On Defining Rules for Data Fabrication

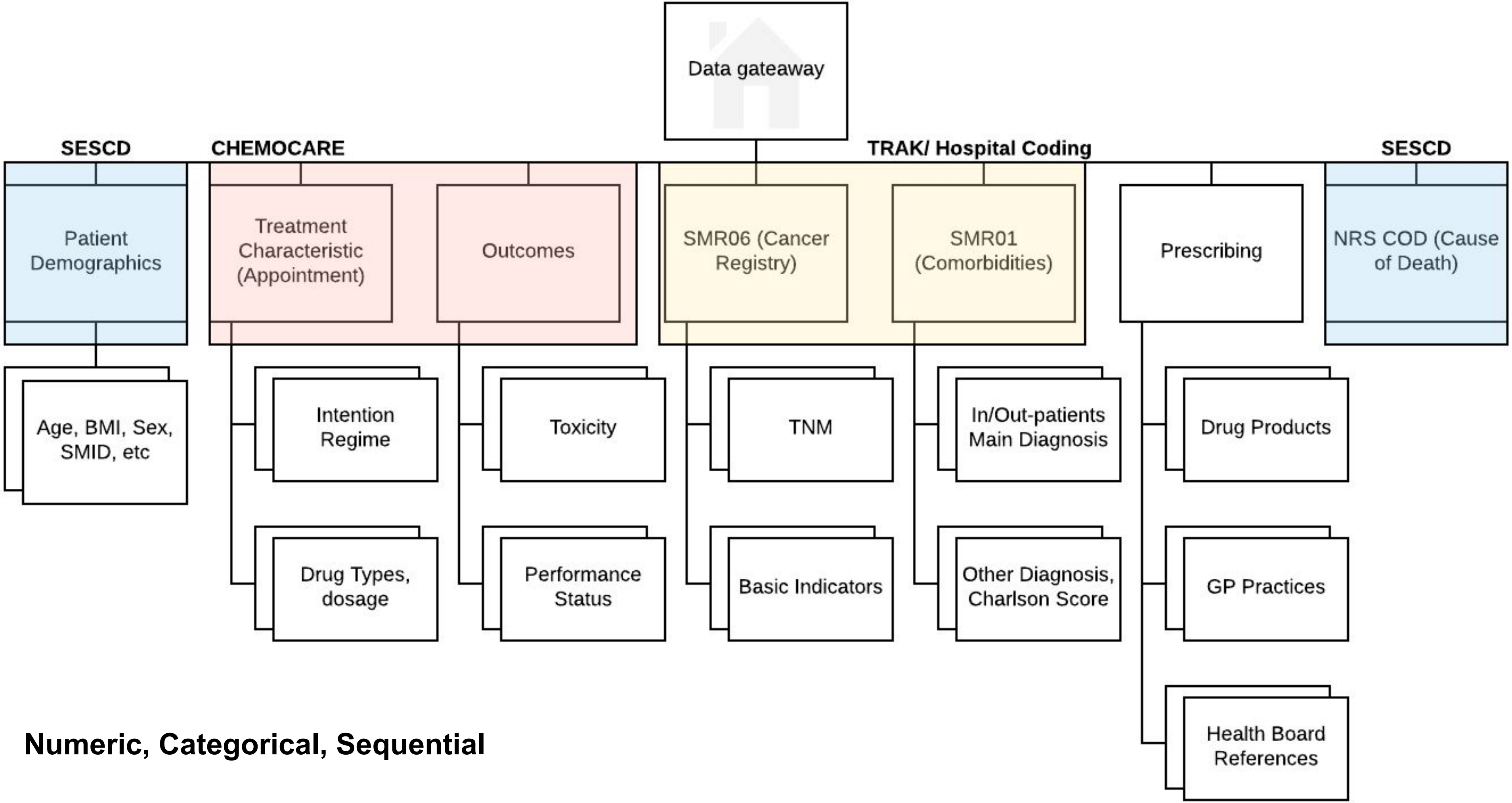
Agastya Silvina

Outline

- Introduction
- Data Structure
- Methodology
- Rule Design
