradiction_rule	Blood_pressure_contradiction_rule
3	Bad in Urine_rule	Good in Urine_rule	Bad in Urine_rule	Bad in Urine_rule	Weight_contradiction_rule
					Haemoglobin_contradiction_rule

3. **ANM diligence with details**

   (outputs/ raw_summary_output.csv)

   - Raw output data
   - Intended for the use of in-depth inspection
   - Contains the average scores, weighted scores, all scores in the past 6 months, trend values (raw)
   - Clusters to which the ANMs fell into in the past 6 months



Average_scores_of_past_6months	Average_scores_giving_more_weight_to_recent_months	Trend_value	Scores_of_past_6_months	Total_months_considered	Worked_months	Clusters_in_past_6months_(Most_recent_in_the_right)
1.931023	1.934074628	-0.00305	[1.81923284 1	6	['01-Aug-	[3 1 3 3 3 1]
1.785809	1.893416799	-0.10761	[1.57694708 1	6	['01-Aug-	[1 1 1 3 3 1]
1.591179	1.590009257	0.00117	[1.57877639 1	6	['01-Aug-	[1 1 1 1 1 1]
1.838549	1.842850849	-0.0043	[1.69646139 1	6	['01-Aug-	[1 2 2 2 1 2]
1.848403	1.786728348	0.061675	[1.84247654 1	6	['01-May-	[3 1 2 1 2 1]
1.757433	1.75525392	0.002179	[1.66640506 2	6	['01-Aug-	[3 2 1 1 1 1]
2.019182	2.021147328	-0.00197	[2.07358622 1	3	['01-Jan-20	[2 2 2]
1.665381	1.655797933	0.009583	[1.83222422 1	6	['01-Aug-	[3 1 1 1 3 1]
1.974953	1.935396454	0.039557	[1.94511746 2	6	['01-Aug-	[3 3 3 3 1 3]
1.751453	1.745903767	0.005549	[1.90244291 1	6	['01-Aug-	[1 1 1 1 1 1]
1.504429	1.486765021	0.017664	[1.5354028 1.	6	['01-Aug-	[1 1 1 1 1 0]
1.612651	1.656156316	-0.0435	[1.30907952 1	6	['01-Jan-	[0 0 0 0 0 0]
1.570364	1.54190994	0.028454	[1.73718398 1	6	['01-Jun-	[3 0 0 0 0 0]

- Display cluster numbers corresponds to their diligence level.

  Ex: 0 good, 3 bad

- Other outputs such as the hyperplane tree figures are saved in the /intermediate_outputs directory.
- Organized input percentages for each ANM are stored in /extra_outputs directory for easy reference.

How to setup the package

1. Install anaconda following these instructions

2. Navigate to the **root project folder**. (**Below commands are given assuming executions are done from the root package**)
3. Execute the below command. This will install all the used dependencies in the MVP with relevant versions of packages.

```
conda env create -f setup/environment.yaml
```

4. Then you can activate the conda environment using below command

```
conda activate hyper2
```

5. Needed R packages can be installed by executing the below command.

```
python setup/setup_r_lib.py
```

6. Choose the needed CRAN mirror when prompted. Recommended to use the 1st mirror.
7. Now the environment is ready with all needed versions of dependencies.

How to run the MVP package

1. Setup your environment following the above
2. Add the data files to a folder you prefer and add it to the ‘location’ field in the configuration file
3. Add the configuration file to the root folder.
4. Execute the below command to get the outputs

```
python -m diligence
```

