

# Data Ingestion and Data Preparation for Gaussian Processes

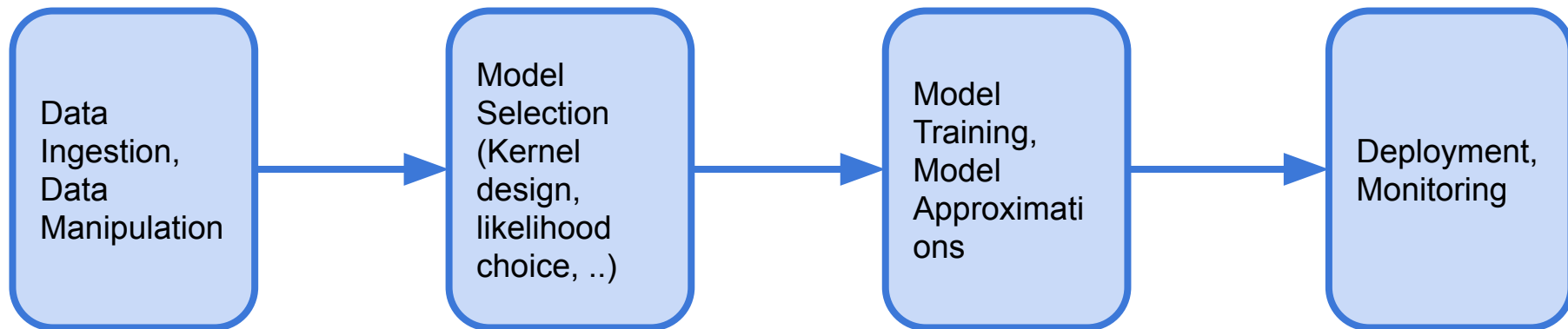
---

Alessandra Tosi

16 Jan 2020 - Gaussian Processes Meetup @Prowler.io



# Working with Gaussian Processes



# Outline

- Data science pipeline
- Data ingestion
- Data Preparation for GPs
  - Standardization
  - Handling missing values
  - Data Types
- Conclusion

# Data Science Pipeline

---

# Data Science Pipeline

Collecting and manipulating data is the first step of the data science pipeline.

Many concepts:



# Data Science Pipeline

With a data pre-processing routine you can:

- Improve your model selection routine
- Make simpler kernel choices and likelihood choices
- Improving training speed

# Data Science Pipeline

With a data pre-processing routine you can:

- Improve your model selection routine
- Make simpler kernel choices and likelihood choices
- Improving training speed



Caution: you can make mistakes!

# Data Science Pipeline

With a data pre-processing routine you can:

- Improve your model selection routine
- Make simpler kernel choices and likelihood choices
- Improving training speed



Caution: you can make mistakes!

If you want to perform a data science task in a scientifically sounded way, you need to acknowledge that each action on your data means that you add some assumptions to your problem.

NEW ACTION == NEW ASSUMPTIONS



# Data Ingestion

---

# Data Ingestion

- Assume we ingest data in a reasonable **tabular format**

# Data Ingestion

- Assume we ingest data in a reasonable **tabular format**
- Good practice: to record some relevant properties of the data which we know we will use to take action during the next steps (e.g. size and type)

# Data Ingestion

- Assume we ingest data in a reasonable **tabular format**
- Good practice: to record some relevant properties of the data which we know we will use to take action during the next steps (e.g. size and type)
- We record (but not impute) **missing data**

# Data Preparation for GPs

---

# Standardization

It is common practice to transform the data to have **zero mean** and **unit variance**.

# Standardization

It is common practice to transform the data to have **zero mean** and **unit variance**.

Problem of GP: to find the local minimum. You could overcome with many restarts, but it is time consuming.

So it is useful to initialize kernel parameters reasonably.

# Standardization

It is common practice to transform the data to have **zero mean** and **unit variance**.

Problem of GP: to find the local minimum. You could overcome with many restarts, but it is time consuming.

So it is useful to initialize kernel parameters reasonably.