- Validation

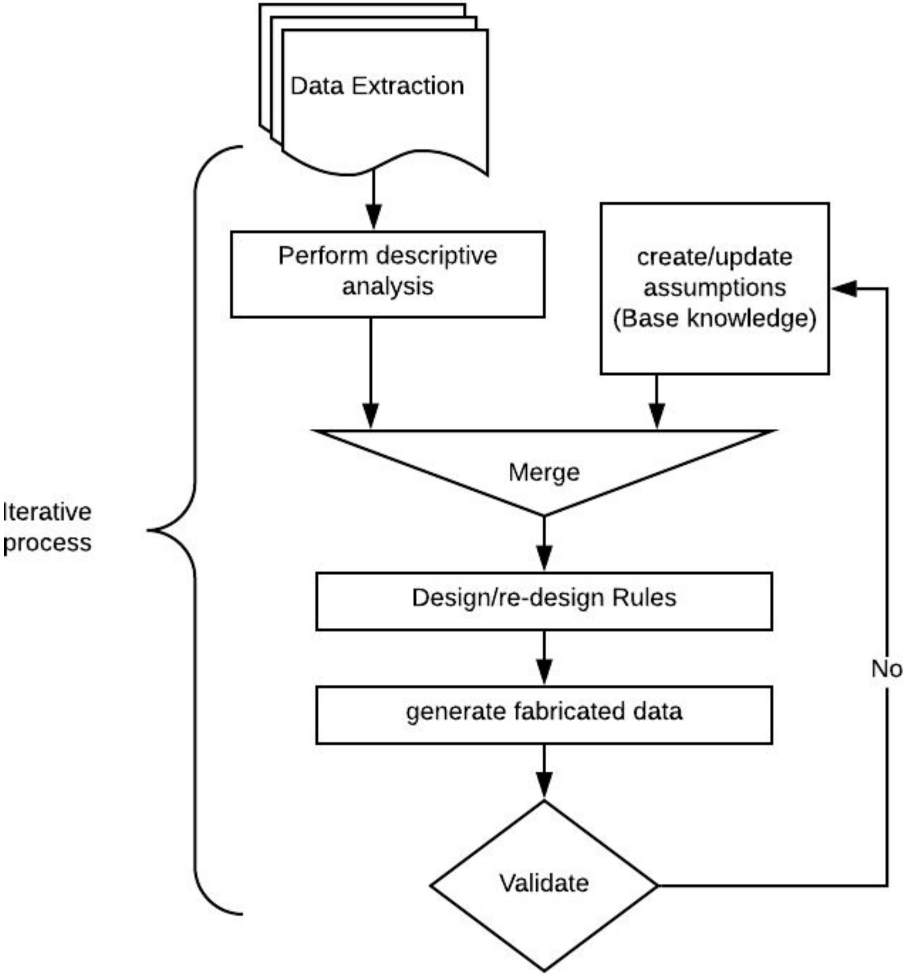
Introduction

- Generating Synthetic data for Cancer treatment
- We use IBM Constraint Solver
  - Determining the rules



