HANDLING DATA GAPS IN TIME SERIES USING IMPUTATION

PRESENTED BY

CLARE JEON
ALF WHITEHEAD
HELLO!

CLARE JEON

ALF WHITEHEAD
1. What is time-series dataset & how do we use the data?

2. Missing values and imputation - What can we do about that missing data?

3. *Real world example:*
   Predicting blood glucose level of Type 1 Diabetes patients at 30 minutes horizon
## CONTENTS

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1</strong></td>
<td>What is time-series dataset &amp; how do we use the data?</td>
</tr>
<tr>
<td><strong>2</strong></td>
<td>Missing values and imputation - What can we do about that missing data?</td>
</tr>
<tr>
<td><strong>3</strong></td>
<td><em>Real world example:</em> Predicting blood glucose level of Type 1 Diabetes patients at 30 minutes horizon</td>
</tr>
</tbody>
</table>
Time series data is:

- A sequence of values of variables
- Measures the same thing over time and stores them in time order

https://www.kaggle.com/team-ai/bitcoin-price-prediction/version/1
**TIME SERIES ARE EVERYWHERE IN SIGNAL PROCESSING**

\[ y = f(t) \]

\( f(t) \) is implicitly defined everywhere, but the data is only sampled.

**A common error** is to use time series to track discrete events (e.g. events on a purchasing journey). This isn’t helpful because there is no “value” between the samples - use a sequence representation instead.
Our study system:
Glucose (Pathogenesis of Diabetes Mellitus)
FORECASTING IN HEALTHCARE: GLUCOSE IN TYPE 1 DIABETES
TYPE 1 DIABETES

Insulin therapy is the only way for patients.

Accurate blood glucose level prediction can help to figure out when insulin should be injected.
FORECASTING IN HEALTHCARE: GLUCOSE IN TYPE 1 DIABETES

Major Complications of Diabetes

Microvascular
- **Eye**: High blood glucose and high blood pressure can damage eye blood vessels, causing retinopathy, cataracts and glaucoma.
- **Kidney**: High blood pressure damages small blood vessels and excess blood glucose overworks the kidneys, resulting in nephropathy.
- **Neuropathy**: Hyperglycemia damages nerves in the peripheral nervous system. This may result in pain and/or numbness. Feet wounds may go undetected, get infected and lead to gangrene.

Macrovascular
- **Brain**: Increased risk of stroke and cerebrovascular disease, including transient ischemic attack, cognitive impairment, etc.
- **Heart**: High blood pressure and insulin resistance increase risk of coronary heart disease.
- **Extremities**: Peripheral vascular disease results from narrowing of blood vessels increasing the risk for reduced or lack of blood flow in legs. Feet wounds are likely to heal slowly contributing to gangrene and other complications.
MANAGE TYPE 1 DIABETES

CGM = Continuous Glucose Monitoring

Photo courtesy: U.S. Food and Drug Administration
REAL WORLD MISSING DATA IN TYPE 1 DIABETES PATIENTS

Continuous Glucose Monitoring device is not perfect!
Faulty sensors, incomplete records, mistakes in data collection, unavailability to report certain information (at a given time point) and others...
<table>
<thead>
<tr>
<th></th>
<th>CONTENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>What is time-series dataset &amp; how do we use the data?</td>
</tr>
<tr>
<td>2</td>
<td>Missing values and imputation - What can we do about that missing data?</td>
</tr>
<tr>
<td>3</td>
<td>Real world example: Predicting blood glucose level of Type 1 Diabetes patients at 30 minutes horizon</td>
</tr>
</tbody>
</table>
## THREE QUESTIONS

1. Which imputation methods are available?
2. How do we use them?
3. How do we evaluate imputation methods?
# Q1. Imputation Methods for Time Series Data

## Univariate Time Series Imputation
- Mean (Median)
- Last Observation Carried Forward
- Linear Interpolation
- Polynomial Interpolation
- Kalman Filter
- Moving Average
- Random

## Multivariate Time Series Imputation
- K-Nearest Neighbors
- Random Forest
- Multiple Singular Spectral Analysis
- Expectation-Maximization
- Multiple Imputation with Chained Equations
Q1. Imputation Methods for Time Series Data

**Canonical ML/DL modeling**
Use both past and future values

- Linear Interpolation
- Polynomial Interpolation
- Kalman Smoothing
- Moving Average
- K-Nearest Neighbors
- Random Forest
- Multiple Singular Spectral Analysis
- Expectation Maximisation
- Multiple Imputation with Chained Equations

**Real-time prediction (Online mode)**
Use only past value

- Last Observed Carried Forward
- Mean (Median)
- Random
- Empirical Based
Q2. HOW DO WE USE IMPUTATION METHODS