Appendix

Configuration file (config.yaml)

```
# configuration file

# Change the below 2 fields every month. Others can be kept unchanged.
# months to be considered in raw data. This has to be changed whenever the script is
executed each month
months: ['01-Jan-2020', '01-Feb-2020', '01-Mar-2020', '01-May-2020', '01-Jun-2020',
         '01-Jul-2020', '01-Aug-2020', '01-Sep-2020', '01-Oct-2020', '01-Nov-2020', '01-
Dec-2020', '01-Jan-2021']
# directory with the raw data percentages of all the months above
location: "input_data/"
```

```yaml
# location and name of the clustering history file. It should be in the same folder
clustering_history: 'clustering_history.csv'
# after how many months re-clustering should be done
# put 0 if it needs to be immediately re-clustered
reclustering_months: 3
# number of clusters
num_clusters: 4

# how the ANMs with low patients should be filtered out
# ANMs who have patients less than threshold_each in each month for each short_rule and
ANMs who have patients less
# than threshold_all * number of months in each short_rule is filtered out as noisy ANMs
filtering:
  threshold_each: 8
  threshold_all: 3

# rule_ignore_std- Rules with standard deviation less than the below provided threshold
will be considered as
# less important rules and will be excluded from the cluster explanations
# hyperplane_tree- whether to draw hyperplane tree or not
explain:
  rule_ignore_std: 10
  hyperplane_tree: True

# num_rules: 11 (as of 12/02/2021)
# short term rules
# name- name of the rule, file_name- file with the raw data related to the rule,
# col_name- column in the raw data file with percentages, col_name_num_patients- column
in raw data file with the number of patients
# good_range- the range in which the percentages related to the rule are considered good
(accepted types: lower, higher, mid)
# if the good_range is mid, provide the start and end of the range
short_rules:
  - name: "Blood_pressure_rule"
    description: 'proportion of 120/80, 110/70 blood pressure readings being high is
suspicious'
    file_name: "Merged_ADQC_Analysis_Merged_BP_Rule_120_80_110_70.csv"
    col_name: 'proportion_of_120/80_110/70_subcenter_level'
    col_name_num_patients: 'Total_BP_assigned_by_the_ANM_subcenter_level'
    good_range: "lower"
  - name: "Hypertension_rule"
    description: 'proportion of hypertension blood pressure readings being low is
suspicious'
    file_name: "Merged_ADQC_Analysis_Merged_BP_Hypertension_Rule.csv"
    col_name: 'proportion_of_Hypertension_subcenter_level'
    col_name_num_patients: 'Total_BP_assigned_by_the_ANM_subcenter_level'
    good_range: "higher"
  - name: "Urine_rule"
    description: 'proportion of urine readings as absent being high is suspicious'
    file_name: "Merged_ADQC_Analysis_Merged_Urine_Absent_Test.csv"
    col_name: 'proportion_of_UrineTest_Absent_subcenter_level'
    col_name_num_patients: 'Total_Urine_assigned_by_the_ANM_subcenter_level'
    good_range: "lower"
  - name: "Anaemia_rule"
    description: 'proportion of haemoglobin readings below 11 being at extremes is
suspicious'
    file_name: "Merged_ADQC_Analysis_Merged_Anemic_Less_Than_11_Rule.csv"
    col_name: 'proportion_of_Anemic_LessThan_11_subcenter_level'
    col_name_num_patients: 'Total_HB_assigned_by_the_ANM_subcenter_level'
    good_range: "mid"
    range:
      start: 50
      end: 70
  - name: "Severe_anaemia_rule"
    description: 'proportion of haemoglobin readings below 7 being low is suspicious'
    file_name: "Merged_ADQC_Analysis_Merged_Anemic_Less_Than_7_Rule.csv"
    col_name: 'proportion_of_Anemic_LessThan_7_subcenter_level'
    col_name_num_patients: 'Total_HB_assigned_by_the_ANM_subcenter_level'
    good_range: "higher"

# contradiction rules
contra_rules:
  - name: "Blood_pressure_contradiction_rule"
    description: 'Recording no equipment for some patients while recording blood pressure
data for other patients is suspicious'
    file_name: "Merged_ADQC_Analysis_Merged_No_Equipment_BP_Test.csv"
    col_name: 'proportion_of_no_equipment_subcenter_level'
    good_range: "lower"
  - name: "Weight_contradiction_rule"
    description: 'Recording no equipment for some patients while recording weight data
for other patients is suspicious'
    file_name: "Merged_ADQC_Analysis_Merged_No_Equipment_Weight_Test.csv"
    col_name: 'proportion_of_no_equipment_subcenter_level'
    good_range: "lower"
  - name: "Haemoglobin_contradiction_rule"
    description: 'Recording no equipment for some patients while recording haemoglobin
data for other patients is suspicious'
    file_name: "Merged_ADQC_Analysis_Merged_No_Equipment_HB_Test.csv"
    col_name: 'proportion_of_no_equipment_subcenter_level'
    good_range: "lower"
  - name: "Blood_sugar_contradiction_rule"
    description: 'Recording no equipment for some patients while recording blood sugar
data for other patients is suspicious'
    file_name: "Merged_ADQC_Analysis_Merged_No_Equipment_Blood_Sugar_Test.csv"
    col_name: 'proportion_of_no_equipment_subcenter_level'
    good_range: "lower"
  - name: "Fetal_heart_rate_contradiction_rule"
    description: 'Recording no equipment for some patients while recording fetal heart
rate data for other patients is suspicious'
    file_name: "Merged_ADQC_Analysis_Merged_No_Equipment_Heart_Rate_Test.csv"
    col_name: 'proportion_of_no_equipment_subcenter_level'
    good_range: "lower"
  - name: 'Urine_contradiction_rule'
    description: 'Recording no equipment for some patients while recording urine data for
other patients is suspicious'
```

```
    file_name: "Merged_ADQC_Analysis_Merged_No_Equipment_Urine_Test.csv"
    col_name: 'proportion_of_no_equipment_subcenter_level'
    good_range: "lower"

# hyperplane tree hyperparameter tuning starting point (Do not change. Only for
debugging)
tree_param:
  a: 500
  b: 10
  c: 500
  t1: 0.05
  t2: 5
  allowed_mistake_proportion: 0.1
  min_levels: 3
```