Numeric, Categorical, Sequential

Methodology



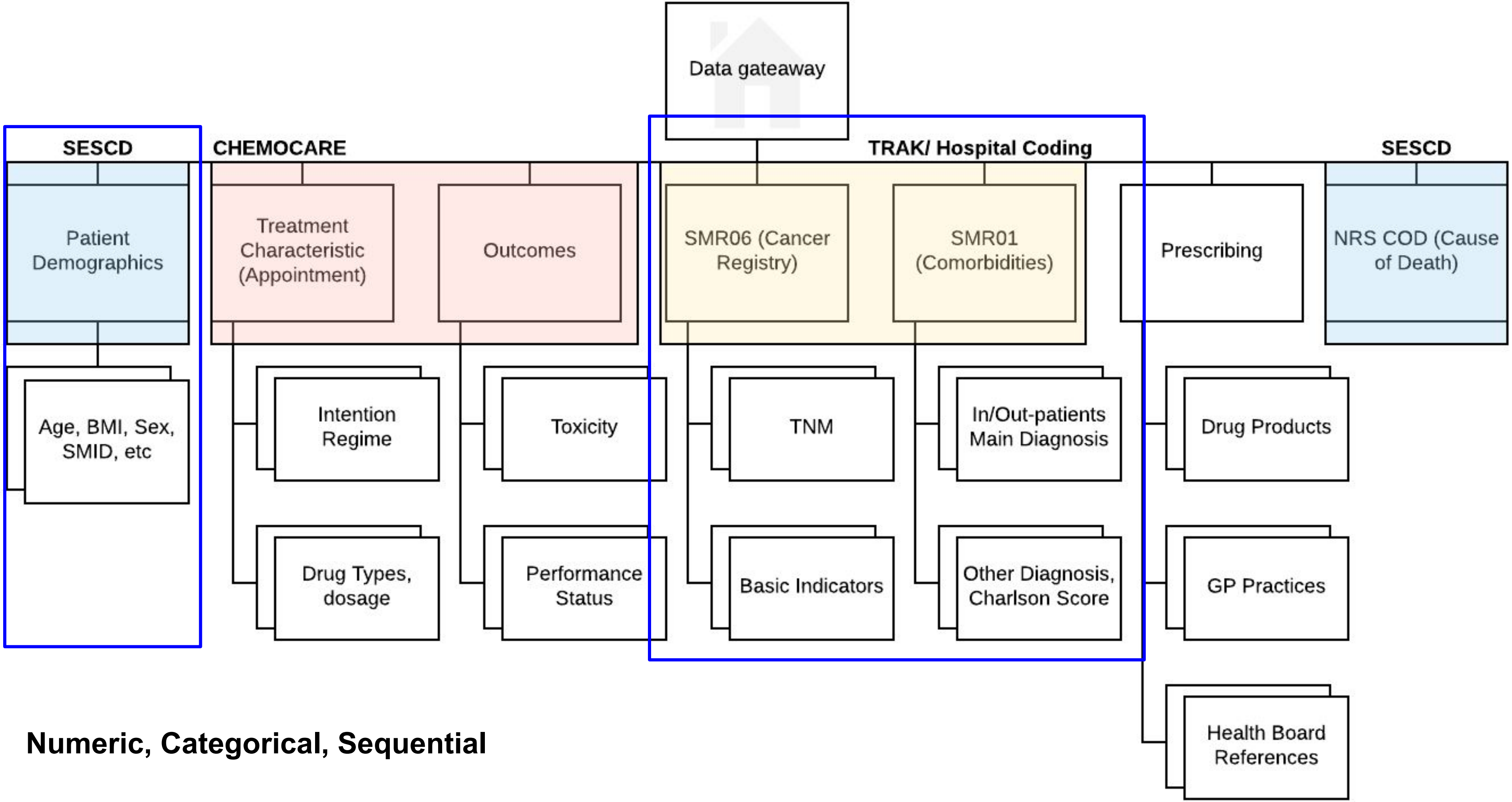
```
general.pulmonary_flag = (
 (general.metastasis1 == 'C34.9' ||
  general.metastasis2 == 'C34.9' ||
  general.metastasis3 == 'C34.9') ? 1 :
  randomWeightedValue(
    general.pulmonary_flag,
        1200? 0,
        120 ? 1
        )
 )
```

Rule Design

- Some Syntaxes:
  - *allDiff,*
  - *randomWeightedValue,*
  - *normalDistributionValue,*
  - *randomBool, randomCover,*
  - *Monotonic,*
  - *Inequality-equality (<, >, =) etc..*

Rule Design

- Manual process.
- **one big table** for fabricating the general data (e.g. patients demographics).
  - No sequences
  - No relation between each rows.
- **another table** for modelling the **chemotherapy**, with **several helper tables**.