Today’s Sample Data: 24 hours Blood glucose levels
**SETUP**

**PYTHON**

```python
import pandas as pd
import numpy as np
import scipy as sp
import matplotlib.pyplot as plt

from fancyimpute import knn
import rpy2.robjects.packages as importr
imputeTS = importr("imputeTS")
kalman_StructTs = robjects.r['na.kalman']
kalman_auto_arima = robjects.r['na.kalman']

df = pd.read_excel('glucose.xlsx')
df.set_index('Minute_of_Day', inplace=True)
missing_minutes = list(df[df['Inferred_Glucose'].isna()].index)
```

**R**

```r
library (xlsx)
library (imputeTS)
library (pracm)
library(spatialEco)
library (VIM)

df <- read.xlsx("glucose.xlsx", sheetName = "Sheet1")
df[1] <- NULL
rownames(df) <- df$Minute_of_Day

missing_minutes <- df[which(is.na(df$Inferred_Glucose)), 'Minute_of_Day']
```
“TRADITIONAL METHOD”

Remove entire missing values.

When there are small fraction of missing values.

This is valid if the data is missing completely at random.

**PYTHON**

```python
df[...].dropna()
```

**R**

```r
df <- na.omit(df)
```
LAST OBSERVATION CARRIED FORWARD

Last observed value of a variable is used for all subsequent.
Can be either conservative or liberal depending on the context.

**PYTHON**

```python
df[...].fillna(method = 'ffill')
```

**R**

```r
imputeTS::na_locf(x)(x, option = 'locf')
```
MEAN VALUE

On average, the signal’s value is the mean.

**PYTHON**

```python
df[...].fillna(df[...].mean())
```

**R**

```r
imputeTS::na_mean(x, option = "mean")
```
LINEAR INTERPOLATION

What the graphing libraries do by default - draw a line connecting the two points at either end.

Can you do this while data is being collected? Nope.

**PYTHON**

```
import pandas as pd
df[...].interpolate(method='linear')
```

**R**

```
imputeTS::na_interpolation(x, option = "linear")
```
NEAREST NEIGHBOR

We can use the measured point nearest to the missing sample as the estimate.

PYTHON

df[...].interpolate(method='nearest')

R

pracma::interpl(x, y, method = 'nearest')
POLYNOMIAL INTERPLOATION

What if we take into account the trajectory of the signal at loss and reacquisition? We could fit a curve using the derivatives, which would be a polynomial.

Stineman is a special type of polynomial fit.

**PYTHON**

```python
df[...].interpolate(method='polynomial', order=2)
df[...].interpolate(method='polynomial', order=3)
```

**R**

```r
spatialEco::poly.regression(df$Inferred_Glucose, s = 0.2, impute = TRUE, na.only = TRUE)
```
SPLINE INTERPOLATION

Splines are an alternate way to fit “curvy lines” to data. They assume the use of “knots,” or fixed points, around which a semi-rigid curve is bent. This comes from pencil-and-paper drafting techniques.

**PYTHON**

df[...].interpolate(method='spline', order=2)

df[...].interpolate(method='spline', order=3)

**R**

imputeTS::na_interpolation(x, option = “spline”)
MOVING AVERAGE IMPUTATION

Missing values are replaced by moving average values. The mean in this implementation taken from an equal number of observations on either side of a central value.

**PYTHON**

```python
df[...].rolling(4).mean()
```

**R**

```r
imputeTS::na_ma(df[...], k = 4)
```
KALMAN SMOOTHING IMPUTATION

Filling missing values by estimating a joint probability distribution over the variables for each timeframe.

**PYTHON**

```python
kalman_StructTs(df[...], model = "StructTS")
kalman_StructTs(df[...], model = "auto.arima")
```

**R**

```r
imputeTS::na_kalman(df[...], model = 'StructTS')
imputeTS::na_kalman(df[...], model = 'auto.arima')
```
KNN IMPUTATION

For k-Nearest Neighbor imputation, the missing values are based on a kNN algorithm. In this method, k neighbors are chosen based on some distance measure and their average is used as an imputation estimate.

**PYTHON**

```
KNN(k = 5).fit_transform(df)[:, 0]
```

**R**

```
VIM::kNN(df[...],
variable = 'Inferred_Glucose', k = 5)
```
EMPIRICAL BASED IMPUTATION

Assumption: Patients have regular life schedule
## Q3. HOW DO WE EVALUATE IMPUTATION METHODS

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Mean Absolute Error (MAE)</td>
<td>( MAE = \frac{1}{N} \sum_{i=1}^{N}</td>
</tr>
<tr>
<td>2</td>
<td>Root Mean Squared Error (RMSE)</td>
<td>( RMSE = \sqrt{MSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y})^2} )</td>
</tr>
<tr>
<td>3</td>
<td>Pearson’s Correlation Coefficient (( R^2 ))</td>
<td>( R^2 = 1 - \frac{\sum(y_i - \hat{y})^2}{\sum(y_i - \bar{y})^2} )</td>
</tr>
<tr>
<td>4</td>
<td>Rank Product</td>
<td>( RP(g) = \sqrt{\left( \prod_{i=1}^{k} r_{g,i} \right)} )</td>
</tr>
</tbody>
</table>