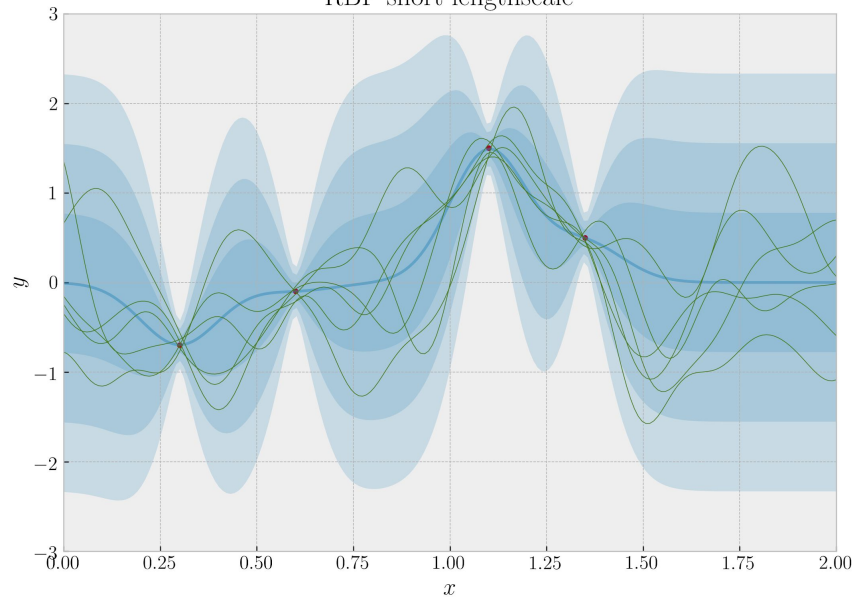


Caution: need to store the original mean and variance if you plan to manipulate new test points (this is one problem of data processing at training time: you need to do the same again at test time)