Data gateaway

SESCD

Patient Demographics

Age, BMI, Sex, SMID, etc

CHEMOCARE

Treatment Characteristic (Appointment)

Intention Regime

Drug Types, dosage

Outcomes

Toxicity

Performance Status

TRAK/ Hospital Coding

SMR06 (Cancer Registry)

TNM

Basic Indicators

SMR01 (Comorbidities)

In/Out-patients Main Diagnosis

Other Diagnosis, Charlson Score

Prescribing

Drug Products

GP Practices

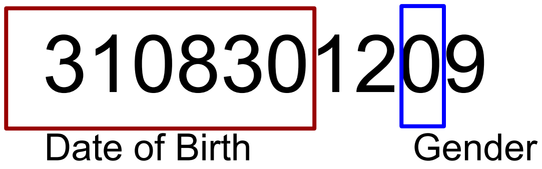
Health Board References

SESCD

NRS COD (Cause of Death)

Numeric, Categorical, Sequential

Rule Design



- Some fields (e.g. **CHI**) have more than one rules.

```
allDiff(from(general), general.chi)
```

```
  general.chi = concat(
        dateToString(general.DOB, DMy),
        intToString(general.D7),
        intToString(general.D8),
        intToString(general.D9),
        intToString(general.D10)
)
```

```
//D7,D8,D10
0 <= general.D7 <= 9
```

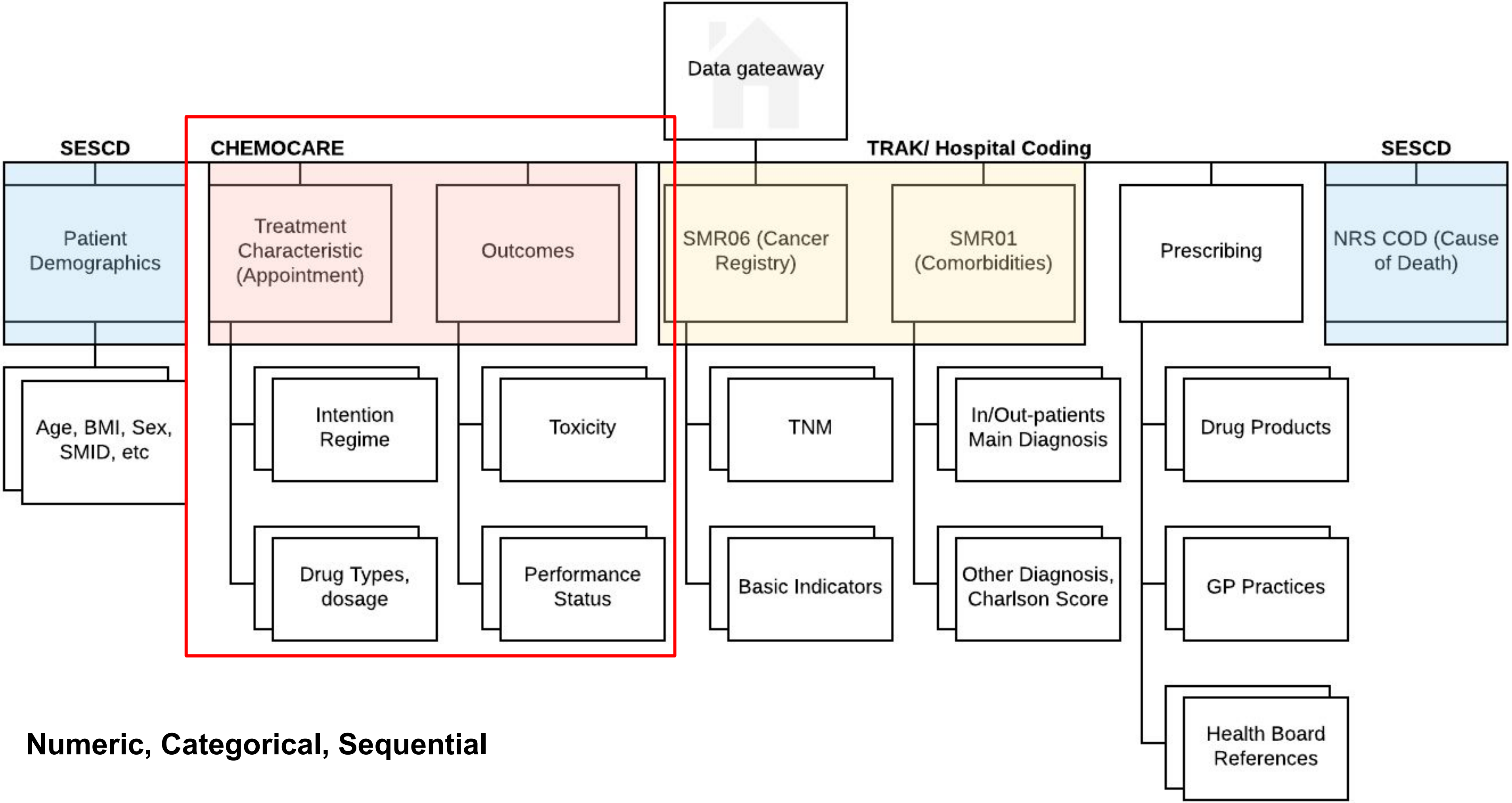
```
randomBool(99) ?
   general.D9 = {0,2,4,6,8} :
   general.D9 = {1,3,5,7,9}
```