Geometric mean of the ranks of imputation methods determined by MAE, RMSE and \( R^2 \)

Generalize performance evaluation
<table>
<thead>
<tr>
<th>CONTENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
</tbody>
</table>
PREDICTING BLOOD GLUCOSE LEVELS OF DIABETES PATIENTS: EXPERIMENTS IN DATA IMPUTATION

Our team were selected as finalists and we presented our work at IJCAI 2018 in Stockholm.
BUILDING PREDICTIVE MODELS FOR BLOOD GLUCOSE LEVEL
BUILDING PREDICTIVE MODELS FOR BLOOD GLUCOSE LEVEL

Collect 15 physiological information

- Training: Day 1
- Test: Day 49, Day 56

Missing value imputation

- 11 Canonical imputation
- Canonical imputation
- LOCF
- Empirical based imputation

Build predictive model

- Training set (imputed by 11 methods)
- 10-fold rolling cross validation

11 Optimize predictive models

- RMSE
- MAE
- PCC

Ensemble modeling

- Select top 5 best models
- Build ensemble model

Evaluate predictive model

Actual vs Predicted
COMPARISON OF METHODS FOR MISSING DATA IMPUTATION

Collect 15 physiological information

Training

Test

Day 1

Day 49

Day 56

Missing value imputation

11 Canonical imputation

Canonical imputation

LOCF

Empirical based imputation

Random

Emp-random

Emp-mean

Mean

KNN-2

KNN-4

KNN-6

KNN-8

KNN-10

MA

STI

SI

KA

SPI

LOCF

RMSE

0

40

80

120

160

200
THE EFFECT OF MISSING DATA IMPUTATION ON BLOOD GLUCOSE LEVEL PREDICTION

- Build predictive model
- Training set (imputed by 11 methods)
- 10-fold rolling cross validation
- 11 Optimize predictive models

(a) Train
(b) Test-canonical
(c) Test-empirical
(d) Test-locf

Klick Health
ENSEMBLE MODELING

Ensemble modeling

- Imputation methods:
  - RMSE
  - MAE
  - PCC

- Select top 5 best models
- Build ensemble model

Evaluate predictive model

<table>
<thead>
<tr>
<th>LI</th>
<th>LOCF</th>
<th>SPI</th>
<th>KS</th>
<th>STI</th>
<th>KA</th>
<th>MA</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>

Ranking of variances

- RMSE: 1 3 5 7
- MAE: 1 3 5 7
- PCC: 1 3 5 7

Rank product: 1 3 5
ENSEMBLE MODELING

**Patient 563**

**test-canonical**
- RMSE = 18.63

**test-empirical**
- RMSE = 18.64

**test-locf**
- RMSE = 18.62

CGM glucose (mg/dL)
- Timestamp range: 10-29 to 11-07

Actual vs. Predicted graphs for different test scenarios.
FOR YOUR WORK

1. We've shown 11 approaches for missing data that you can choose from

2. Important questions to ask:
   - Is your dataset univariate or multivariate?
   - Do you need the ability to make real-time predictions with missing data?

3. Pick error metric(s) and test the appropriate imputation methods using those metrics

4. For best results, ensemble the best-performing methods
KLICK – WHO WE ARE

RESEARCH
- Digital medicine
- Translational Research
- Publication
- Digital Biomarkers

DATA SCIENCE
- Data & Statistical Modeling
- Algorithm Development
- Machine Vision
- NLP
- Structured & Unstructured Data Engineering

TECHNOLOGY RESEARCH
- Technology Curating
- Proof of Concept
- Vendor Partnerships

Klick Labs
- Design Thinking
- Strategy
- Foresight

FAST COMPANY’S BEST WORKPLACES FOR INNOVATORS 2019 HONOREE
THANK YOU.

https://github.com/KlickInc/datasci-strata-talk-missing-data
Rate today’s session

Cyberconflict: A new era of war, sabotage, and fear

David Sanger
The New York Times

We’re living in a new era of constant sabotage, misinformation, and fear, in which everyone is a target, and you’re often the collateral damage in a growing conflict among states. From crippling infrastructure to sowing discord and doubt, cyber is now the weapon of choice for democracies, dictators, and terrorists.

David Sanger explains how the rise of cyberweapons has transformed geopolitics like nothing since the invention of the atomic bomb. Moving from the White House Situation Room to the dens of Chinese, Russian, North Korean, and Iranian hackers to the boardrooms of Silicon Valley, David reveals a world coming face-to-face with the perils of technological revolution—a conflict that the United States helped start when it began using cyberweapons against Iranian nuclear plants and North Korean missile launches. But now we find ourselves in a conflict we’re uncertain how to control, as our adversaries exploit vulnerabilities in our hyperconnected nation and we struggle to figure out how to deter these complex, short-of-war attacks.

David Sanger
National Security Correspondent
The New York Times

O’Reilly Events App