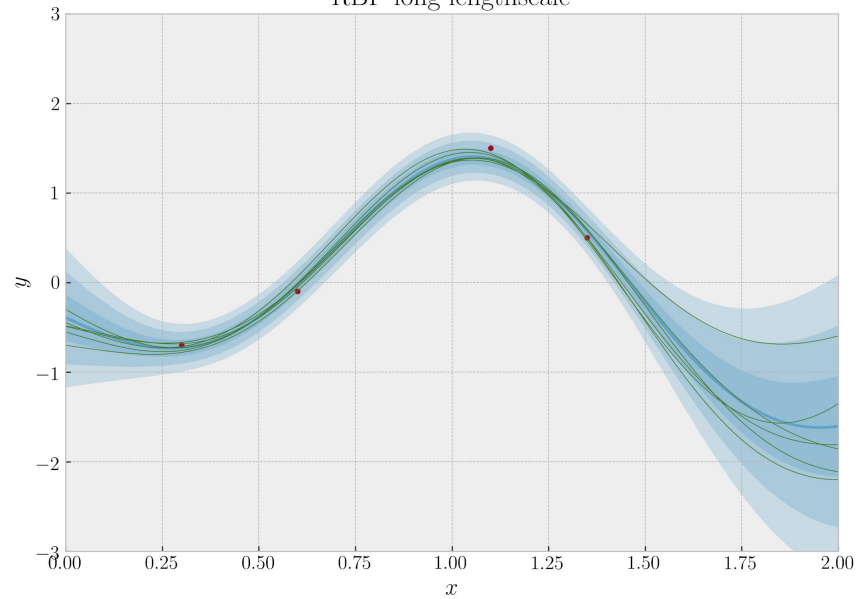


# Standardization

RBF short lengthscale

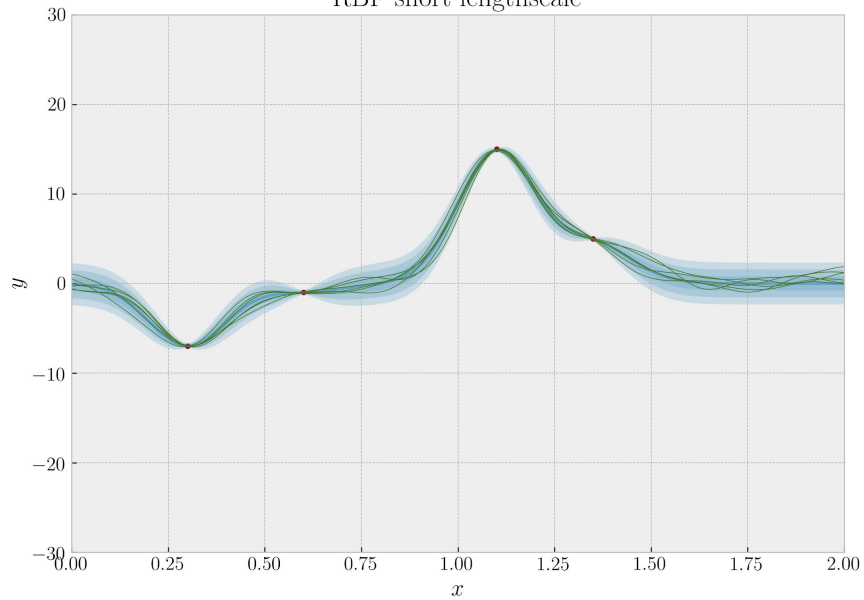


RBF long lengthscale

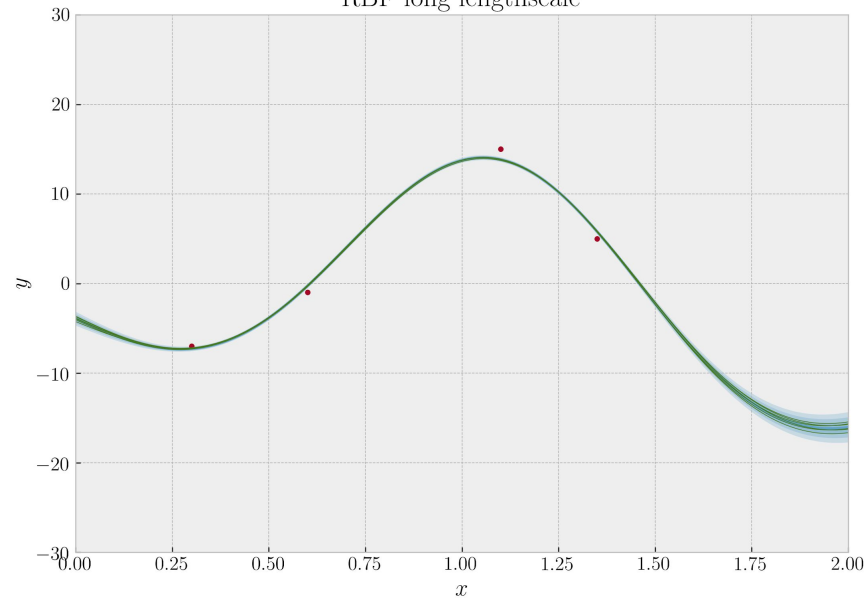


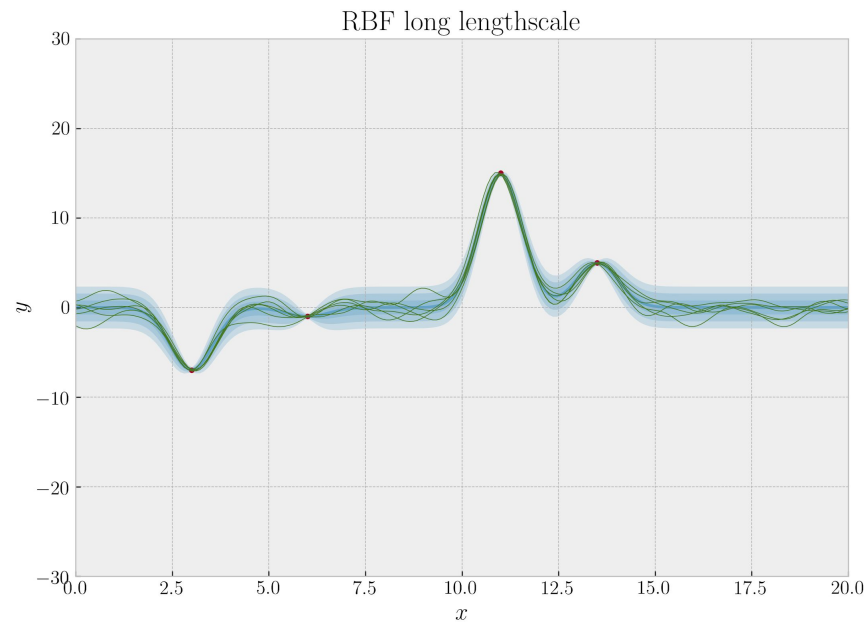
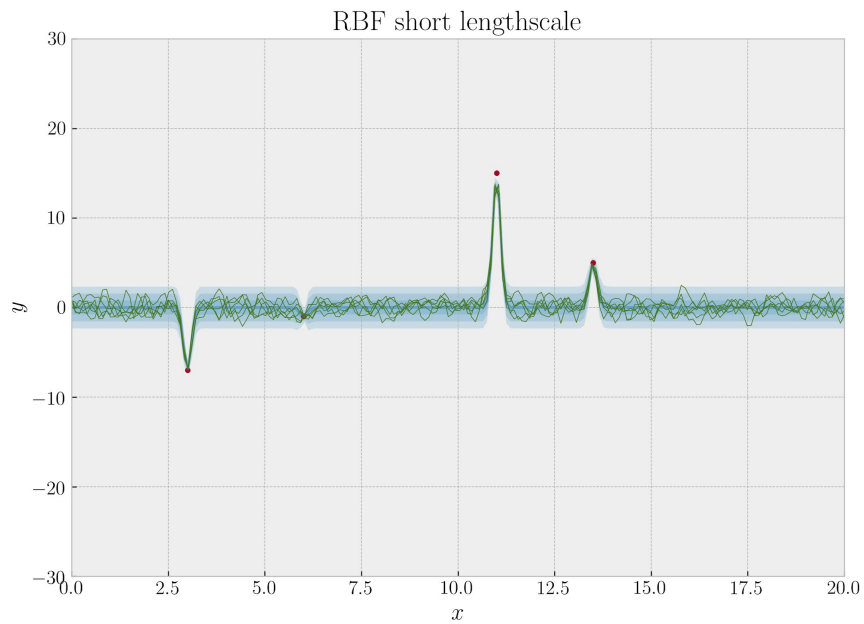
# Standardization

RBF short lengthscale



RBF long lengthscale





Reference:

Nicolas Durrande. **Kernel Design**. Presentation at the Gaussian Processes Summer School 2019 (<http://gpss.cc/gpss19/program>). Slides [here](#).

The automatic statistician project [<https://www.automaticstatistician.com/index/>].

# Missing Values

**Data imputation** (different methods: mean, median, most frequent, previous value, etc).

# Missing Values

**Data imputation** (different methods: mean, median, most frequent, previous value, etc).



Missing values can contain important information about your problem, if you impute them you might lose this information.

# Missing Values

**Data imputation** (different methods: mean, median, most frequent, previous value, etc).



Missing values can contain important information about your problem, if you impute them you might lose this information.

Always take into account model assumptions when doing imputation.

Data imputation can be used together with the model. [Quiñonero-Candela, Joaquin, and Sam T. Roweis. "Data imputation and robust training with Gaussian processes." (2003)]

# Time Index

It is common practice to **identify a time index** (e.g. date time) and transform it to float.



It is common practice to **identify a time index** (e.g. date time) and transform it to float.

## References to time series modelling and dynamical systems

- Roberts, S., Osborne, M., Ebden, M., Reece, S., Gibson, N., & Aigrain, S. (2013). **Gaussian processes for time-series modelling**. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 371(1984), 20110550.
- Thomas Schön. **GP-based probabilistic modelling of dynamical systems**. Presentation at the Gaussian Processes Summer School 2018 (<http://gpss.cc/gpss18/program>). Slides [here](#).

# Oddball data and outliers

It is common practice to delete “oddball” data.

# Oddball data and outliers

It is common practice to delete “oddball” data.



Outliers can contain important information about your problem, you might want to include them into your model.

# Oddball data and outliers

It is common practice to delete “oddball” data.