Rule Design (AGE, BMI)

```
general.age = normalDistributionNumber(52.8, 3.112)
```

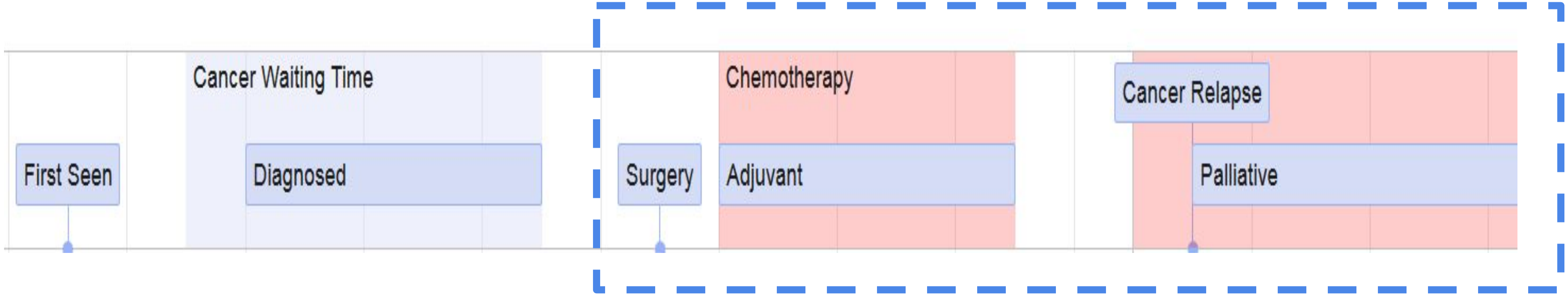
Shapiro-Wilk test

```
general.bmi =
  randomDistributionValue(general.bmi,
    50  : normalDistributionNumber(17.09, 1.25)
    450: normalDistributionNumber(22.6, 1.21 )
    400: normalDistributionNumber(27.33, 0.5)
    700: normalDistributionNumber(32.12,0.86)
    900: normalDistributionNumber(39.84,1.24)
  )
```