Folder structure

```
DiligenceMeasuringSystem:.
|   clustering_history.csv
|   config.yaml
|   tree.txt
|
├───diligence
|   |   clustering.py
|   |   explain.py
|   |   kde.py
|   |   predictor.py
|   |   processinputdata.py
|   |   processoutputdata.py
|   |   scores.py
|   |   tree.py
|   |   tunetree.py
|   |   __init__.py
|   |   __main__.py
```

```
│   │
│   ├───rfunc
│   │       helpers.r
│
├───extra_outputs
│   │   all_reorganized_data_without_nan_filled.csv
│   │   all_reorganized_data_with_nan_filled.csv
│   │
│   ├───nan_filled
│   │       103_anm_data_nan_filled.csv
│   │       -------
│   └───without_nan_filled
│           103_anm_data_without_nan_filled.csv
│           ------
│
├───input_data
│       Merged_ADQC_Analysis_Merged_Anemic_Less_Than_11_Rule.csv
│       Merged_ADQC_Analysis_Merged_Anemic_Less_Than_7_Rule.csv
│       Merged_ADQC_Analysis_Merged_Blood_group_Rule.csv
│       Merged_ADQC_Analysis_Merged_BP_Hypertension_Rule.csv
│       Merged_ADQC_Analysis_Merged_BP_Rule_120_80_110_70.csv
│       Merged_ADQC_Analysis_Merged_Child_Death_Rule.csv
│       Merged_ADQC_Analysis_Merged_Child_EBF_Rule.csv
│       Merged_ADQC_Analysis_Merged_Diabetic_Urine_Sugar.csv
│       Merged_ADQC_Analysis_Merged_Fundal_Height_Rule.csv
│       Merged_ADQC_Analysis_Merged_HIV_Rule.csv
│       Merged_ADQC_Analysis_Merged_No_Equipment_Blood_Sugar_Test.csv
│       Merged_ADQC_Analysis_Merged_No_Equipment_BP_Test.csv
│       Merged_ADQC_Analysis_Merged_No_Equipment_HB_Test.csv
│       Merged_ADQC_Analysis_Merged_No_Equipment_Heart_Rate_Test.csv
│       Merged_ADQC_Analysis_Merged_No_Equipment_Urine_Test.csv
│       Merged_ADQC_Analysis_Merged_No_Equipment_Weight_Test.csv
│       Merged_ADQC_Analysis_Merged_Still_Birth_Rule.csv
│       Merged_ADQC_Analysis_Merged_Urine_Absent_Test.csv
│       Merged_ADQC_Analysis_Merged_VDRL_Rule.csv
│
├───intermediate_outputs
│       centers.csv
│       kb_tree.gv
│       kb_tree.gv.png
│       sub_center_id_new.csv
│       weights.csv
│
├───outputs
│       cluster_centers.csv
│       cluster_explain.csv
│       cluster_explain_validate.csv
│       raw_summary_output.csv
│       summary_output.csv
│
└───setup
        environment.yaml
        setup_r_lib.py
```