Outliers can contain important information about your problem, you might want to include them into your model.

You can also include these data into your model (heteroscedastic noise)

- Kersting, Plagemann, Pfaff, Burgard. (2007). **Most Likely Heteroscedastic Gaussian Process Regression**

# Data Types

GPs do not off-the-shelf work with all data types.

GPs do not off-the-shelf work with all data types.

If we need to work with a variety of data type, we identify two common practices:

1. To **transform** your data
  - E.g. use one-hot encoding
2. To **choose** an appropriate kernel / combination of kernels
  - E.g. use string kernels

---

## Automatic Type Inferential General Latent Feature Model

---

Neil Dhir

Thomas Rudny

Davide Zilli

Alessandra Tosi

### Abstract

With ever more data becoming available, there has been a recent drive to develop modelling tools for heterogeneous datasets such as electronic healthcare records. Therein appears both Bayesian nonparametric latent feature models, as well as methods for automatically determining the statistical data type (e.g. ordinal or categorical) of the attributes present in the data. We present a model which combines both of these attractive features in an end-to-end framework. By jointly learning the model complexity and statistical type of the data, we demonstrate that redundant information can be discarded while higher accuracy statistical type estimates are produced.

sample from a bag of M&M sweets? The former has order, the latter does not. Without further semantic knowledge, *heuristically*, this is an almost impossible inference task, but one which can be addressed by stochastic data modelling and which would have high utility in an AutoML framework.

### 1.1 On the importance of type

One of the primary assumptions in data analysis is that we can describe a data set of objects (examples), using a common set of attributes. Typically though, for further analysis to proceed, the attributes need to be assigned a *type* (categorical, ordinal, real, Boolean etc.). The type informs the inference, hence its importance. Valera & Ghahramani (2017) explains that prediction tasks differ depending on the kind of data we are considering. If it is an ordinal type, we employ ordinal regression. If it is a real type, we may employ

# Conclusion

---



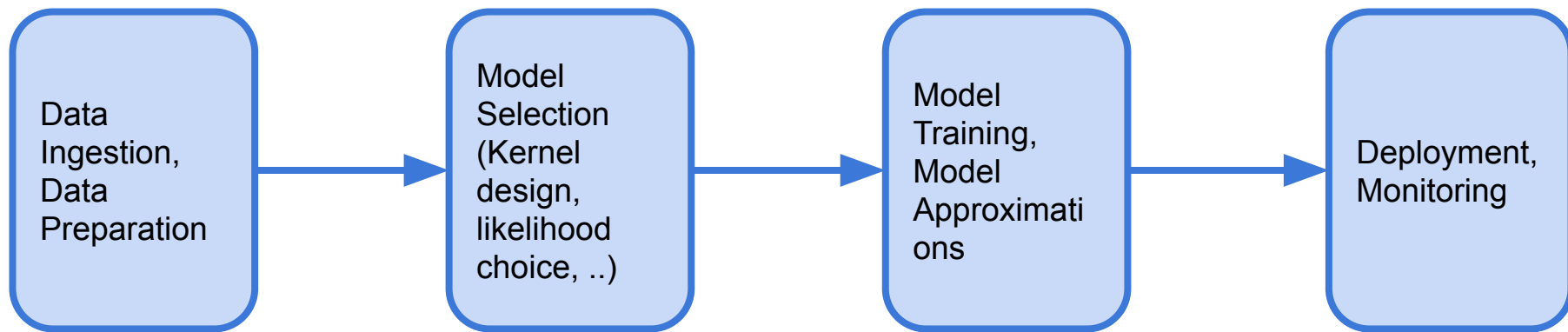
# Summary

1) We have discussed Data ingestion and Data preparation for GPs, with a focus on: standardization, handling missing values and data types.

2) We have discussed the relationship between the Data Preparation step with the other steps of the data science pipeline

# Summary

2) We have discussed the relationship between the Data Preparation step with the other steps of the data science pipeline.



# Take home message

Each **action** we take at each step of the Gaussian Process Modelling Pipeline has an impact on the other steps of the Pipeline because we are adding **new assumptions** to our problem.

# Questions?

---