Numeric, Categorical, Sequential

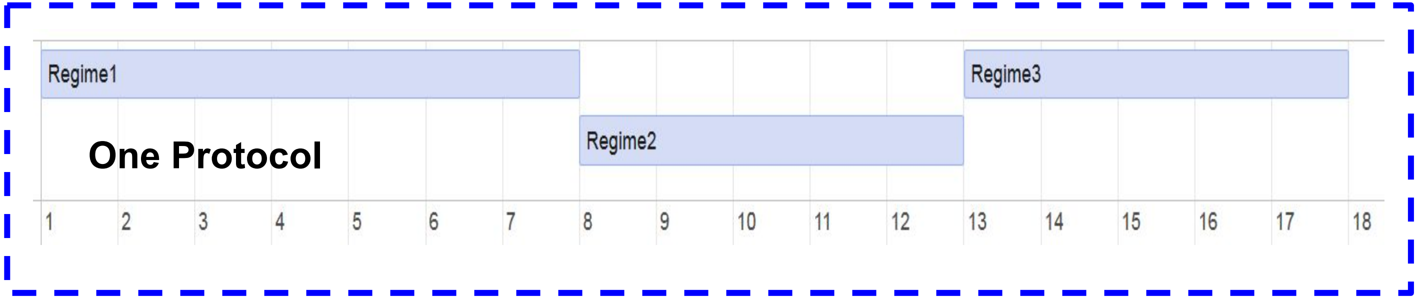
Patients’ Treatment Pathway



- A patient can only be treated with one intention at a time (e.g., Adjuvant)
- After a specific time has passed, the patient might be treated with other treatments with different intentions (e.g., Palliative, Curative)

Treatment Regimes

- Each intention has different regimes.
- Each regime has several different drugs.
- The treatment may last for several weeks or months
- A patient may be treated with several regimes at time.
- Each regime has one or more treatment cycles.
- Several different regimes may belong to one protocol.

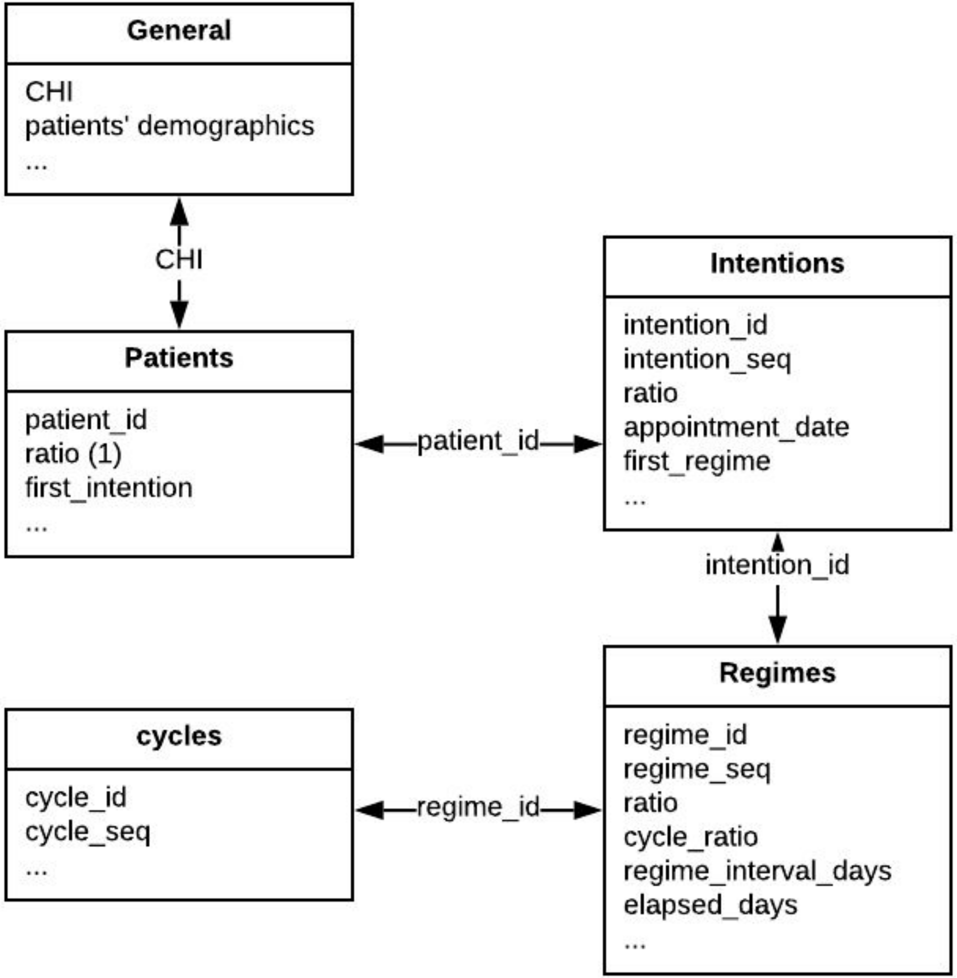


One Protocol

How does the table represent the treatment?

CHI	**APPOINTMENT DATE**	**INTENTION**	**REGIME**	**DRUG**	**CYCLE**
patient1	1/12/2019	Adjuvant	Regime A	drug1	1
patient1	1/12/2019	Adjuvant	Regime A	drug2	1
patient1	7/12/2019	Adjuvant	Regime A	drug1	2
patient1	7/12/2019	Adjuvant	Regime A	drug2	2
patient1	14/12/2019	Adjuvant	Regime A	drug1	3
patient1	14/12/2019	Adjuvant	Regime A	drug2	3

Rule Design



General

CHI
patients' demographics
...

CHI

Patients

patient_id
ratio (1)
first_intention
...

patient_id

Intentions

intention_id
intention_seq
ratio
appointment_date
first_regime
...

intention_id

Regimes

regime_id
regime_seq
ratio
cycle_ratio
regime_interval_days
elapsed_days
...

regime_id

cycles

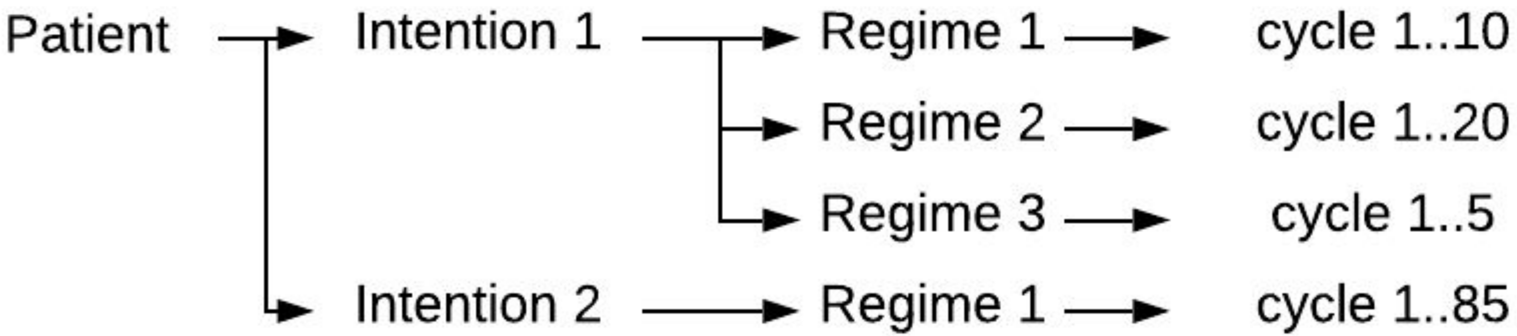
cycle_id
cycle_seq
...

```
monotonic(from(patients),
    patients.patient_id, {1}, 1)
```

```
monotonic(from(intentions),
  per(intentions.patient_id),
  intentions.intention_seq, {1}, 1)
```

```
numOf(from(regimes), regimes.intention_id
= intentions.intention_id) = (
 regimes.first_regime ==  'FEC-D (D)' ||
 regimes.first_regime ==  'FEC-D NEO’
  ….
    ? 3 : 2
)
```

ratio



Patient

Intention 1

Regime 1 cycle 1..10

Regime 2 cycle 1..20

Regime 3 cycle 1..5

Intention 2

Regime 1 cycle 1..85

regimes.**init_appointment_date** = intention.**appointment_date**

regimes.**appointment_date** =
regime.**init_appointment_date** + regime.**elapsed_days** +randomNumber(20,60)

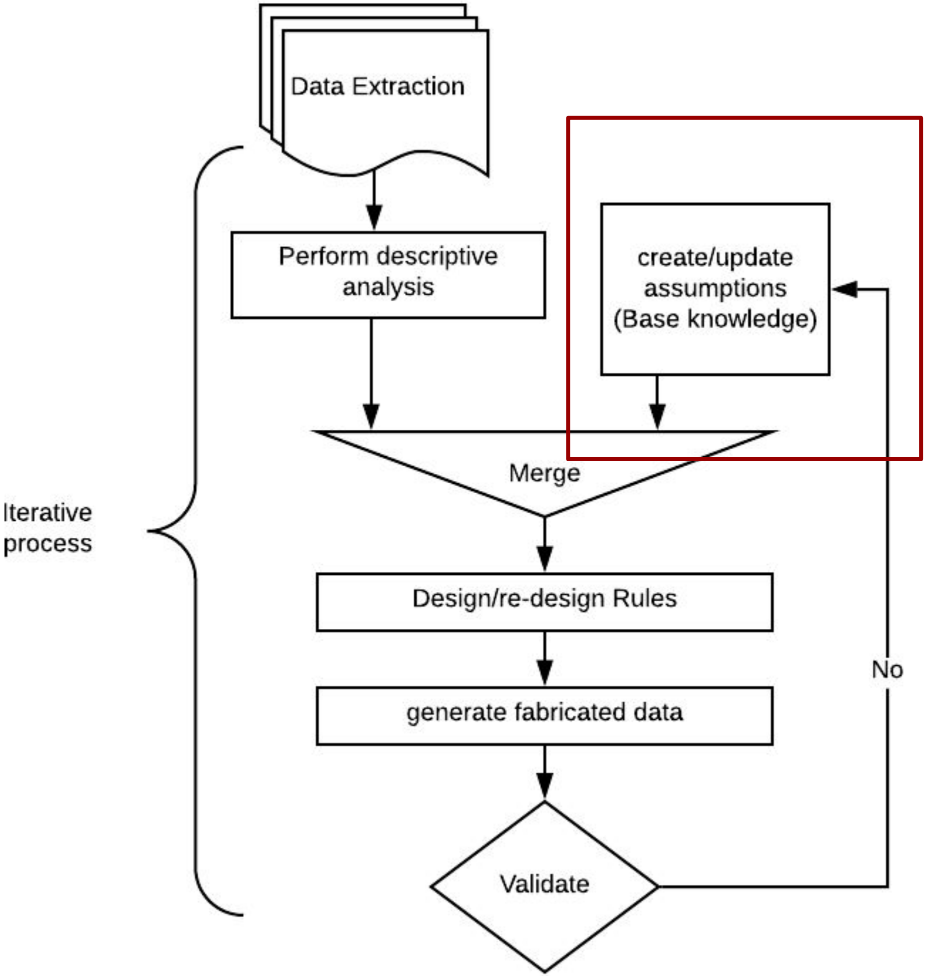
```
monotonic (from (regimes), per(regime.intention_id),
        regime.elapsed_days,
        regime.init_appointment_date,
        (cycle_ratio * regime_interval_days)
  )
```

Elapsed days

Data Validation

- How to differentiate real and fabricated data?
- We need **tools to differentiate between the real and fabricated data.**

Solution:
ML?



Data Extraction

Perform descriptive analysis

create/update assumptions (Base knowledge)

Merge

Iterative process

Design/re-design Rules

generate fabricated data

Validate

No

THANK YOU